SEARCH AND RESCUE MANAGEMENT: MODELLING AND DEVELOPMENT OF HEURISTIC STRATEGIES WITHIN A SIMULATION ENVIRONMENT

A thesis presented in partial fulfilment
of the requirements for
the degree of Doctor of Philosophy in
Operations Research
at Massey University, Palmerston North,
New Zealand.

Fiona Helen Wharton

2000

Where such references to "man", "he", or "his" appear in the body of this thesis, they have been used only to avoid the awkwardness of "man/woman", "she/he", or "his/her" constructions, and as such should be understood in their generic context.

Abstract

The search for a lost person on land has been the subject of relatively little research to date in comparison to other search problems. This thesis addresses this imbalance by examining the search for a stationary object that does not attempt to avoid detection. The problem is defined as a synthesis of the coverage, routing, and allocation problems that exist in the literature, and its complexity and unique aspects are discussed.

A physical model of the search terrain is developed using a Triangulated Irregular Network (TIN). This model incorporates the vegetation and natural features of the terrain, and is extended to model access paths and traversal speeds between any two points. A visibility model is developed over the TIN in order to define a detection model for both a human subject and any clues placed by him. Correction factors are used to model visibility and traversal speeds under different search environments.

Methods to define search regions as components of the elements of the TIN are described. Heuristic resource allocation methods are then developed for both the reconnaissance and general phases of a search operation. These methods allocate search tasks to resources individually or in parallel, and in real-time. Dynamic heuristic search strategies to respond to changing search conditions and the discovery of new information are then developed.

A Discrete Event Simulation (DES) model of a Search and Rescue (SAR) operation is developed. This model incorporates: siting a search base; search resource deployment and searching; clue and subject detection; communication between resources and search management; flooding and resource deployment under adverse weather conditions; and responsiveness of the subject over time.

The simulation model is used to perform some preliminary computational experiments on a restricted set of resource allocation methods and search strategies. Initial trends indicated from these experiments are: the general superiority of methods which do not

restrict the set of regions to be allocated for searching to an initial primary search area; the dominance of a night searching strategy; the dominance of using a sound detection method when a subject is responsive; and the benefits of applying diversifying search strategies.

Acknowledgements

The completion of this thesis has been achieved with the input, resources and encouragement of many people.

- *Tony* - my husband and best friend, for his unending encouragement, love and understanding, and for his patient data entry and editing.
- *My family* - for their years of support and continued attempts to understand what this "OR thing" is all about, and for their practical help with proofreading, editing and diagram creation.
- *John Giffin* - my chief supervisor, for his perseverance and technical critique.
- *Mark Johnston* - for his patience, time and practical expertise in sourcing, trialling and installing software for me, and showing me what to do with it. Particular thanks are due for creating the majority of the diagrams.
- *Fana To'omaga* - my brilliant typist who deciphered my scrawlings and dictation to put many of these words on paper; and to *Workbridge* for providing her funding.
- *Ross Gordon, Emergency Management Ltd.* - for his realist's point of view, total enthusiasm for the research, and the contacts he enabled with overseas SAR experts.
- *New Zealand Police Search and Rescue personnel* - particularly Inspector John Meads, NZ SAR Coordinator, for allowing me access to Police resources and providing the opportunity to participate in the national SAR training course.
- *Richard Rayner* - for initially showing me the ropes with C, LaTeX and UNIX.

- *My good friends* - for keeping me sane. Special thanks to *Kathryn Redhead, Shalome Campbell, Sharlene Pilkinton, Bernadette Murray and Nic Gidden*, for their help with typing and proofreading in the final weeks.

Glossary of Terms

The following specific terms are referenced throughout this thesis:

Binary Search A search method used to eliminate areas that the subject has not passed through.

Coverage (C) The ratio of search effort to the size of the search area.

Critical Separation The spacing of ground search resources at a distance equal to two times the visibility measure of that terrain.

Double Strip Search A form of grid searching where a region is searched twice from two different angles.

Hot Spots Likely places for physical clues to be detected.

Lateral Range (x) The perpendicular distance between a search resource and the search object at the point on the resource's path which is closest to the object.

Lateral Range Curve A curve depicting the cumulative probability of detection for a given search resource as a function of x, with one pass.

Mattson Consensus Technique A technique which guides search management to a consensus decision in defining POA values for search regions.

PDEN The Probability of Density. A measure used to rank search regions calculated on the POA value divided by the size of the search region.

Perimeter Cut A search technique where resources search along the boundaries of the search area for clues.

PLS The Point Last Seen represents the last known location of a search object.

POA The Probability Of Area.[1] The probability that the search object is located within a given search region.

POD The Probability Of Detection. The conditional probability that if the search object is in the search region it will be detected by the search resource. We use the word detection in the context of detection with recognition.

$\mathbf{POD_{cum}}$ The cumulative probability of detection over a number of successive searches of a search region.

POS The Probability Of Success of a search as measured by the detection of the search object and calculated as POA $\times$ POD.

$\mathbf{POS_{cum}}$ The cumulative probability of success over the search operation.

Priority Search Area (G_P) The subset of the search region graph which is identified as having the highest likelihood of containing the subject and on which searching is concentrated. This area is of a size that is able to be searched by the search effort on hand within one search period.

Probability Map A map of the search area depicting POA values in each subarea.

PSR The Probable Success Rate. A measure of the rate of POS increase to be expected when searching a region.

Repeated Expansion A search technique which begins by searching a small area centred on a specific reference point, and then successively re-searching this area in incremental expansions in an outwards direction.

Risk Map A visual map identifying hazards over a given region.

ROC Relevance Of a Clue to the search operation.

ROW The Rest Of the World. A pseudo-region representing any area outside of the defined search area.

Search Priority A myopic planning tool which ranks search regions for searching based on their contribution to POS and the time taken to achieve this.

Search Effort (Z) The area which can be effectively swept by a given search resource given its sweep width and the distance travelled by the resource.

[1] Also referred to in the literature as the Probability of Containment.

Search Object The object of a search. This includes human subjects as well as organic and inorganic clues.

Search Path The path of edges and/or triangular regions that a search resource is assigned.

Search Region A well-defined region assigned to a search resource for searching.

Search Region Graph (G) The 2-D graph derived from the TIN.

Search Resource A resource assigned to the search area to search for the search object. Such resources include human searchers, aircraft, dogs, and mechanical or electronic devices. Also referred to as a sensor, we refer particularly to a human searcher when using the term search resource within the body of the thesis.

Sector Stripping A method which removes search resources from one search region in favour of searching an alternative region in order to increase POS.

Sector Laddering A method which ranks search regions in a ladder formation with the top-most region having highest priority and regions being placed at the bottom of the ladder upon search completion. Regions whose priority is adjusted throughout the operation are moved to appropriate positions on the ladder.

Sound Sweep A search technique where search resources moving in a grid formation aim to detect a responsive subject by calling out at regular time intervals and listening for a response.

Sweep Width (W) The area under the lateral range curve - "a measure of the amount of 'detecting' being done" [150, page 4-4]. The sweep width differs for different search objects, search resources, and search conditions.

TIN Triangulated Irregular Network. A digital terrain model which geometrically partitions the terrain into triangles by a triangulation generated over a representative set of data points.

Track Traps Ground cleared for the purpose of observing if fresh tracks are laid.

Trail-Based POA A method of assigning POA values to a search area based on the possible behaviour of a subject initially known to have followed a marked path. POA values are estimated from the findings of a team who follow this path identifying and ranking decision points where the subject may have left the path.

The following variable definitions are utilized in the algorithm descriptions:

$start_k$ = the starting position (vertex) of resource k

$resources$ = number of search resources

$path_{k,j}$ = vertex at position j on the nodepath of resource k

num_k = number of vertices in the path of resource k

$time_k$ = amount of time required to complete the path of resource k

$path\_limit$ = the duration limit of any search path assignment

$path_{I,j}$ = vertex at position j in the intended path of the subject

num_I = number of vertices in the intended path of the subject

$time_I$ = amount of time needed to complete the intended path of the subject

$c_{i,j}$ = time cost of traversing edge (i, j)

$stpath_{i,j}$ = shortest path from vertex i to vertex j

$D_{i,j}$ = time length of shortest path from vertex i to vertex j

PLS = point which the subject was last seen at

$POS_{i,j}$ = POS value predicted from the search of edge (i, j)

$base$ = search base

$limit_k$ = 1 if resource k is at their search hour limit, = 0 otherwise

POA_i = POA of region i

POD_i = POD level at which region i is to be searched at

$POD_{cum,i}$ = cumulative POD of region i

$area_i$ = area of region i

$nregions$ = number of regions in the search region graph

$find\_team$ = resource which detects the subject

$urgency$ = urgency level of the search

$period\_start$ = commencement time of the next search period

period_end = time at which the current search period will be completed

down_time = amount of non-searching time between consecutive search periods

weather_level = level of current weather conditions

new_weather = predicted weather level arising at *weather_clock*

flood_time = the time at which regions of the TIN susceptible to flooding will flood and become impassable

lost_region = the region of the TIN in which the subject is located

cost_change = array which monitors the fraction of the current search task completed under differing environmental conditions, for each active resource

periodct = index of search periods

recall = indicator of whether or not resources are being recalled to the search base

suspend = indicator of whether or not the operation is being suspended

Table of Contents

List of Figures

List of Tables

List of Algorithms

CHAPTER 1

Introduction

"Now" said the rabbit. "This is a search and I've organized it ... "
"Done what to it?" said Pooh.
"Organized it. Which means ... well it's what you do to a search, when you don't all look in the same place at once ... "

— A.A.MILNE, 1928

1.1 Introduction

New Zealand as a country has a reputation for beauty and ruggedness — expansive national parks of forested canopies, tussocked slopes and crystal streams, and a people who "get out there and do it". Whether to tramp or climb, orienteer or hunt, the outdoors draws many to explore its freedom. However, at times experience can be too limited for the conditions, disaster or accident can strike, the conditions can turn hostile, or the route can be lost, and Search and Rescue (SAR) operations are called into place. The aim of this thesis is to model such operations and examine differing decision tools and planning methods as an initial investigation towards the provision of efficient search management techniques for New Zealand conditions.

The SAR problem, in its most generic sense, is that of search resources traversing a physical terrain in search of a number of objects whose whereabouts are unknown. The most familiar instance of this is a SAR operation undertaken to locate missing persons. Other types of search may take the form of searching for missing craft, or physical evidence in a criminal investigation. The terrain which the search is to cover may be urban, mountainous, bush or snow covered, consist of an expanse of water, or involve such specialist features as caves.

Specifically, the definition of a SAR operation within New Zealand Police policy is as follows:

"A search or rescue operation which is commenced with the aim of saving

> life, preventing injury, or removing a person from a situation of peril and includes any operation mounted solely for the purpose of recovering bodies from a remote location", [121, section 102].

Analytical approaches to searching were developed in response to military requirements during World War 2, with particular application to the search for enemy submarines. Since that time the original detection models and solution methods have been further developed and extended to incorporate: other search applications; the evolution of computers and advancement in computing power; and real-time responses to changes in the information on which the search is planned.

Of the specific search problems addressed, military applications, and particularly Naval applications, have received much of the research attention over the years. Searching for a lost person on land has, in comparison, been the subject of little research and it is this imbalance which this thesis aims to, in part, address. In particular, we seek to synthesize current techniques used in land SAR, classical search theory, terrain modelling, and dynamic routing heuristics, to develop a prototypical Discrete Event Simulation (DES) model of a SAR operation and solution approaches to the problem of allocating resources throughout the operation.

1.2 Thesis Objectives and Approach

1.2.1 Objectives

Stone [153] states that:

> "Planning a search is not solely an analytical exercise. Since subjective judgements are crucial to good search planning, search will always, to some degree, be an art. Nevertheless, applying good analytical techniques will improve the art and provide better search planning. Applying analytical techniques to those parts of an operational problem which can benefit from analysis is the heart of practicing operations research."

The aim of this thesis is to address a selection of those components of a land SAR operation which can benefit from the application of analytical techniques, with a particular emphasis on practical implementation rather than theoretical analysis. Specifically, the main objectives are fourfold, namely:

1. to develop a physical model of the terrain of a search area that captures the physical features of the region in such a way that enables the calculation of traversal speeds

and shortest time access paths between any two points under different weather and light conditions;

2. to develop a visibility model that is inherently linked to this terrain model and which responds to environmental changes, and to then extend this to develop a model for search object detection;[1]

3. to develop search strategies and analytical methods for search resource allocation which are capable of responding in real-time to the dynamically changing environment that typifies a search operation; and

4. to develop a flexible simulation model which allows varying search strategies and resource allocation methods to be evaluated under different search scenarios, and to use this to conduct an initial investigation into a subset of the resource allocation methods developed.

In order to meet these objectives we limit the scope of this thesis to that of the search over land for a single stationary human subject who does not attempt to avoid detection. The search application is civilian rather than military and search resources are limited to human ground searchers of identical skill. The major thrust of the study is the modelling of different aspects of a SAR operation, the development of resource allocation methods and dynamic search strategies, and the development of a simulation approach which enables different search strategies and solution methods to be tested; consequently the computational experimental component of the study is only cursory, creating a foundation for future research.

1.2.2 Content Outline

The study begins by reviewing the development of search theory, and the statistics and current methods used to predict: subject movement and final location; subject detection; and overall search success. Chapter 3 then presents the organizational structure, decision tools and search methods currently used in executing land SAR operations, including computer applications available to assist search management in their decision-making. The limitations of these methods and strategies are discussed, together with the potential of greater incorporation of developing technology, such as the Global Positioning System (GPS).

Chapter 4 examines the research field of terrain modelling, reviewing the advantages and disadvantages of existing digital terrain models in terms of their applicability to

[1]When the word 'detection' is used within the body of the thesis it refers to detection associated with identification.

modelling the search area. As a result of this investigation a Triangulated Irregular Network (TIN) is selected to develop a terrain model of the search area on which to model the movement of both the subject and the search resources, and enable definition of individual search regions. This basic model is then enhanced to incorporate vegetation, specific terrain features such as ridges and streams, and traversal speeds. A visibility model is then defined, and correction factors are incorporated to adjust both visibility limits and traversal speeds, in response to changing search conditions.

The study then reviews subject detection models described in the literature and compares these to the experimental field results available. A subject detection model is then developed to operate over the TIN representation. Two types of subject movement are considered and these are modelled as paths over the terrain model. Clue placement is simulated as the subject moves. Chapter 5 concludes with the development of a clue detection model and a method to update the subject location probability distribution on the discovery of a clue. This chapter completes the components needed to physically model the search environment and search process.

Having developed the physical search model, the next section of the thesis concentrates on the development of resource allocation methods. Chapter 6 explores the fundamental aspects of the SAR problem and how it relates to existing coverage, routing and allocation problems in the Operations Research literature. The problem is shown to be NP-complete and unique in many respects. Mathematical formulations are presented for different search phases, for completeness, rather than for implementation.

The initial reconnaissance phase of searching is addressed in Chapter 7, and heuristic methods are developed to mimic current search practices and objectives. Additional methods are then proposed to allocate resources to search regions, in order to meet objectives that are different from those currently followed. In particular, the objective of hedging the current search strategy against incorrect subject location assumptions is addressed. Further methods are developed to construct reconnaissance search paths for a number of search resources simultaneously, adapting routing methods from the literature. A selection of different ways in which the spatial coverage of search regions can be achieved in the general search phase of the operation are examined in Chapter 8. The problem of defining the boundaries of search regions is also considered and heuristic methods are proposed to accomplish this. The general problem of effort allocation in this search phase is then addressed and solution approaches are presented.

The specific dynamic and stochastic components of a search operation are discussed in Chapter 9 within the framework developed by Séguin *et.al.* [146] for real-time decision problems. Dynamic real-time strategies for solving the SAR problem under changing knowledge and search conditions are presented.

The development of a model to simulate a search operation is presented in Chapter 10. Discrete event simulation is used due to the nature of the problem, with the response to each event that may occur being governed by a set of rules outlined in Chapter 11. Further heuristic methods are developed within this chapter to implement the simulation of each stage of a SAR operation. These include such decisions faced by search management as: where to locate the search base; how to redeploy resources already searching in the field; and how to coordinate the retrieval of an injured subject.

Chapter 12 concludes the study by using the physical and simulation models developed in the preceding chapters to conduct some preliminary computational investigations into the relative performance of four resource allocation methods under three representative search scenarios. The results of the experiments are analysed and conclusions presented. The contributions that the different components of the study makes toward solving the problem of detecting a single, stationary human subject lost on land are summarized in Chapter 13, and avenues for future research are outlined and discussed.

CHAPTER 2

Search Concepts

"The theory of search, as a scientific discipline uniting physical and operational facts through mathematical concepts and theorems, may fairly be said to date from World War 2, and to have received its major impulse from various phases of the war at sea, in particular, those involving the submarine. At the centre of the development of this new branch of operations research was the group of scientists assigned during 1942–1945 to the Commander in Chief, US Fleet, designated first by the name Anti-Submarine Warfare Operations Group (ASWORG); later, Operations Research Group (ORG)."

— KOOPMAN [100, PAGE 1]

2.1 Search Theory

Four eras in search theory are identified by Stone in 1989 [154]. The first era from 1942 to 1965 is described as the Classical Era, followed by the Mathematical Era from 1965 until 1975. The Algorithmic Era then followed, up until 1985 where it was superseded by the Dynamic Era.

The Classical Era gained momentum with the work by Koopman and other members of ASWORG who, at the close of the war, collaborated to compile and publish the search theory developed during the war years, illustrated by applications [100]. These applications drew largely upon the search for hostile submarines and as such concentrated on passive searching, *i.e.*, searching which does not alert the search object, in comparison to active searching where the presence of searchers can alert the subject that it is being searched for. A theoretical mathematical framework for the search of an object was developed, including models of object motion and location, and detection. Effort allocation techniques were also described and illustrated with military examples, with the emphasis being placed on the passive search of hostile targets in motion.

Stone [152] presents the major results in the field of search theory through to the end of the Mathematical Era. The main emphasis of the Mathematical Era was to seek to understand the mathematical nature of search problems, with necessary and sufficient conditions being found for a general class of optimization problems involving a stationary subject. Results during this period tended to be "presented in a classical theorem-proof format", [154, page 502]. Most results centred around a stationary subject with an extension to models of limited subject motion, for example, two-celled Markovian motion and conditionally deterministic subject motion. The problem considered was that of allocating search effort optimally to detect a subject when a prior subject location distribution was known along with an effort-related detection function. Pursuit and evasion problems were not considered. Results were presented for problems where the search space was continuous or discrete, where the search effort applied was continuous or discrete, and where the detection function was regular[1] or non-regular.

The Algorithmic Era focused mainly on expanding the understanding of and approaches to solving moving subject problems via algorithms, which drew upon the greater availability of computing power. One-sided search problems were mainly studied, where the subject does not react to the searcher. Stone [154] divides such problems into two classes: optimal search density problems and optimal searcher path problems. The first class of problems are where the search effort is infinitely divisible and where "the application of search in one place and time does not constrain the location of effort at any other time", [154, page 502]. Location of search effort is constrained by the search effort previously expended in the optimal searcher path problem. This problem class can be further divided into discrete and continuous, time and space problems.

Of particular interest are the complexity results shown by Trummel and Weisinger [166] in 1986 for the optimal searcher path problem involving a stationary subject in discrete time and space. Trummel and Weisinger considered the search for a stationary subject over both a finite and infinite search time. The search was conducted over a finite, connected graph and the subject was located at a single vertex on the graph. Two versions of the problem were examined by Trummel and Weisinger. In the first, the search was carried out within a fixed time limit, and the effectiveness of the search was measured by the probability of detecting the subject. The authors showed that this problem is NP-complete by transforming the NP-complete hamiltonian path problem to it. The second version of the problem had no time limit on the search, with the measure of effectiveness being the mean time to the detection of the subject. This problem was shown to be of NP-hard complexity. Due to their complexity these problems require heuristic solution

[1] A regular detection function b is defined by Stone as one whose derivative is continuous, positive, and strictly decreasing, and where $b(0) = 0$ [152].

methods, thus further extensions to the problems which consider searches conducted over a more complex representation than a finite, connected graph and which use more than one searcher, also require such solution methods.

Unlike the previous eras where the search plan was followed until the subject was detected, the Dynamic Era began to incorporate subject and environmental information which arose as the search unfolded. Dynamic search planning systems were developed for Naval applications and utilized further computational advances; such systems modify search plans by maintaining a real time estimate of the subject's location based on all available information.

2.2 Application to Land SAR

Mathematical techniques and computing technology have been used successfully for many years in marine and aviation searches, but have only been seen in the domain of land SAR operations since the early 1970s. In 1975 Syrotuck [159] introduced the application of a mathematical approach to land search by adapting tools used in searching for lost ships and aircraft. His work is described by the USA National Association for Search and Rescue (NASAR)[159, page i] as providing "the basis for modern search and rescue management philosophies and providing a useful tool for strategy and tactics planning." Since this time there has been a gradual increase in the application of mathematical, statistical and psychological sciences to decision-making, with mathematics becoming "firmly entrenched into the theory, if not necessarily the practice, of search planning" in the 1980s [24]. The development of these approaches has largely been seen and documented in the United States of America (USA), Canada and the United Kingdom. However, more recently there has been a gradual introduction and discussion of many of these principles and techniques internationally, including New Zealand. This has been particularly seen in the the form of training courses run by the Emergency Response Institute (ERI), USA, which accumulates new research and data from around the world. The content of these courses is referred to by Swombow [158, page 27] as a "system of search management, which is now accepted as an international standard."

However, the management procedures developed have revealed a "disconnect between the science of search theory and practice of searching in the inland environment" [58] due to this development occurring largely independently of the classical search theory literature. Hence, a re-education of practitioners has occurred in the last couple of years via conferences and Internet communications to correct incorrectly held assumptions. In particular, Frost has compiled introductory search theory articles with the purpose of establishing a correct and common understanding of definitions and theories amongst land search managers [62, 63, 64, 65].

While the utilization of a scientific basis for searching is gaining momentum and is strongly advocated by many, its validity in the field is debated by others. This is especially true among those used to previous "tried and true" methods. Roberts [137, page 11] cautions that "any attempt to saturate everyone with science will kill it before it starts." He advocates teaching the skills and working knowledge of the science to the level of detail that those working in the field require. Perkins [134, page 49] considers "that simplification and demystification are important steps in the overall process of gaining general acceptance within [his] Team of the importance of Search Management per se."

Generally, acceptance of the science is occurring as the evidential proof of successful results is attested to, and where, in countries such as the USA, accountable decision-making is called upon when search outcomes result in litigation.

Syrotuck [159, page 43] states that:

> "Search and rescue is both an art and a science. There are theoretical approaches and practical ones. As one must compromise between optimum and realistic planning, one should also compromise between the art and the science. Search directors should never become slaves to the computers, nor should they be slaves to tradition."

The elements of an optimal search as defined by Koopman and cited by Stone [154, page 501] comprise:

> "a prior distribution on target location, a function relating search effort and detection probability, a constrained amount of search effort, and the optimization criterion of maximizing probability of detection subject to a constraint on effort."

These elements of a search are now detailed — predicting subject movement and location, resource detection, and ultimately search success.

2.3 Subject Movement and Location

The probable location of a search object is described by a probability density distribution. This distribution is generated from known subject information, local information and historical statistical data. If the subject is assumed to be in motion then the location distribution is developed in consideration to a model of subject motion. Due to varying terrain structure and the presence of terrain barriers in land-based search operations, standard probability density distributions, such as those employed in marine SAR, are not applicable. Hence, the best approach to generating a location probability density distribution for such operations is to divide the search area into logical regions and assign each an estimated probability of containing the search object [150]. A *probability map* can be created for a given time in the operation by overlaying a grid onto the search area and computing probabilities of containment for each grid cell, based on the assumption of a uniform probability density within each region [150]. The term ***Probability of Area (POA)*** is used to define the probability that the object being searched for is located in a given region.

Although establishing POA values is a critical step of planning upon which later search strategy is dependent, Colwell [27] describes it as "one of the weakest aspects of search planning" which "remains a somewhat subjective and poorly substantiated 'black art' ". The process is further complicated by the difficulties people have in estimating probabilities. Tversky and Kahneman [168] overview studies researching how people estimate probabilities via a set of heuristics. In particular they identify three heuristics used: representativeness; availability; and adjustment-and-anchoring.

Representativeness estimates the probability that an event A belongs to a class B by how much A resembles B. This heuristic is shown to be insensitive to: the prior probability outcomes; the size of the sample; and predictability. Further it is also influenced by misconceptions that the person may have, particularly regarding the role of chance and regression. Additionally, if the information they are basing their estimation upon is limited in accuracy, this is often overlooked if there appears to be a strong match between A and B. This bias is termed 'the illusion of validity'.

The availability heuristic is employed to estimate the probability of an event occurring and is based upon how easily such occurrences can be remembered. This technique can lead to biases due to the ease at which instances of the event can be retrieved. More familiar, salient or recent occurrences are shown to result in higher estimations of probabilities. The estimation of the probability of the event occurring is also affected by how the person searches for occurrences of such events. The ability to imagine or construct events also leads to biases, with those events more easily imagined leading to higher probability estimates and an underestimation of those events which are "either difficult to conceive of, or simply do not come to mind," [168, page 13]. 'Illusory correlation' is also another bias associated with availability, where events with strong associations are determined as frequently co-occurring, often being given greater weighting than is deserved.

The third judgement heuristic examined is that of adjustment and anchoring. Here a quantitative prediction is derived by adjusting an initial value (the anchor). Differing initial values are shown to lead to differing estimates.

Tversky and Kahneman [168, page 20] conclude by stating:

> "These heuristics are highly economical and usually effective, but they lead to systematic and predictable errors. A better understanding of these heuristics and of the biases to which they lead could improve judgements and decisions in situations of uncertainty."

In reference to these biases, Stone [154, page 504] notes that:

> "If these biases are indeed characteristic of how people operate, then an expert system, which is designed to mimic what a person does, is doomed to be a poor estimator of probabilities and to produce poor search plans."

The heuristic of availability is perhaps the most utilized in developing search scenarios and POA values, with imaginability, familiarity, recency and saliency impacting on such estimates. Conclusions drawn from the subject profile and statistical history of the area would also be influenced by the representativeness heuristic and its associated biases. By estimating POA values via a consensus method and utilizing experienced decision-makers it would be hoped that some of these biases are 'diluted'. However, the rigidity to which any derived POA values are adhered to should be cautioned, and the uncertainty of the situation should be reduced by utilizing as much available information as possible before making any initial estimates.

Methods of assigning POA values that attempt to address the weaknesses of probability estimation are now examined.

2.3.1 Consensus-Based Probability Of Area

Swombow identifies the three elements which impact on the success of a search as: resources; strategy and tactics; and most crucial, the function of management. Swombow [158, page 28] advocates "never plan alone" but to instead form consensus amongst a search management team. Hill [88] concurs with this, stating that "consultation requires discussion, and discussion facilitates a rational and systematic approach to search planning, where ideas are analysed and re-evaluated through dialogue."

Recent approaches have advocated that POA values be determined by combining the joint expertise of the search managers on a particular operation, in preference to sole reliance on historical analysis or one decision-maker's judgement call. One such method is the Mattson Consensus Technique[2] presented in [103]. In this method the search area is divided into search regions and each person on the management team independently assigns a POA value to each search region. This value relates to their perceived likelihood

[2]The method was named after Bob Mattson who applied the technique within the USA Air Force.

that the subject will be in that area — a decision that is made on the basis of known information, experience, local knowledge and intuition. The assigned values must sum to 100 over all the regions, including an additional region which represents the *Rest Of the World (ROW)* — any area not inside the defined search area. These individually established POA values are then averaged over the search managers to give a POA value for each search region.

Colwell [27] describes the consensus decision method as:

> "a semi-subjective blend combining the previous incident history for the region, local terrain evaluation and a judgement of the expected mental and physical condition of the subject."

The complexity of this approach increases with the number of search regions and decision-makers. Some practical criticisms of this approach have been raised by Perkins [134]. Perkins [134, page 45] states that the scale of 0 to 100 becomes unmanageable, observing participants who "tended to simplify the values into steps of 5% and then fill in the leftover sectors with values which typically added up to 10%." In addition he states that having to ensure that these decisions sum to 100 leaves the participants "being asked to do two things at the same time", [134, page 45]. This second concern was evident in observing those involved in following the Mattson technique at an exercise at the New Zealand SAR training course, April 1996.

Perkins [134] presents a consensus region weighting method which addresses these concerns. The essential difference in approach is that each decision-maker is required to give each region a value between 0 and 10 to indicate the likelihood that the subject will be in that region, with 0 representing an opinion of no likelihood and 10 representing an absolute opinion of the subject's location in that region. The scale is also an indicator of relative importance. Totals are then found for each region over all the decision-makers and used to rank the regions for searching. To assign POA values to each region in order to measure how much priorities change between search phases, a ROW region is established and its POA value determined by consensus among the decision-makers. The values of the remaining regions are then normalized to arrive at actual POA values for each region. Perkins emphasizes that the approach should be simple and meaningful for those not familiar with POA terminology.

2.3.1.1 ROW

Cornell [31] recommends that search managers examine statistical records of search histories to obtain empirical estimates for the POA value assigned to the ROW region. In particular, he cites the research into 163 search operations conducted in wilderness areas of South Western Alberta, Canada. In 131 of these operations the subject was found in the search area defined in the initial operational period indicating an appropriate initial ROW estimate to be 20%. Cornell states that such information (when recorded over multiple operational periods) can be used to guide resource deployment in extended operations.

In reply, Tate [161] states that "I can't recall any searches where the subject was outside the search area and still lost (several where the person was outside the search area in, say, another town)." Tate questions Cornell's analysis of the ROW POA and queries whether search managers were more inclined to create search areas which were too small or too large in the operations studied. He also considers that if the ROW POA value is approaching 10% this would require a serious look at the operation "to see why we thought that there was a real probability that the subject wasn't in the search area."

Swombow [158] uses the value of the ROW region to monitor search progress with possible justification for an extension of the current search area or a suspension of the operation if it reaches a significantly high value.

2.3.2 Scenario Analysis

Conducting a search operation under conditions of uncertain or contradicting information will generally result in search management considering a number of scenarios of what may have taken place, and, in particular, direction options which the subject may have chosen. Such alternative scenarios result in differing probability density distributions and hence different POA values being assigned to individual search regions. Hill [90] presents an approach which integrates "competing scenarios into a single plan". The approach considers the problem of locating missing persons on land and adapts procedures utilized by the USA Coast Guard towards analysing the possible causes behind the disappearance of vessels and their subsequent movements.

The mathematical procedure derives a single set of POA values and draws upon Bayesian probability analysis by allocating a weight to each possible scenario. This weight is then multiplied to the POA value of a search region (this POA value being determined for that specific scenario) and summed over all scenarios considered. This results in a single POA value for that region which can be utilized in planning search strategies. The ROW region is also evaluated for each scenario. The resulting POA values are updated as search regions are searched without success, and the initial weights assigned to individual scenarios may also 'shift' as the search progresses and new information comes to hand.

The POA value of a search region i is calculated as:

$$POA_i = w_1(POA_{i1}/POA_{sum1}) + w_2(POA_{i2}/POA_{sum2}) + \ldots + w_n(POA_{in}/POA_{sumn})$$

where w_n=weight (as %) assigned to scenario n and $sumn$= the sum of POA values over all regions for that scenario. Effectively the approach normalizes POA values for each scenario with respect to the scenario weight.

The *scenario analyzer* computing software written by Hill to compute POA values by this method assigns each search region a likelihood rating of 1 to 9 under each scenario.[3] A rating of 1 represents a low possibility of the region containing the subject and a rating of 9 represents a high possibility of containment. The weightings of each scenario are also input into the program as a rating of 1 to 9. The rating scheme simplifies the process of assigning POA values and the software assigns a uniform probability distribution to this qualitative scheme. For example, a small problem with two scenarios rated 6 and 4, a ROW region rated 1 under both scenarios, and two search regions assigned the ratings 7 and 4 under the first scenario, and 5 and 8 under the second scenario, have their POA values calculated as:

Region	Scenario One	Scenario Two	POA Value
1	$0.6 \times 7/12$	$0.4 \times 5/14$	49%
2	$0.6 \times 4/12$	$0.4 \times 8/14$	43%
ROW	$0.6 \times 1/12$	$0.4 \times 1/14$	8%

If multiple decision-makers are involved then the POA values are averaged over all planners. As each decision-maker estimates the likelihood of the possible scenarios and then evaluates POA values conditional on these, the approach circumvents the problems which arise with the Mattson consensus method of POA definition when several scenarios may be discussed among the planners before individual POA ratings are assigned. The method also avoids the difficulties that people have in estimating conditional probabilities and does not require the decision-maker to sum region POA values to 100 as part of the process. Hill recommends that the order in which individual decision-makers analyse scenarios be varied and that any scenario having a "non-trivial possibility of being valid," *e.g.*, a 10% criterion, be included.

The simple and systematic approach aids decisions on strategy and resource deployment by avoiding decisions such as how long should one scenario be followed before considering alternative possibilities. Frost [64] defines the term *scenario lock* to such instances where search management fixes on one scenario to the exclusion of others. In particular he cautions against planning on assumptions where "an assumption, if repeated too often and questioned too seldom, will gradually take on the appearance of fact."

[3] A variation of the O'Connor method, named after its originator.

2.3.3 Statistically-Based Probability of Area

Colwell [24] refers to the accurate prediction of the subject's movements as the " 'Holy Grail' of SAR planning". He recommends that more detailed information be gathered from search operations to form a database on which to determine "new Rules of Behaviour". With a large number of operations and hence statistics gathered, patterns can emerge within the data which allow predictions to be formed as to the possible movements of a lost person in a newly initiated search — drawing upon the profile of the subject. In addition to statistical databases there exists the expert knowledge of local people who, from their own experience and local terrain knowledge, are able to offer invaluable insight into the probable movements of different categories of subjects. However, unfortunately these people take their knowledge with them when they move out of the region. Statistics from SAR operations conducted in the USA [103, 159] have collected information on the lost person's age, sex, fitness, experience, activity, mental condition and their movements if known, including the distance and direction that they travelled from the initial track or separation point.

Within New Zealand the number of SAR operations conducted within any one region is significantly less than those in the USA and statistics for each region are scant. Currently the only knowledge base which exists in this form are collective statistics for the country as a whole. These statistics are collected by the Police staff co-ordinating each operation, and are forwarded to Police National Headquarters where they are incorporated once a year into a national set. Since 1994 there has been an attempt to generate a more comprehensive database than in previous years and the statistics produced now contain many of the elements gathered in American statistics. However, there are currently no statistics available on the actual movement of subjects. In response to this, Meads states in the 1997/98 SAR statistical report that future report forms will include a field to measure the distance between where the subject was found and the point where they were last seen [122]. This had not yet been actioned by the 1998/99 report. With greater resource investment in both the collection and analysis of such statistics trends could be analysed, particularly at a region-specific level, with the increased potential for predictive modelling.

LaValla and Stoffel [103] recommend that regions be assessed before the event of a SAR operation with the aim of identifying potential search situations, based upon demographic information, geographic features, historical statistics, seasonal activity and available resources. This is primarily a planning tool and can be specifically applied to produce *risk maps.* These visual maps identify hazards and risks in specific regions. Such maps can be useful in identifying locations where, due to a lack of distinguishing terrain features, people become disoriented and lose their way, with similar incidents occurring over a period of time.

McDonald [117], of the National Search and Rescue Secretariat (NSS), Canada, considers the feasibility of developing a national ground and inland water SAR database. She proposes over the Internet a rationale and overview of the design of such a database, inviting comment by potential users [117].

The data elements of the proposed database fall into three groupings: information on the incident; the subject; and the response. The database would collect data to address questions applicable to SAR operations internationally, such as:

- what differences exist between various geographic regions?
- what seasonal effects exist and do they vary geographically?
- what incidents result in serious outcomes and could these outcomes be improved by changes in the SAR response?
- which people are most at risk of becoming the subject of a SAR operation, and which types of activities being undertaken are more likely to result in SAR operations? — are prevention methods effective?
- should those undertaking high risk activities carry communication equipment or transmitter beacons? — to what extent are these currently utilized and how does this impact upon the outcome of the operation?
- what level of impact do weather conditions have on the operation and its outcome? — and are some regions more affected than others?

Sweere [155] defines a stereotype as:

> "a set of pre-determined values for variables describing an area, a lost subject, command post, or a situation ... Stereotypes assist in rapid preprocessing of probabilities, are based upon many prior searches, and can be tailored to the search group's specific area of operations."

The specifics of stereotypes can be modified to meet the current situation. Sweere considers, as a future possibility, the use of stereotypes and algorithms to define POA values.

We now look at the specifics of statistical information gathered on the behaviour of subjects in New Zealand and the USA.

2.3.3.1 Lost Person Behaviour — New Zealand Statistics

In New Zealand, land SAR operations are initiated as a result of a number of causes. These are detailed in Table 2.1 for the years 1994–1999. The category for 'psychological disorder' was introduced in the 1996/97 period, as was 'attempted suicides' (combined in the table with suicide fatalities). However, in subsequent years the 'psychological disorder' category was instead represented in the type of subject tabulation.

Table 2.1: The cause of NZ land SAR operations.

Cause	1994/95	1995/96	1996/97	1997/98	1998/99
Medical	159(31%)	118(24%)	117(24%)	128(30%)	104(26%)
Navigational	130(25%)	104(22%)	92(19%)	93(22%)	81(20%)
Weather	44(9%)	37(8%)	35(7%)	33(8%)	35(9%)
Rivers	43(8%)	21(4%)	14(3%)	6(1%)	25(6%)
Inexperience	18(4%)	2(< 1%)	11(2%)	9(2%)	6(2%)
Suicide and attempts	16(3%)	17(4%)	11(2%)	17(4%)	10(3%)
Equipment	7(1%)	15(3%)	16(3%)	10(2%)	7(2%)
Avalanche	4(1%)	8(2%)	5(1%)	4(< 1%)	–
Alcohol	2(< 1%)	–	3(< 1%)	6(1%)	2(< 1%)
Homicide	1(< 1%)	–	1(< 1%)	–	2(< 1%)
Flares negative	–	1(< 1%)	–	–	1(< 1%)
Engine failure	–	–	2(< 1%)	–	–
Psychological disorder	–	–	25(5%)	–	–
Overdue	–	11(2%)	48(10%)	43(10%)	32(8%)
Non-specified	87(17%)	149(31%)	101(21%)	74(17%)	93(23%)
Total	511	483	481	423	398

A large percentage of operations arise from navigational problems. In particular, as stated in the New Zealand Bushcraft Manual [75, page 92], these occur when a party:

- "picks the wrong spur[4] when coming off the tops[5]";
- "drops into the bush when caught out in bad weather on the tops, without knowing which watershed[6] is below"; and
- "mistakes a junction[7] and travels up the wrong creek".

Many areas have specific places where confusing navigational references exist and people lose their way time after time. In particular "trampers and hunters are more

[4] A projection from a mountain or mountain range.
[5] Above the bush line.
[6] The line of separation between waters flowing to different rivers.
[7] A point where more than one stream or track meet.

likely to lose tracks when going downhill than uphill. When going uphill, all spurs and ridges tend to converge, whereas coming downhill they diverge", [75, page 70]. In addition many hazards exist within the bush itself such as natural terrain features, *eg.*, waterfalls, gorges, fallen logs, algae-covered rocks and wasp nests.

Those activities leading to the greatest number of land SAR operations over the previous nine years are recorded in Table 2.2 and the subjects of land SAR operations for the years 1994–1999 are defined in Table 2.3. The group making up the largest number of search subjects was males aged 20 to 29 years, consistent over the nine year period (1990–1999). Statistics were not available for the breakdown of groups and ages for specific activities. The largest injury categories for those same years (1994–1999) are presented in Table 2.4.

Table 2.2: The activities resulting in NZ land SAR operations.

Activity	90/91	91/92	92/93	93/94	94/95	95/96	96/97	97/98	98/99
Tramping	152	163	134	181	179	166	138	140	124
Hunting	79	48	54	61	65	63	59	58	55
Walking	63	40	56	61	92	59	80	73	69
Climbing	47	29	22	24	33	30	36	32	37
Skiing	10	21	20	17	23	24	21	13	5
Aircraft accident	13	13	11	17	16	19	9	9	7
Lost Children	11	11	15	14	21	27	30	17	16

Table 2.3: The subjects of NZ land SAR operations.

Type	1994/95	1995/96	1996/97	1997/98	1998/99
Private group	223(44%)	175(36%)	185(38)	169(40)	171(43%)
Solo	157(31%)	159(33%)	160(33%)	164(39%)	114(29%)
Recreational club	30(6%)	16(3%)	10(2%)	9(2%)	16(4%)
School group	26(5%)	14(3%)	18(4%)	11(3%)	10(3%)
Children	21 (4%)	27(5%)	30(6%)	17(4%)	16(4%)
Commercial group	21(4%)	26(5%)	12(2%)	11(3%)	9(2%)
Senior citizens	16(3%)	22(5%)	37(8%)	11(3%)	15(4%)
Youth group	13(3%)	12(2%)	9(2%)	6(1%)	6(1%)
Psychological disorder	–	–	–	15(4%)	14(3%)
Other	4(1%)	32(7%)	20(4%)	10(2%)	27(7%)
Total	511	483	481	423	398

The number of occasions in which satisfactory equipment was carried by the subjects

Table 2.4: The highest rating categories of injuries received by subjects of NZ land SAR operations.

Injury	1994/95	1995/96	1996/97	1997/98	1998/99
Fracture	54 (20%)	28 (11%)	25(12%)	39(17%)	47(24%)
Fatal	51 (19%)	61 (25%)	38(18%)	27(12%)	35(18%)
Sprain	27 (10%)	15 (6%)	18(8%)	26(12%)	18(9%)
Medical condition	25 (9%)	26 (10%)	18(8%)	18(8%)	16(8%)
Hypothermia	23 (8%)	18 (7%)	20(9%)	20(9%)	12(6%)
Leg	12 (4%)	20 (8%)	13(6%)	5(2%)	2(1%)
Multiple	–	12(5%)	18(8%)	32(14%)	21(11%)

was recorded[8] and is given in Table 2.5. The number of operations where the subject's intentions were known was also recorded and is given in Table 2.6.

Table 2.5: The status of equipment carried by subjects of NZ land SAR operations.

Equipment status	1994/95	1995/96	1996/97	1997/98	1998/99
Equipment adequate	570 (63%)	474 (65%)	275(56%)	255(61%)	223(56%)
Equipment not adequate	342(37%)	259(35%)	121(25%)	82(19%)	74(19%)
Equipment unknown	–	–	19(4%)	21(5%)	30(7%)
Not applicable	–	–	73(15%)	61(15%)	71(18%)

Table 2.6: The intention record left by subjects of NZ land SAR operations.

Intentions record	1994/95	1995/96	1996/97	1997/98	1998/99
Intentions record left	540 (59%)	454 (62%)	425(60%)	232(55%)	216(54%)
Intentions record not left	372(41%)	279(38%)	197(28%)	114(27%)	82(21%)
Unknown	–	–	23(3%)	20(5%)	31(8%)
Not applicable	–	–	60(9%)	57(13%)	69(17%)

The prevailing weather conditions at the time of each alert are recorded in Table 2.7 and the total number of operation hours used over all land SAR operations are given in Table 2.8.

[8]Additional categories of 'unknown' and 'not 'applicable' were introduced from the 1996/97 period.

2.3.3.2 Lost Person Behaviour — USA Statistics

LaValla and Stoffel [103] give the main factors which effect a person's behaviour when lost as their general state of health, experience, biological cycles and the physiological effects of the environment. They describe the general behaviour exhibited by categories of subjects, as revealed by search statistics. These categories include children of various age groups, the elderly, the mentally disabled and the emotionally depressed, with additional classifications by activity type *eg.,* hunters, trampers, climbers, berry pickers and photographers.

Analysis of USA statistics [103, 156] reveal some interesting generalizations which may be transferable to New Zealand conditions. In particular, the behaviour exhibited by different classifications of subjects is now outlined.

- Trampers generally rely on tracks and travel aids for navigation with 33% travelling at night, 10% travelling for more than 24 hours, 55% descending in elevation, 49% being found by a reconnaissance search, and 40% being well equipped. The majority are young with little experience.

- Hunters tend to move off-tracks, with greater concentration on their game rather than navigation. They may often move at night, and most are communicative and mobile. Of those lost, 45% find their own way out and 66% are found within 3 km.

- Climbers generally remain on or near designated tracks.

- Children aged 1 to 3 years old will wander aimlessly with no sense of direction, finding convenient places to sleep.

- Children aged 3 to 6 years old will try and find their way to a familiar place and may not respond to their name being called, due to a fear of strangers.

- Children aged 6 to 12 years old will often become confused in a strange environment, and may not respond to their name being called if they have run away or are afraid of getting into trouble.

- Mentally disabled people will often not respond to their name being called, and may hide or remain in one location for some days.

- The emotionally depressed will often be found within sight and sound of civilization or a prominent location, and may be unresponsive, preferring solitude.

- Skiers are generally communicative and mobile, 45% will travel at night with 50% finding their own way out and most are found within 8 km.

- Berry pickers, photographers and other hobbyists are often misled by subtle terrain changes and become disoriented as their attention is focused elsewhere. They are likely to wander, be communicative and carry no survival equipment or clothing; 90% are found within 8 km.

In general:

- most people are found within 8 km of where they were last sighted, and many within 3 km;
- approximately 90% of subjects do not keep moving for more than 24 hours; and
- the majority of lost people will travel in a downhill direction.

From experience, as a search advisor in New Zealand, Dittmer states that most people, unless injured, will not remain in one place but will keep moving for warmth [42].

Syrotuck [159] collected statistics from the Pacific Northwest but considers that the 'average' probabilities obtained could be applicable to other regions. He presents tables of POA distribution patterns from statistics gathered and analysed from 100 searches under varying circumstances. The statistics indicate strong trends and Syrotuck [159, 42] considers that they "can be used with a certain amount of confidence" despite the relatively small sample. The tables give POA information for general cases as well as for different categories of subjects. These are then further analysed over flat and mountainous terrain to give probabilities that the subject will remain at a similar elevation, or choose to move up or downhill, from the position that they were last seen or predicted to be at.

Syrotuck gives the following statistics for level terrain:

- the average probability that a person will be within 1.6 km of the point they were last seen at is 45%;
- the average probability that a person will be between 1.6 km and 3.2 km of the point they were last seen at is 37%;
- the average probability that a person will be between 3.2 km and 4.8 km of the point they were last seen at is 13%; and
- the average probability that a person will have travelled beyond 4.8 km of the point they were last seen at is 5%.

Syrotuck [159, page 18] states that "terrain basically influences the direction of travel and distance from the last seen point to where they are found."

In Table 2.9 Syrotuck [159] provides generalities from the statistics collated, which are unaffected by the terrain type.

Additional research into the behaviour of children lost in urban and suburban environments was conducted by Cornell and Heth [32], behavioural scientists. Cornell and Heth observed children aged 3 to 12 years walking from their home to "the farthest place" that they thought that they could walk to. Observations on the children's rate of walking, their chosen path and the total distance covered were recorded and analysed. Their analysis gives results as percentiles for differing age groups which could be used as probable performance estimates for travel by lost children and as a decision tool for search management. In particular their research gives information on time-based travel and dispersion which was not previously available.

2.3.3.3 Predicting Subject Position

Predicting the position of the subject is a way of attempting to increase the POA value and provides a strategic place to centre the search. Predicted positions are found based upon available statistics relevant to the subject, terrain and conditions. Searching is then conducted in both upward and downward directions from this position. Syrotuck [159] advocates searching from a predicted position rather than from the *Point Last Seen(PLS)* as this uses the power of statistics to place searchers into areas of higher probability. For example, statistics show that 28.94% of people are located within 1 km of the PLS, but 81.25% are found within 1 km of the predicted position [159]. Syrotuck cautions that, in cases where the subject is lost in flat terrain with no direction of travel indicated, it is preferable to search from the PLS. Swombow [158] uses the term *Initial Planning Point* to represent either the position where the subject was last seen, or their last known position that can be identified by a definitive clue.

2.3.4 Trail-Based Probability of Area

Colwell [27] gives the main disadvantage of the consensus approach to POA value determination as the subjective opinions involved, which could "simply be wrong or perhaps unconsciously biased". He considers that the utilization of statistics, from previous searches, in assigning POA values is more rigorous than the subjectivity of consensus decision-making but is only useful when relevant to the geography, subject and environmental conditions present. Colwell [27] states that:

> "Both consensus-based and statistically-based POA estimations have intrinsic advantages. In practice SAR planners will frequently tend to choose

Table 2.7: The weather conditions of NZ land SAR operations.

Weather	1994/95	1995/96	1996/97	1997/98	1998/99
Good	233 (45%)	104 (21%)	227(47%)	211(50%)	219(55%)
Fair	127 (25%)	248 (51%)	162(34%)	126(30%)	100(25%)
Poor	101 (20%)	80 (17%)	82(17%)	67(16%)	69(17%)
Extreme	39 (8%)	18 (4%)	10(2%)	19(4%)	10(3%)
Not specified	11 (2%)	36 (7%)	–	–	–

Table 2.8: The total operation hours of NZ land SAR operations.

	1994/95	1995/96	1996/97	1997/98	1998/99
Hours	4125.2	3051.5	2618.5	2666.8	3147.8

Table 2.9: Generalities for USA land SAR operations.

Attribute	Children	Trampers	Hunters	Elderly
Not concealed in good weather	75%	82%	83%	58%
Not concealed in bad weather	57%	60%	66%	10%
Move upwards	36%	8%	0%	0%
Stay at the same elevation	9%	4%	24%	25%
Move downwards	55%	88%	76%	75%
Survive (good weather)	95%	87%	91%	58%
Survive (bad weather)	29%	30%	50%	2%
Use a path or trail	71%	73%	59%	48%
Dressed brightly	44%	73%	37%	46%

whichever approach appears to have the greatest quantity of useful support information available for developing the POAs."

When data on comparable incidents is unavailable or insufficient, a new terrain-based method for developing POA values is proposed by Colwell [27] — that of trail-based probability of areas. This tool can be utilized in situations where the subject has, initially at least, followed a recognized path from a known origin to an expected destination. The assumption of the method is that this path was followed for some distance before the subject moved away from it. As many trampers, in particular, leave details of an intended path at the outset of their trip (a practice strongly encouraged by Department of Conservation (DOC) staff in national parks and in SAR education) this method is very applicable to the New Zealand search environment.

The method of trail-based POA determination begins during the initial search phase of a search (the reconnaissance search). During this search a team of searchers will follow the intended path of the subject, recording in detail any possible point on the track where there exists a likelihood that the subject may have been tempted to leave the track. Such points are termed *decision points* and may include such geographical features as apparent shortcuts, minor intersecting tracks, points where off-track attractions are visible and "enticing 'natural routes' ". The team estimates the relative probability that the subject may have left the track at each decision point. This is done using a descriptive scheme of 'very unlikely' to 'very likely'. The associated quantitative measures are detailed in Table 2.10.

Table 2.10: Probability decision data for determining trail-based POA values.

Probability description	Numerical value
Very likely	9.0
Likely	7.0
Even chance	5.0
Unlikely	3.0
Very unlikely	1.0

The searchers are able to define any numerical value falling within this range. Each team member's individual assessment is averaged over the team for consistency. Colwell notes that for additional consistency and accuracy it is preferable, if possible, that only one team assesses a track, and that they follow the track in the same direction as the subject and under similar conditions that the subject may have experienced. Collection of data on decision points can be done as training exercises prior to any search operation.

The search manager then partitions the track into numbered segments defining search areas by natural features. These areas are further partitioned into regions able to be searched within one day. (Alternatively search regions which have been defined for the area in pre-planning sessions can be utilized.) Within each search region the numerical probability ratings, assigned to each decision point falling within that region, are summed to give a relative POA value for each region. All POA values for search regions comprising the path, including the POA value assigned to the ROW region, are then normalized.

If more than one track could have been initially followed by the subject, or if the main track branches into definite side tracks, then a multiple track method can be applied. Essentially this method follows the same approach as for a single track but weights the POA values of each search region by a weighting factor[9] assigned to each track. The weighting factor represents the probability that the subject followed that track and the factors sum to one over all the possible tracks.

Colwell [27] states that the consideration of both the number and likelihood value of decision points along a track in determining the POA value of regions adjacent to the track, is the main advantage of the method. He also considers it an advantage that the method requires the searchers "to think much more like the subject". The method, however, is limited in that it is unable to provide much useful information on tracks which have only few low probability (or no) decision points, and where physically recognized main tracks are not present.

2.3.5 Composite POA method

Colwell [27] further proposes a method for obtaining POA values based on all information available. The approach weights the POA values determined by the trail-based method, the consensus approach and the statistical data available, to balance all available information. The degree of weighting used emphasizes the approach considered most relevant by the search manager for the current scenario faced. Colwell considers that this approach can balance theoretical and practical methods, and can be used to "reinforce and fine-tune" information from other techniques.

Stone [153, page 209] describes the process of arriving at a subject location distribution as:

> "...a blend of objective and quantified subjective inputs. One cannot say the result is right or wrong in the traditional scientific sense. However, the resulting distribution does provide the foundation for rational planning based

[9] Termed the scenario Probability Factor by Hill (in [27]).

> on the search planner's best understanding, both subjective and objective, of the search problem. Moreover, there is no good substitute for this combined subjective/objective approach. To leave out the subjective information is to throw away valuable information because there is no unique or 'scientific' way to quantify it."

All the approaches to POA estimation described here have some advantages. The consensus method, if utilizing experienced decision-makers and eliminating as many preconceived biases as possible, promotes discussion on all available information and provides a good approach to integrating a number of possibilities into probabilities. The multiple scenario analysis method promoted by Hill provides a simple way of ensuring that a number of scenarios are considered when allocating POAs in an approach that does not require decision-makers to implicitly incorporate more than one scenario into their POA calculations, as required by both the Mattson and Perkins techniques. The trail-based POA method of Colwell is novel in approach and attempts to arrive at more objective probabilities. Both this method, Perkins' consensus technique, and Hill's scenario analyser, all reduce the level of probability estimation to a simple weighting scheme to reduce the complexity of the decisions required yet still maintaining the relative importance of one search region to another.

However, all these approaches are enhanced when the information that decision-makers draw upon is supported by statistical information concluded from significant sample sizes. The statistical approach alone, however, is vulnerable to incorrect analysis and application if not carefully fitted to the local terrain and subject profile, and in any operation the subject may respond in an unpredictable way.

In conclusion, a balanced approach that involves consensus, statistical input, local experience, and the consideration of multiple scenarios in a well managed approach eliminating biases as much as possible, will provide a set of initial POA values which will withstand scrutiny and, if updated with changing information, should provide a sound basis for developing search strategy. It can only be an advantage to further incorporate research into subjective decision-making into this step of establishing the subject's initial location distribution.

2.4 Search Object Detection

The ***Probability of Detection (POD)*** is a conditional probability and relates to the likelihood that, if the search object is in the region searched, it will be detected by the search resource searching that region. It can be defined as "a measure of sensor performance in a particular search or, alternatively, a measure of how well an area has been searched", [150, page 2-2].

The POD of a lost person or clue, given that they are in a particular region, is affected by a number of factors. These include: varying terrain; weather; light; season; the fatigue level of the searchers; the size, position and colour of the object being searched for; and a subject's level of responsiveness. POD must also be determined for the type of searching resource, detection sense and search tactic being employed. A search pattern can simultaneously give varying POD values for differing search objects, *e.g.* a responsive or non-responsive person, and a clue left by them. These can be recorded as 'multiple POD levels' [22].

In general the value of the POD can be increased for a given search resource by searching a region using a more accurate pattern, *e.g.* by decreasing the spacing between ground searchers. Hence, the POD level attained will be influenced by the number of resources available for the search at hand.

Frost states [59]:

> "I think many people, at their first exposure to the idea of measuring detectability, overestimate the difficulty of obtaining reasonably accurate, objective approximations of detectability for a reasonably comprehensive set of environmental situations, search objects and sensors, and underestimate the accuracy and objectivity having and knowing how to use such detectability measures will bring to their POD estimates."

A distinction is drawn between predictive POD, the estimates prior to searching, and retrospective POD, the estimate by the search resource as to their actual performance level. However, due to the range of influencing factors and the mathematical complexity involved in estimating POD values, along with the confusion that appears to exist among SAR personnel in relation to POD as recognized by Hill [89], we agree with Perkins and Roberts that "to ask search group leaders to estimate a POD value retrospectively is fraught with danger" [138].

Goodman [77, page 60] promotes what he sees as the more meaningful use of recording search information for linear features, such as roads, in terms of "searched x number of times", in preference to explicit POD values.

Syrotuck [159, page 5] advocates the use of POD determination to avoid what he sees as a common fault, "to re-search an area over and over again not realizing to what extent it has been covered." He also warns against being too thorough and over-utilizing resources in one area to the detriment of other areas, without sufficient justification.

Cumulative POD can be calculated as

$$\text{POD}_{\text{cum}} = 1 - (1 - POD_1)(1 - POD_2)\ldots.(1 - POD_n)$$

where the POD_i values correspond to the independent searches over that region.

2.4.1 Lateral Range and Sweep Width

The *lateral range (x)* of a sensor is defined as "the perpendicular distance from the sensor's relative track to the object", [150, page 4-1], assuming straight line relative motion between the sensor and search object. The lateral range of a sensor is usually expressed as a signed quantity where a negative value indicates that the object is positioned to the left of the sensor. The curve which depicts the cumulative probability of the detection of the object by the sensor as a function of x, with one pass, is termed the *lateral range curve* of the sensor and is denoted as $p(x)$. Lateral range curves can be used as a means of classifying sensors.

The *sweep width (W)*[10] of a sensor is distinguished from the width of the swept area, and is defined as the area under the lateral range curve.

$$W = \int_{-\infty}^{+\infty} p(x)dx$$

Alternatively, W can be calculated as [63]:

$$W = \frac{\text{number objects detected per unit time}}{(\text{number of objects per unit area}) \times (\text{searcher speed})}$$

Frost [63] describes sweep width as "a basic, objective, quantitative measure of *detectability*" which is valid for visual searches where a resource's view is "relatively unobstructed ... or situations where obstructions are common, such as searching in or over forests".

The sweep width of a sensor depends on many, often interdependent factors, impacting on detection. These factors are described by Frost's *detection triangle* which he defines as having three sides: sensor characteristics, search object characteristics and environmental factors [60].

The detection factors that Frost considers are significant to visual searching are [59]:

- *sensor characteristics* — type, speed, night vision potential, light power, training and experience, navigational capability, time on the task and fatigue level.
- *search object characteristics* — size, colour and brightness contrast to the surroundings, potential for attention-getting movement, behaviour, signaling device and the ability to make fire, smoke, etc.

[10] Also termed the *effective search width* [100].

- *environmental factors* — visibility and weather, vegetation, terrain, season and lighting conditions.

In general, larger sweep widths are attributed to larger search objects, better weather conditions, easier terrain with less vegetation, and rested search resources [150]. The amount of 'dwell time' spent looking at a region and the positioning of an object, will also effect the recognition of that object (Frost citing a talk by Ken Hill on the visual detection process) [59].

2.4.2 POD Data for Search Resources

> "An important part of effective search and rescue (SAR) planning is the accurate prediction of the detection performance of SAR units. On the one hand, an underestimate of detection performance can result in an unnecessary extension of search in a particular area. On the other hand, an overestimate can produce the premature termination of search in a particular area. In both cases, the SAR resources are not being effectively used", [49, page 651].

While some field experiments have been conducted to estimate POD values under different operational conditions, and hence develop lateral range functions for different sensors, constructing accurate instantaneous detection functions "may not be practical" due to the large number of dependent factors [150]. Marine experiments by the U.S. Coast Guard Research and Development Centre, cited in [150], attempted to directly infer a lateral range curve under differing conditions as less data is needed for this process than to develop an instantaneous detection model.

Colwell [22] gives his opinion that "the only good POD is the 'measured' POD" — the POD values which are derived from field experiments conducted under the same conditions as those which can be experienced in the search area that the data is to be applied to. He accepts that in the 'real world' POD values are not always known; he therefore ranks POD reliability and credibility in the following order:

1. field-measured POD
2. adjusted POD — field-measurements are adjusted for the current conditions
3. team estimated POD
4. 'fudge' factor adjustments of POD estimates

Colwell [22] also considers that search teams cannot accurately estimate their own POD as they have "no frame of reference against which to compare". He is particularly sceptical of any 'fudge' factors utilized. He includes within this category the extrapolation from other factors such as the time taken to conduct the search or the 'visibility distance', when this is not applied to measured POD values. He states that the type of subject and terrain can not be "easily translated into simple 'fudge' factors".

Currently there are not many POD data sets available for different search resources searching for varying objects, in varying terrain under different environmental conditions. This is especially true for New Zealand conditions. Internationally the largest contributor of published land POD data sets has been Colwell, who has conducted field experiments for ground resources searching in Canadian dense coniferous forest and subalpine forest under differing environmental conditions and search patterns [20, 21, 25]. One of the earliest recorded field trials to determine POD values for ground searchers was conducted by Wartes in moderately dense underbrush, in 1974. These are given in [159] and reproduced in Table 2.11.

Table 2.11: Wartes' field trial POD data for ground searchers in moderately dense underbrush.

Spacing in feet (metres)	POD %
100 (30.48)	50
60 (18.29)	70
20 (6.10)	90

Syrotuck [159] derives the following formula from this data, using linear interpolation, to determine POD values for differing search resource spacings:

$$\text{POD \%} = 100 - (0.5 \times \text{searcher spacing in feet})$$

However, no correction factors for varying terrain density are utilized.

The above field trials led to Wartes reaching a conclusion which is now referred to as *Wartes Theorem*. That is,

> "repeated sweeps of the same area with wide spacing will be more efficient than a single sweep with close spacing", [103, page 116].

While this conclusion has long been adhered to by the land SAR community, Frost [62] states that "there is no theoretical basis for the claim that two successive low-coverage (*i.e.*, low POD) searches of a region will produce a higher cumulative POD for the same effort as a single higher-coverage (*i.e.* higher POD) search would. In fact, search theory suggests the opposite effect is far more likely." In particular, he states that two sweeps of a region at one half of the coverage of one sweep can produce results no better than the single sweep and may in fact produce poorer results due to the difficulty of executing two independent parallel searches [65].

For air-based search resources Syrotuck [159] gives POD data for helicopters as varying over the range of 20–80%; 80% given for open, flat terrain in good weather, with a mobile victim wearing bright clothing. While Tristram [165] indicates that the POD from a helicopter is very dependent upon the terrain and can range from a possible 90% on a ski slope in good conditions, to an approximate 5% in dense bush. In practice POD will also vary with respect to environmental conditions, the subject's position and responsiveness, and the observers' training and level of fatigue. Keane [95] cites POD data for air searchers for lost persons that he received from the USA AFRCC.[11] This information is tabulated in Table 2.12.

Table 2.12: USA AFRCC POD data for air searches of lost people.

Terrain type	POD
Flat terrain with limited vegetation	49%
Hilly terrain with moderate vegetation	29%
Mountainous terrain with heavy vegetation	19%

Keane considers this data to be consistent with searches by helicopter, from his own experience, but not with searches from fixed-wing craft. He is unaware of POD data available for air searchers which utilize infrared equipment.

Syrotuck [159] allocates a POD rating to search dogs of 70%.

2.4.3 Critical Separation

Perkins [137] addresses the trade-off between POD values and searcher spacing by introducing the concept of *critical separation*, developed out of a general theory which links POD to the spacing between searchers in the field. This model is considered by Gordon [80] to be more appropriate to the terrain of the United Kingdom (where it was developed) where larger areas of the same terrain type are found, than for the usual New Zealand terrain which varies considerably over small areas.

Critical separation is defined as the distance between two searchers such that an "object placed midway between them would be on the limit of visibility for both of them", [137, page 15], *i.e.*, "their visibility horizons coincide", [137, page 16]. Perkins describes this measure as *elastic* — being a dynamic parameter reflecting differing terrain and search conditions.

[11] Air Force Rescue Coordination Center.

Roberts [137], Team Leader for the Northumberland National Park Rescue Team, outlines the practical determination of critical separation which is carried out by the search teams who will be implementing it, in the area they will be searching. The process involves the team locating an area representative of their search region, then placing an object on the ground, of similar size and colour as the search object. Two searchers then circle this object determining the limit of their visibility. The distance between the searchers is then taken as the critical separation for searching that area under the present conditions. If these conditions alter, then the distance is re-determined. This process can be done without introducing the specific terminology involved and can also be conducted at night with torches. It should also be conducted with consideration to the direction that the team will be searching in when unusual terrain features exist.

It is assumed that a searcher's visibility will decrease linearly over their visibility range, having a value of 100% POD if the search object is on the searcher's path, and a value of 0% POD if the object is just past their limit of visibility. Perkins recognizes that this graph is not a "strict representation" of reality but uses it as a theoretical base model. Theoretically, spacing searchers conducting a parallel sweep search of a region at critical separation will result in a POD of 50%.

Perkins extends the results to give POD values for given searcher spacings, where spacings are defined as non-integer units of critical separation:

$$\begin{array}{ll} \frac{50}{N} & N \geq 1 \\ 50(2-N) & N < 1 \end{array}$$

where N = the number, or fraction, of critical separations, and constant spacings are assumed. These POD values are applicable to any terrain and conditions and agree with Wartes' published field data for moderately dense underbrush (Table 2.11), which is widely accepted, giving greater credibility to the theory.

2.4.3.1 Purposeful Wandering

Perkins and Roberts [137] emphasize the ease and practicalities of the critical separation approach, along with its increased effectiveness. An effectiveness which has shown by experience to be further increased by *purposeful wandering*, where the "searcher wanders about as he moves through the terrain rather than attempt to maintain a constant spacing", [137, page 9]. The searcher stops regularly to look around, particularly in difficult terrain and to search possible places of concealment. As Perkins [137, page 24] states:

> "searching is a positive activity; it entails much more than just walking forwards along an imaginary line — a searcher must move around within his strip

of ground to look in likely hiding places and to gain good vantage points, and must stop at frequent intervals to look back."

Field observations by Perkins and Roberts indicate that employing this tactic with experienced searchers increases the POD when using critical separation from a theoretical 50% to approximately 75–80%. This measures to some extent the effect and importance of purposeful wandering.

Another method which is assigned the name of purposeful wandering is a method which was adapted in Germany during the Second World War. In this instance purposeful wandering involved sending additional people into the search area to wander through the area following their gut instincts and known information on the lost person [81].

2.5 Success of a Search Operation

The ***Probability of Success (POS)*** of a search in a region i, *i.e.* the probability of the detection of the subject in region i, is found by the multiplication of POA and POD values.

$$POS_i = P(\text{subject in region } i \textbf{ and } \text{detected}) = POA_i \times POD_i$$

POS is used as a predictive measure of how effective one choice of search resource and search technique will be in comparison to another. As POS values for each search region are independent, they can be summed to give a total POS measure for a given phase in the SAR operation. POS is a useful planning tool in determining the overall search strategy. However, as it depends upon POA values which are subjective; "any notion of obtaining 'an exact solution' is unreasonable", [136, page 51].

Cumulative POS (POS_{cum}), measuring the likelihood of detection based on all searching on the operation up until the current time, can be calculated by three methods [150]:

1. POScum $= \sum_i^{all\ regions} POA_i \times POD_{cum,i}$
   where $POD_{cum,i}$ represents the POD_{cum} value for search region i
2. POScum $= \sum_i^{all\ searches} POS_i$
3. POScum $= 1 - \sum_i^{all\ regions} POA_i$

The first method uses the initial POA values assigned to the search regions. The method is not recommended as the assumptions underlying the formula are not usually valid in reality [150] (those assumptions are that the only influence on the subject's location probability distribution are the actual searches). Both the second and the third method of calculation are valid under any search situation provided that the POA values of each search region are adjusted to account for non-detection on the completion of each search. The methods also assume that POA values are not normalized when updated after each search.

The calculation of the POS_{cum} can be used to monitor whether or not search effort is being allocated to the correct area. A high value, indicating a continued lack of success, could indicate that the wrong region is being searched [150] and that the current scenario being followed is incorrect [62]. However, Stone [153] cautions that the definition of 'high' is subjective. POS_{cum} can also be used as one decision factor when search suspension is under consideration.

2.6 Location Probability Distribution Update

"Most searches take place sequentially, or over a long enough period of time, so that the search planner can receive feedback from the search and adjust his plan accordingly", [153, page 227]. The subject location probability distribution, and hence POA values, can be updated to account for all of the information gained during searching up to a given point in time. If a region is searched without subject detection then the probability that the subject is in another region may be increased, taking into consideration the level at which the region was searched, *i.e.,* the possibility that the subject may have been missed. As more regions are searched the current POA value of previously searched regions may be high enough to warrant those regions being re-searched. These probability calculations can be time consuming (although much less so with advances in computing power), but Syrotuck [159] emphasizes their utility as an aid to monitoring the progress of the operation and the likely position of the subject.

The subject's prior probability distribution is replaced by a posteriori probability distribution by a Bayesian update of the POA values. Bayes' theorem for mutually exclusive events $B_1, B_2, ..., B_k$, of which one must occur, is given on page 150 of [55] as:

$$P(B_i|A) = \frac{P(B_i) \cdot P(A|B_i)}{P(B_1) \cdot P(A|B_1) + P(B_2) \cdot P(A|B_2) + ... + P(B_k) \cdot P(A|B_k)}$$

which is equivalent to:

$$P(B_i|A) = \frac{P(B_i) \cdot P(A|B_i)}{P(A)}$$

If we describe the event B_i as the event that the subject is located in region i, we can see that the B_i events for $i = 1, 2...$ or k, for k regions, are indeed mutually exclusive; the subject cannot be located in more than one region at any given time. If the ROW region is included, say as event B_k, then it is also true that one of the B_i events must occur. Hence, it is valid to apply Bayes' theorem to calculate the probability $P(B_i|A)$, namely the probability that the subject is located in region i, given that region i has been searched and the subject was not detected. This is equivalent to finding the updated POA value for region i, POA_i^*.

Decomposing the right hand side of the equation we can see that the probabilities can be individually expressed as:

$P(A)$ the probability that region i was searched without detection, which is equivalent to the probability that the subject is in region i and was missed or that the subject is not in region i $= POA_i \cdot (1 - POD_i) + (1 - POA_i)$

$P(B_i)$ the probability that the subject is located in region i $= POA_i$

$P(A|B_i)$ the probability that region i was searched without detection given that the subject is located in region i $= (1 - POD_i)$

Therefore,

$$POA_i^* = \frac{POA_i \cdot (1 - POD_i)}{POA_i \cdot (1 - POD_i) + (1 - POA_i)}$$

This equation can be simplified to:

$$POA_i^* = \frac{POA_i \cdot (1 - POD_i)}{1 - POA_i POD_i}$$

To ensure that all subject location probabilities sum to one at any time the POA values of the other search regions can be normalized. For such a region j the updated POA value is then:

$$POA_j^* = \frac{POA_j \cdot (1 - POA_i^*)}{1 - POA_i}$$

Substituting the simplified expression for POA_i^* into this equation yields:

$$POA_j^* = \frac{POA_j}{1 - POA_i POD_i}$$

Clark [19] refers to the term (1-$POA_i POD_i$) as the probability of "non-success" as this is equivalent to $(1 - POS_i)$.

However, Cooper [30] states that the most effective and undisputed way to adjust POA values upon the receipt of substantial new information is to obtain a new consensus, this approach being "superior and simpler to alternative methods of either subjectively or mathematically adjusting POA in similar situations."

2.7 Search Measurements

Hill [88] states that "using resources efficiently means getting the most coverage from the resource for the number of man hours expended, and, ultimately, finding the lost person sooner rather that later."

A number of quantitative measurements exist to monitor the effectiveness of the search operation. These are largely contained in the classical search theory literature [100, 150] and consist of the following.

Effective search or sweep rate is calculated as:

$$\text{Effective Search Rate} = W \times searchspeed$$

The area effectively swept is then defined as:

$$\text{Area Effectively Swept} = \text{Effective Search Rate} \times t$$

where t represents the time spent in the search region.

Search effort, Z, is defined in search theory as the area which can be effectively swept [150] and is calculated as:

$$Z = W \times L$$

where L represents the distance travelled by the sensor in the search area. Or, alternatively, it may be defined as:

$$Z = W \times V \times T$$

where V is the search speed and T the hours spent in the search area. This definition of effort is different to the searcher-hour definition commonly used by the land SAR community.

The *coverage, C,*[12] of a search area is determined as a ratio of the search effort and the size of the search area.

$$C = \frac{Z}{A}$$

For straight, equally spaced, parallel sweep searches, the coverage can equivalently be defined as:

$$C = \frac{W}{S}$$

where S denotes the track spacing. Where k equivalent search resources are searching simultaneously in one area, C is calculated as [65]:

$$C = \frac{Z \times k}{A}$$

Overall search effectiveness is measured by the cumulative POS and search efficiency is measured by the rate of POS growth over the search operation [62].

[12] Also referred to as *effort density* [152].

2.7.1 Searcher Utilization

Perkins [137] gives three parameters by which to measure searcher utilization for ground searchers conducting a visual sweep search: POD value; the number of searchers; and the width of the search corridor (given in standardized units of critical separation). He places these in the following formula,

$$\text{utilization} = (\text{corridor width} \times \text{POD}) \;/\; \text{number of searchers},$$

where the search corridor is the total width covered by the searchers spaced in a line, from end searcher to end searcher, and also includes the outer distance searched by the end searchers. The corridor width is calculated as $(M-1)N+1$ critical separations if $N > 1$ or as $2N+1$ if $N \leq 1$, where M represents the number of searchers and N equals the spacing in critical separations.

Perkins investigates differing values for the search parameters, finding that spacing searchers at critical separation gives the best searcher utilization for larger search teams. Perkins claims that, theoretically, critical separation between searchers results in the best utilization of searchers, as the entire area between any two searchers is covered and no area is covered by more than one searcher. Any other search spacing "will result in either a lower POD or a lower utilization", [137, page 23]. Particularly, the use of a wider spacing will produce no gains but higher priority areas may require a closer spacing to increase the POD. However, Frost [62] cautions against jumping to the conclusion "that the overlapping of detection profiles from adjacent searcher tracks is to be avoided under all circumstances ... with realistic detection profiles, some overlap is often required to achieve a practical approximation to the optimal search plan."

Perkins also identifies an effect on searcher utilization for smaller teams which he terms the *'end man effect'*. This effect identifies that the search terrain covered at either edge of the team's search corridor by the outside searchers, as a proportion of the entire corridor, is larger for smaller teams. In this instance the searcher utilization is a maximum when searchers are spaced at 0.7 critical separations.

These concepts, predictive statistics and search measures comprise the fundamentals of effective search management. We now review how these concepts are currently incorporated in the procedures followed by land SAR managers, examining the decisions required in initiating a SAR response, allocating search resources and executing specific search techniques.

Chapter 3

Current Search Methods

"No hard and fast criteria can be laid down for the functions, tasks, staffing, accommodation or equipment. Each operation will build from a nucleus as the factors dictate."

— Lee [121, section 106]

In particular, when organizing a SAR operation, the following must be considered:

- type of incident and available information;
- level of urgency and subsequent priorities;
- weather and climatic conditions, and weather forecast;
- terrain — including access, and any features which require specialist skills or equipment *eg.*, caves, rivers, swamps or cliffs;
- available search time and search resources;
- the definition of a primary search area and the segmentation of this into search regions;
- the allocation of search resources and, in particular, the type of search techniques to utilize;
- search base location, transport, food and equipment; and
- communications.

We now examine the details of these search considerations.

3.1 New Zealand SAR Organization

In New Zealand, SAR is divided into three classes, dependent upon the seriousness and scale of the search operation. *Class One* searches are Police controlled and employ only Police resources. *Class Two* searches are managed as a combined effort between the Police and civilians. *Class Three* searches cover wide areas and usually involve missing craft such as boats and planes. This class of search is coordinated by the centralized National Rescue Co-ordination Centre, and involves the military working alongside civilians.

The New Zealand Search and Rescue organization was officially set up in 1948 [99]. Currently land SAR in New Zealand is coordinated by the New Zealand Land Search and Rescue network. This network is comprised of police, Department of Conservation (DOC) staff, and mountain and caving experts [115]. Jack McConchie [50, page iii], Convenor of the New Zealand Land SAR Committee, reports that:

> "Search and Rescue in New Zealand is internationally unique in that it operates entirely in the absence of full-time paid professionals involved solely in SAR. The fact is that the population and financial base are too small to support a 'professional' system, and a 'centralized unit' could not effectively cover the wide area and diverse terrain. The real strength of the 'New Zealand system' is its ability to call, from a wide range of skills in the community, those that are needed for a particular operation. It is however not a club or organization, but a collection of individuals who apply their skills *"in the best interest of the victim."* "

However, Martin [115, section C1] reports on "a fragmented rescue system heavily reliant on volunteers" with only the Police and Defence Departments, and the Civil Aviation Authority receiving government funding. This problem was highlighted in January 1997 when DOC, which had previously received government funding, had this source of finance removed and consequently removed their SAR services. (Although DOC staff may still participate in SAR operations in a volunteer capacity.) Statistics gathered from New Zealand SAR operations [122] indicate the high utilization of volunteer search resources and the low proportion of paid reimbursement. This is evident from Table 3.1.

Financial grants are given to community-based SAR teams by the police, but as these are non-representative of the full costs of an operation, these teams struggle. This is highlighted by Martin, where the rescue of a lost and sick tramper in one region cost \$13000, \$3000 more than their Police grant for a full year. When there is a chance that a lost person may still be alive, however, no real monetary restrictions exist in attempting to rescue them in time [165].

Table 3.1: The utilization of volunteers in NZ land SAR operations.

	1994/95	1995/96	1996/97	1997/98	1998/99
Number	2858(64%)	3954(68%)	3139(65%)	2829(68%)	2108(63%)
Hours	21669(53%)	22011(69%)	15812(63%)	20147(66%)	14368(54%)
Payment	$38 363(2%)	$8 388(< 1%)	$1 748(< 1%)	$2 921(< 1%)	$40(< 1%)

3.2 Urgency

SAR differs from some other operations in being "time critical", with the strength of the response being related to the perceived urgency of the situation. A statement on urgency is somewhat subjective and LaValla and Stoffel [103, page 13] consider it often difficult to justify "because a certain percentage of lost people, if left on their own, would survive and walk out." Marin County Sheriff's Department SAR team [114] point out that "time and weather may well make any search an emergency as they destroy clues."

Swombow [158, page 28] expresses the need for "documented analysis of the determination of search urgency which will stand up to later scrutiny." Meeting this need is a system devised by Wade [86] which assigns a quantitative value to measure relative urgency. This is achieved by working through an urgency chart which assigns specific values to such attributes as the subject's age, medical condition, experience, equipment carried, number in the group, terrain hazards, area incident history, 'bastard search',[1] and weather conditions. These values are then added together and different levels of response are actioned depending on the sum attained. This method provides a defendable approach, which clarifies a decision that can otherwise be influenced by external factors, especially when conducted under pressure. It has been trialled in New Zealand and found to be particularly helpful in deciding upon initial actions [42].

3.2.1 Initial Response

The initial response calls upon available local search resources as the situation requires. In New Zealand, lists of available personnel are maintained with their fitness, skill and leadership rating, and contact number [165]. Volunteers can often be contacted at work, resulting in a good response time, although it is difficult to initiate searches during commuter hours in the larger cities. A shortage of resources is not usually encountered as resources from other centres can be called in. Currently there is a move towards having smaller, highly trained teams with resourcing from nearby centres, if necessary.

[1] The occasion when a subject has returned home without informing anyone.

Additionally there exists the objective that, by the year 2000, all searchers within New Zealand will be trained in clue awareness and be of similar ability and skill [81].

3.3 Definition of the Search Area

An initial search area, or possibility area [150], is found by deciding upon the potential maximum distance that the subject may have covered. The distance covered is calculated with respect to the length of time which the subject has been missing, and their estimated search speed as impacted by the terrain, weather and light [155]. This decision involves a number of other considerations including lost subject behaviour patterns, natural terrain barriers, any clues that may have been found, the search history of the region, significant features such as sources of water or shelter, and a profile of the subject that includes any physical and mental limitations. The decision will involve a certain amount of 'gut feeling' and deductive reasoning as a management team, and will lead into discussion on possible POA values. The search area should also include the subject's home, in order to search for clues and to ensure that they do not return there unknown to searchers. The search area is then divided into search regions for physical searching. Particular division techniques are discussed in detail in Chapter 8.

LaValla and Stoffel [103] state that the chance of success of the search is directly related to the size of the search area — a smaller area being significantly easier to search. To aid in this, confinement tactics are used — although this may not be possible in every instance. Such tactics establish a search perimeter by using road or trail blocks, track traps,[2] lines of string with attached arrows indicating the direction of help, and personnel positioned at observation points or camping at particular spots within the search area, such as streams. Decisions must necessarily be made as to whether resources would be better used in constricting the search area or in actively searching.

If the search area is large and not enough information exists on the possible locations or intentions of the subject, then a search will not be mounted until more information becomes available. Police will alert the public to the disappearance and anybody going into that particular area will be asked to keep an eye open for the subject, or signs of them [165].

Keane [96] emphasizes the need to base search planning on more than just information but also on search intelligence. He states that what is required is a

> "method to acquire, compile, verify, analyse and disseminate information about the subject, the events surrounding the search and the natural factors present in the search area."

[2]Pieces of cleared ground which are regularly checked for clues to the subject's movements.

Misinformation and disinformation should be avoided or detected if at all possible. This involves evaluating both the sources and information received.

Information which is identified and processed in planning the search operation includes [103]:

- subject profile — age, physical and mental condition, experience, personality, equipment and clothing worn;
- position where the subject was last seen (PLS);
- circumstances triggering the alert, *eg.,* amount of time over-due;
- alternative trip plans;
- weather;
- terrain analysis *eg.,* barriers, confusion factors, vicinity to civilization, possible shortcuts and hazards; and
- available resources.

Much of this information gathering requires an initial interview with friends and family of the subject, and any eyewitnesses. Such investigation continues throughout the entirety of the operation until the subject is found or the operation is suspended.

Interactive subject profiles need to be considered when more that one lost person are involved. The person with the dominant personality is likely to affect the others' decisions and possible actions. Swombow [157, page 67] introduces the notion of a changing subject profile when it is realized that "another person is influencing or controlling the movements of the missing person", as would be seen in an abduction case. Here the subject profile of the abductor is the key factor for search planning rather than the subject profile of the victim.

3.4 Search Base Location

The search operation is coordinated from a central base, and standby personnel are both based and deployed from there. The location of the search base should be chosen with consideration to the terrain, accessibility, communication transmission, distance to the search area and the closest water supply.

Gallas [66] states that a permanent building for a search base location is preferable, if available, and that use of Police or Armed Forces command vehicles may serve well as part of the search base in the field. Gallas states that sub-bases[3] may be a necessity when searching a topographically divided region. He gives as an example a mountain range which is simultaneously searched from both sides.

3.5 Search Resources

A search requires "cooperation and coordination among diverse multi-skilled responders" [103] with a commonality of purpose. Each type of initial response team has its advantages and should be applied to the areas that best suit its expertise. Before assigning resources to the field, their capability and training must be assessed to ensure that they are safely and efficiently matched to their tasks, and that those tasks are well defined. Search resources are pulled out of the search area if adverse conditions arise.

Syrotuck [159] states that there is often a need for compromise due to fewer resource numbers available than what would be optimal. Such compromise usually impacts the size of the search area, POD, time and quality of coverage.

3.5.1 Search and Rescue Personnel

Within the New Zealand Police Department are police personnel who are Police SAR Squad members. These members are called to assist volunteers in SAR operations which arise in their district. The person coordinating the New Zealand SAR operation at the search area, is titled the Field Controller. The Field Controller manages the physical SAR operation with advice and assistance from a local civilian expert, a SAR advisor. They are responsible for [50]:

- defining search objectives and urgency;
- defining the search area and the search base location;
- prioritizing the search area into regions for searching, and the search method to be employed;
- defining the searcher resources required, both in number and skill requirements;
- determining search backup and supplies;
- coordinating search tasks of the search teams with constant reassessment of the situation;
- determining communication schedules for teams;

[3] Sub-bases, alternatively termed spike camps or semi-fixed bases [103], are smaller bases designed to service a portion of search personnel located in a set area.

- communication with the Police Search Controller;
- recommending search suspension; and
- conducting debriefing at the search conclusion.

In the New Zealand environment searchers are initially fairly independent in their decision-making; given an area to search, they can make judgment calls as to how they will search it. This independence moves to a greater dependence upon the search manager as the search progresses [165].

Syrotuck [159, page 5] states that of all search resources, people are the ones that are

> "least affected by external factors ... they can search in most weather conditions and terrain and are not influenced by the colour of the victim's clothes. Furthermore, they are the most available resource across the country."

Searchers should have a high degree of fitness and competence in the bush with Sweere [156] extending this to the requirement that

> "All searchers entering the search area must be prepared to survive on their own, search for the lost subject, navigate over varied terrain, communicate with other searchers, direct assistance to (their) position, and assist in first aid and extraction of the subject."

Perkins and Roberts [137] prefer a basic search unit comprising three searchers, over larger teams previously used. They consider that a team of three is the most easily managed, and allows for easier movement and communication between the searchers. Teams appear to work more efficiently due to psychological factors such as a greater feeling of personal responsibility in their searching. Perkins and Roberts reduce this number to two searchers when this is deemed more appropriate for the terrain being searched *eg.*, when searching either side of a ridge. One searcher, alone, can be assigned to search a tributary if other searchers are close by to ensure their safety. New Zealand search teams comprise four members in most instances [165] as this is practical when using four person tents when searching over several days. One is selected as the leader and usually one team member will be a Police SAR squad member trained in disaster victim identification. This is to meet legal requirements in reporting to a coroner if a fatality is encountered (New Zealand National Search and Rescue Course, April 1996). However, this number will be dependent upon the stage and nature of the search.

The leader of a search team is responsible for coordinating the team's activities, and monitoring the mental and physical condition of the members of his team. The time limit of usefulness of a search team in the field will depend upon the nature of the search

— generally this will be four days or less. In particular, experience has shown that more than six hours of continuous searching will give rise to fatigue induced errors [121]. Gallas [66] describes this performance factor as each team having an "operational ceiling." If the operation is prolonged, reserves will be put into the area to replace 'forward teams'.

Perkins and Roberts [138] state: "it is our contention that the search manager delegates responsibility for the operational control of the search plan to the search group leaders, and they manage the search plan." Hence, training of these leaders is key to the success of a search operation.

Jones [93, page 68] states that:

> "An aspect that separates leadership requirements in Search and Rescue as opposed to that required in other outdoor activities is that it is largely applied in crisis under pressures of all sorts and under considerable time constraints."

Due to the pressures faced, maintaining the morale of the search resources is an important factor. Hill [88] gives the following as a selection of factors which could detrimentally effect searcher morale:

- insufficient briefing;
- "not being properly utilized in the field";
- not having their needs considered; and
- being assigned tasks which are lengthy or unrealistically difficult.

3.5.2 Aircraft

In New Zealand searches there is a high utilization of helicopters. Due to their ability to hover and land in quite restricted areas, they can be specifically utilized in a number of roles. In particular, helicopters are used to collect and transport teams or team leaders, to search areas or the base; perform reconnaissance searches; deliver equipment and supplies; arrange the siting of temporary communication equipment; recover victims; or lift out any evidence found. Often a helicopter will be used for body retrieval as this is less distressing for searchers. Commercial helicopters are used along with Air Force Iroquois helicopters. Iroquois helicopters are able to carry three crew and nine other people [99], but their pilots are known to be more reluctant to fly into regions or conditions which may be hazardous (New Zealand National Search and Rescue Course, April 1996).

Jennex [92] states that "aircraft are best used when the missing subject wants to be found." Colwell [22] stresses that "air support is fast and efficient — and may be far more cost-effective than many days of ground searching" — but suggests that the patterns used in air searches could be re-examined in terms of efficiency, particularly in light of the significant costs involved.

Helicopters and fixed-wing craft are most useful in searching less dense areas and can be utilized to fly up and down main tracks to indicate their location to the subject. If the conditions are cloudy helicopters are preferable to fixed wing craft due to their hovering ability [165]. Aircraft that are fitted with a Global Positioning System also have the capacity for the direct input of longitude and latitude coordinates, for map references. The maximum flight time of a helicopter is approximately two hours unless loaded with extra fuel and the slowest flying speed of a fixed-wing aircraft is approximately 80 knots (New Zealand National Search and Rescue Course, April 1996).

Factors which limit the use of aircraft, both rotary and fixed wing include [50]:

- strong wind;
- air temperature and/or pressure;
- altitude;
- poor visibility *eg.*, low cloud;
- fuel capacity;
- thick bush canopy;
- aircraft weight capacity; and
- size, position, and terrain or other hazards, at the landing site.

Obviously if the size of the object being searched for can not be easily detected from the air, search by air is impractical.

3.5.3 Search and Tracking Dogs

Syrotuck [159] advocates the effectiveness of search (or air-scenting) dogs, especially in dense terrain, as they can cover large areas more quickly than people. Search dogs can search one square mile (1.6 km sq) in four hours, but they are not generally placed into hazardous regions or adverse conditions. The dogs run ahead of their handler using air currents to locate a subject, hence they are not affected by time or weather and can be effectively used after several days, and after rain or snow. Such air-scenting dogs differ from tracking dogs (who follow a person's footsteps [156]) and are able to be used after other people have been in the area. Dog teams are generally given more freedom of movement than other search teams.

Currently in New Zealand, trained search dogs are a scarce commodity except in alpine snow conditions where teams are available. Currently both the dog and its owner must pass tests in the field to be listed and used. Their usefulness in a bush setting is a subject of debate among search management and personnel. There exists a limited use of police dogs in locating subjects along clearly marked tracks.

3.6 Communication

New Zealand SAR operations utilize the resources of the Amateur Radio Emergency Corps (AREC) to handle communications from the search base. Radio contact is required from the search base to search teams, sub-bases, town base, road-end base, and any aircraft. In the field each search team will carry portable radio equipment. New Zealand SAR teams will primarily carry High Frequency (HF) radios but will also use Very High Frequency (VHF) and Ultra High Frequency (UHF) radios. VHF radios utilize dedicated SAR frequencies [50].

HF radios have the advantage that they can cover a range of thousands of kilometres at a low cost. Due to the ionospheric reflection used, HF radios give very good coverage over mountainous regions, though they do not always work well if searchers are close together. HF radios utilize separate channels for day and night communication, transmitting at 3MHz during night hours and 5MHz during the daytime.

One of the main disadvantages of these radios is that large aerials are required (with nighttime aerials twice as long as daytime aerials, and aerials up to 45m in length), which should be completely unravelled for optimum use and strung up high, straight and parallel to the receiver's aerial, *i.e.*, perpendicular to the direction of transmission. Aerials should also be strung away from hut rooves or wet bush, which absorb the radio waves resulting in transmission problems. For this reason, dew first thing in the morning can also lead to transmission difficulties. Due to these requirements, use of HF radios requires more training than other types and larger clearings to operate in. They can also suffer from fading, and can be subject to overseas interference and solar activity. HF radios' batteries have an approximate life of four days.

In practice, however, the use of the aerials with HF radios does not necessarily have to follow the guidelines set out for successful transmission or reception. Aerials can be used in more confined spaces, damp conditions and strung over hut rooves, and communication can still be achieved (New Zealand National Search and Rescue Course, April 1996). Such preparation can be done very quickly if the need arises.

VHF and UHF radios have advantages and disadvantages over the HF radios. They are not affected by overseas interference, solar activity or fading, and are easy to operate with a small aerial (20cm). VHF and UHF radios are smaller and easier to carry. They can also be used for ground to air communications, while HF radios can only be used in this capacity if the aircraft is especially equipped. However, the radios have only a short range cover (line of sight) and transmission is prevented by obstacles such as ridges, unless repeaters are used. Because of these short range capabilities they are ideal for communication in contact searches.

HF radios use simplex operation — all searchers use the same frequency for both sending and receiving. This means that every searcher can hear one another directly and are able to relay transmissions to another searcher who is experiencing reception difficulties. As repeaters cannot use simplex operation, one of the main problems of their use is that unless another channel is available a station out of range of a repeater cannot communicate, even with searchers close by.

3.6.1 Reporting Schedules

Reporting schedules between the search teams and the search base may follow either a fixed or flexible schedule. A *fixed reporting schedule* involves search teams reporting at fixed intervals. The main advantage of this schedule is that radio time is managed efficiently and operations from the base can follow an organized routine. A *flexible reporting schedule* is generally more convenient for the searchers in the field who report at regular intervals during the day, but at times practical to them. This system can result in communication bottlenecks occurring at the search base and it can take some time to amend the tasks of teams currently searching. Fixed and flexible schedules may be used at differing stages of the search [66].

When radio communication is not possible other forms of communication can be utilized instead. These may include: cell phones, loud hailers, whistles, rifle shots, flares, smoke, or mirrors [50]. In addition a recall signal of four evenly spaced signals repeated after five minute intervals can be employed, with the signal used to communicate distress being three evenly spaced signals repeated after at least one minute [50].

3.6.2 Briefing and De-briefing

Everyone involved in the search should be briefed – "failure to do this will compound itself in terms of deficiencies and gaps in the search plan implementation", [103, page 118]. In particular the briefing of team leaders should include, [103]:

- a description of the current situation along with information on the subject and any concerns;
- the objectives of the search and specific search strategies;

- information on the terrain, any existing hazards, and the predicted weather;
- required equipment, communication and reporting procedures; and
- transportation and tactical assignments, including explicit instructions on search configurations, region boundaries, adjacent teams, *etc.*

Perkins and Roberts [137], unlike other search managers, do not inform their search teams of the priority of the search region assigned to them. They reason that this reduces the possibility of the region being 'over' or 'under searched'.

De-briefing is used primarily as a means of evaluating the components of the operation as a foundation for future planning. As search resources complete their assignments they are questioned as to their actual coverage; any difficulties or hazards encountered; any clues found; their estimated POD (by some search managers); and their recommendations for future efforts [103].

3.7 Resource Allocation

With a given search effort "there are infinitely many possibilities, assuming both the search effort and search object location probability distribution are infinitely divisible (a luxury not available to search planners in the real world but nevertheless quite useful to theorists). The question optimal search theory tries to answer is, 'Which combination of area and coverage will produce the highest POS?'", [150, page 6-1].

Determining just how to allocate a limited number of search resources to the field throughout the operational periods of a search is the crux of search management. Not only is it necessary to determine which resources to employ, but also which sequence of search regions to assign these to and by which search pattern. Search strategies must be developed and then adjusted to respond to changing information and search conditions.

The term *uniformly optimal* was introduced by Arkin [152] and describes a search plan which dominates another search plan for every time period such that:

$POS_t^* \geq POS_t \quad \forall t \geq 0$ where POS_t^* is the optimal POS_t.

A uniformly optimal search plan is described by Frost [62] as one which "increases POS at the maximum possible rate for the effort that's available." He states that the next most desirable plan is a *T-optimal search plan* which attains the same final POS for effort as a uniformly optimal search plan over time T, where T is larger or the initial POS growth rate is less than for the uniformly optimal plan.

Koopman [100] distinguishes between a *single-try search* and an *iterated or progressive search.* One search only, committing the total available effort, is executed in the instance of a single-try search, whilst effort is allocated over time in an iterated search with revision of the operation over this time. A single-try search is generally required when there exists only one chance at a successful conclusion to the search, *eg.*, the case where an undetected subject would perish before another search could be executed.

Stone [152] shows that for iterated searches for a stationary subject employing a regular detection function, an optimal search effort allocation in each increment of the search will result in an optimal search plan when the posterior subject location distribution is utilized in each increment. Frost [150, page 6-10] refers to this result as the *Additive Principal of Optimal Search* and states it as follows:

> "Given an amount of search effort Z, the maximum cumulative probability of success may be obtained by either a single optimal search using all of Z or by smaller, individually optimized searches using effort z_i, where $Z = \sum z_i$, provided each of the searches is optimized with respect to an updated version of the search object location probability density distribution which accounts for the effects of all previous searching."

Stone [153, page 228] notes that this result "means that the search planner need not consider the time horizon in planning stationary searches, but need plan only each increment optimally." For a moving subject problem, however, the optimal effort allocation at each search phase is dependent upon the duration of the search. Stone [152] also shows by example that a uniformly optimal plan may not exist for a non-regular detection function over a discrete search space.

Stone [152] presents an adaptation of Charnes and Cooper's 1958 algorithm to optimally allocate effort over a finite number of cells when the detection function is exponential. The algorithm assumes that there is no restriction on effort allocation. Stone then extends this solution method to consider restricted effort allocations, where the optimal plans are approximated for practical realization. Solution techniques based on Lagrange Multipliers are also presented to solve the resource allocation problem when searching for stationary subjects.

Often what would constitute a uniformly optimal search plan is not practical or feasible in application. In particular, Frost [150, page 6-19] notes that in some situations it may be necessary to sacrifice POS "early in the search effort in order to maximize the cumulative POS."

To create an optimal search plan it is necessary to either, first select the desired level of POS and then calculate the minimum search effort needed to achieve this, or given a level of search effort determine the optimal search task allocation [150]. The second method is usually necessitated as search effort is not generally under the search management's control.

Syrotuck [159] recommends that, prior to an actual search, various search combinations be evaluated for each region for the searching conditions most likely to be encountered. Syrotuck [159, page 8] advocates the use of a payoff table to determine the combination "which yields the highest, but reasonable POS payoff."

LaValla and Stoffel [103], supported by others, recommend that the first resources to ideally commit to an area are those skilled in clue awareness or tracking. Tracking dogs, if used, should be sent in first before the confusion of other searchers' tracks complicates the search. Secondly, search dogs and clue-aware "hasty teams" should be utilized. Subsequently, searchers and volunteers, whose primary job is to search for the subject, are sent in to the field, together with helicopters and fixed-wing aircraft if the conditions allow. Colwell [26] states that a search resource's predicted POD should be related to the priority given to that resource.

Sweere [155] comments on those computer programmes which recommend search resource distribution by computing all permutations of resources and task allocation. He states that:

> "Their greatest weakness lies in the massive amounts of scenario-specific information needed and the limited number of areas and resources that can be considered in a reasonable amount of time."

The problem of effort allocation in land SAR is currently addressed by a number of solution approaches, mainly comprising heuristics. These solution approaches are outlined below.

3.7.1 Probability of Density

Generally search effort is concentrated on those areas of highest probability density [150]. To this end the measure, *Probability of Density (PDEN)* is defined. PDEN is measured as the POA value of a search region divided by its size. Hence a higher PDEN value indicates that there is a greater likelihood of discovering the subject faster. The PDEN concept is a 'greedy' indicator which recommends first searching the smaller of two search regions of equal POA value, as this requires less searcher effort. A weakness with the PDEN statistic identified by Colwell [26] is that, if the size of the search region is not accurately calculated, "virtually any PDEN value can be obtained."

3.7.2 Sector Laddering

In the United Kingdom the visual *sector laddering* approach is currently used to rank regions for search allocation [80]. This method does not explicitly rely on POA, POD and POS calculations. Sector laddering is a dynamic list method which ranks sectors[4] in a ladder formation with the sector at the top of the ladder being the one with highest priority. Once the sector at the top of the ladder has been searched it is moved to the bottom rung and the sector now at the top is searched. Sectors can be inserted at the top of the ladder if new clues or information come to light that result in a higher priority rating.

Perkins and Roberts [138] state that they "have serious doubts about the validity of POS as a tool for management control" due to the high dependency on numbers where it becomes "all too easy for the search manager to lose sight of what is going on in the field." They instead advocate this sector laddering approach which does not rely on numbers and is simple to use. However, while the approach is simple to use it still relies on the subjective determination of the relative priority of the search regions and does not take into account the relative superiority of one region compared with the next region on the ladder, particularly when considering the time needed to search the regions and the rate at which information can be gathered.

3.7.3 Sector Stripping

Perkins [136] proposes a simple approach, which can be used under pressure, to determine which resource allocation to employ to optimize the total POS value at a given resource allocation stage *i.e.*, for one shift. The method performs an initial allocation of resources and then seeks to improve the POS value resulting from this allocation via heuristic improvements. These improvements are achieved by the application of simple rules derived from a restricted solution space of resource allocations.

Perkins develops the approach by limiting the number of possible solutions, assuming that search teams are spaced at critical separation to give a theoretical POD value of 50% for a single sweep of any area. Computer trials of a small model with three search regions (where a region was defined to be equal to one shift's worth of searching for a team of searchers) and three teams resulted in the identification of simple rules which could be applied to determine a resource placement which maximized POS. One such rule was that if

$$\text{highest POA} > 2 \times \text{lowest POA}$$

then a higher POS value was gained if two teams were sent to search the region of greatest priority and no teams were sent to the region of lowest priority. Furthermore, the second highest priority region should not be searched, in favour of all three teams searching the highest priority region if

[4] In the United Kingdom the search regions are defined as sectors, rather than regions or segments.

$$\text{highest POA} > 4 \times \text{second highest POA}$$

These rules were derived from considering the maximum POA value for lower priority regions which could be tolerated before an increase in POS was seen by not searching that region in preference to one of higher priority weighting. Such simplified rules are removed from physical calculations and are easy to implement.

Perkins identified the term *sector stripping* to this method where search teams initially tasked to search those regions of lowest priority are re-allocated to the regions of highest priority if this move results in a greater total POS — hence stripping the lowest priority regions of searchers. The initial allocation of resources is made by ranking search regions in order of their POA values and allocating each of the k available search teams to the top k ranking regions. The ROW region is not included in the sector stripping as it is not a region which is physically searched due to insufficient resources. Perkins advises discretion in applying the method due to the subjective nature of POA values, the method itself being reliant upon these estimates.

It is for this reason that we advocate no reallocations of resources in situations where a reallocation will result in an equal POS measure. In such instances it would be preferable to achieve some coverage of the lower priority area which would otherwise remain uncovered.

Perkins' analysis of this small problem, though practically unrealistic in size, indicates the usefulness of formulating simple rules that utilize comparison of the POA values of regions to determine search resource placement.

Application of the sector stripping approach to maximize the total POS for a given assignment is translated into stripping region i of search resources, to reassign to search region j, if

$$POS_i < POS_j^* - POS_j$$

or equivalently,

$$POS_i < POA_j(POD_j^* - POD_j)$$

leading to

$$POA_i < \frac{(POD_j^* - POD_j)}{POD_i} POA_j$$

where POD_j^* = the POD level of region j if reallocation of resources occur, and POD_j = the POD level of region j under the current allocation. This generalizes the rules of Perkins.

Instead of just applying the approach in a static manner at the commencement of each search phase, such rules could also be invoked in a dynamic application to reallocate additional resources during an operation to regions whose POA values have moved beyond some threshold.

3.7.4 Search Priority

Colwell [26] introduces the concept of search priority to provide an integrated planning tool which realistically takes into account the limited number of search resources available and the search effort required to achieve a set level of coverage. Search priority is described by Colwell [22] as an "holistic approach" which prioritizes and ranks the possible search resource assignments.

Colwell gives two definitions of search priority. The first is area-based and is identical to PDEN, defined as

$$\text{search priority} = \frac{\text{POA}}{\text{search area (sq. km)}}$$

This equation indirectly incorporates the search effort required as a factor of area. The second equation presented defines search effort more accurately as the manpower needed to search an area to a given POD level. Here search priority is defined as

$$\text{search priority} = \frac{\text{POA} \times \text{POD}}{\text{number of searchers} \times (\text{access hours} + \text{search hours} + \text{exit hours})}$$

i.e. the probability of success divided by the total searcher-hours required to achieve that level of success. The total searcher-hours is a measure of total resource utilization rather than the linear time between resources moving to the search region and returning upon the completion of the search task. In essence, search priority is a cost-benefit analysis of the available search options. Each search option is ranked according to the associated search priority value, with the options scoring highly being searched first. Colwell [26] proposes that a tentative search priority list can be calculated as part of the pre-planning process, and modified to reflect information available for an actual operation. Search priority values are updated with changing POA values after each region has been searched.

In comparing the theoretical search results obtained when applying the search priority method to those obtained in actual search operations, Colwell [26] concludes that "the mathematical Search Priority model quite closely mirrors, and may help justify, the intuitive decision-making of experienced search managers." Colwell views the ease of applying this method to be advantageous in facilitating the fast deployment of resources.

Colwell [28] subsequently standardizes the search priority formula to apply to both grid and trail searches and regards them as search assignments of different sizes, the only change being to replace the *number of searchers* in the denominator with the value of *one* when the region to be searched is a trail. Colwell reasons that, when searching a trail with a team of searchers, the POD achieved is only a marginal increase over searching the trail with one searcher as "they are all searching along exactly the same very narrow area, in the same direction and at the same time." Hence the team can be viewed as "one effective searcher." In fact Colwell [28] assumes a default POD value of 100% along a trail being searched by a team of searchers.

3.7.4.1 Multiple Resource Deployment

Colwell [23] proposes a four step procedure to determine the most efficient and best sequence of resources to deploy to one search region, when multiple search resources are considered. The approach draws upon search priority and updates POA values as searches are executed. The approach considers the multiple resources as a total deployment entity. The four step procedure is described as follows:

1. Calculate the cumulative POD for the combined resources, *i.e.*, the probability that the subject is detected by one of the resources.
2. Sum the total search effort required by these resources to obtain this cumulative POD, where the total search effort incorporates the search hours along with the time needed to access and exit the region.
3. Update the region's POA value after each resource has been deployed at its predicted POD level.
4. Calculate the search priority value for the combined resources.

The formula for calculating the search priority for a sequence of resources to one search region is defined as

$$\text{search priority} = \frac{POA^* \times POD_{cum}}{\text{total resource hours}}$$

where POA^* is the updated POA value of the search region after the penultimate resource has completed their search of the region, and POD_{cum} is the POD level achieved by the sequence of resources, assuming independent searches.

The procedure works on the basis that a list of alternative deployment sequences is available, hence reducing the size of the solution space and the computational requirements of determining the best solution.

It could be argued however, that the search priority value resulting from the search of the region by the multiple resource deployment could be more accurately expressed as

$$\text{search priority} = \frac{POS}{\text{total resource hours}}$$

where

$$POS = POA_0 \times POD_1 + POA_1 \times (1 - POD_1)POD_2 + \ldots$$

$$+POA_{k-1} \times (1 - POD_1)(1 - POD_2)\ldots(1 - POD_{k-1})POD_k$$

and k resources are deployed consecutively. In this formula the probability of detecting the subject with each search is determined by the adjusted probability that the subject is in the region, where this is adjusted to reflect all searching to date.

An alternative strategy that presents itself, which has lower computational requirements and is myopic in nature, is to originally rank (with respect to search priority) all available search resources as single deployment entities. The resource having the highest ranking would then be allocated to the search region producing that ranking. The POA value of that region would then be updated to account for non-detection by that resource and search priority values for remaining resources available to search that region would be adjusted to represent the updated POS. The resource now having the highest ranking would be allocated the associated search task and the process would continue until all resources had been allocated to search regions. This method independently calculates a search priority for each prospective resource of a multiple deployment and sequences these individually.

3.7.5 Probable Success Rate

The *Probable Success Rate (PSR)* [62] for a search region estimates the rate of POS growth which could be achieved by searching that region and is calculated as

$$\text{Probable Success Rate} = \text{Effective Search Rate} \times PDEN = \frac{W \times speed \times POA}{area}$$

Both the PSR and Colwell's search priority measure are examples of harvesting rates, measuring the POS returned from a given effort. Both equations have units of the probability of detecting the subject per unit of time. The main difference between the two approaches is that Colwell directly incorporates the time needed to access and exit the search region as well as the total hours needed to search it, while the PSR formula indirectly incorporates the search effort required and the time needed to search the region in the variables *speed* and *area*. Detection is measured specifically by W in the PSR formula.

Frost [62] proposes a resource allocation strategy which searches the region having the highest PSR value at each stage of the search operation. This strategy seeks to search a region until its PSR is reduced to that of the region with next highest PSR, then both these regions are searched together. Frost provides small numerical examples to show that, in comparison, searching by the simplistic strategies of:

- searching regions in decreasing order of POA;

- searching regions in decreasing order of PDEN; and
- searching all regions with a uniform distribution of resources;

are unreliable and can result in sub-optimal search plans. In particular the uniform distribution of effort strategy was shown to be the worst performer.

3.7.6 A Lagrangian Technique

Sweere [155] resolves the problem of resource allocation by applying Lagrangian techniques. The approach is based upon a new definition of the POD value as a polynomial function, there being one function for each resource assigned to a specific search task. The coefficients in the function are determined from field-tests and statistics of past operations with "limited on-scene modifications." Using these definitions of POD a new definition of POS is obtained which considers the time taken to access and search a region by a resource, the re-searching of regions and the search of a region by multiple resources. Specifically:

$$\begin{aligned} POS_a &= POA_a \times 100\,(1 - (1 - (POD_{a,1} \times (time_{a,1} - Taccess_{a,olda,1}))) \times \\ &\quad (1 - (POD_{a,1} \times time_{a,1})y_{a,1}) \times \\ &\quad \ldots \times \\ &\quad (1 - (POD_{a,b} \times (time_{a,b} - Taccess_{a,olda,b}))) \times \\ &\quad (1 - (POD_{a,b} \times time_{a,b})y_{a,b})) \end{aligned}$$

where

$y_{a,b}$ is the number of times resource b re-searches region a;
$time_{a,b}$ is the time that is needed by resource b to search region a;
$Taccess_{a,olda,b}$ is the time it takes for resource b to travel from their current location, $olda$, to a new region a.

Furthermore, POA_a is the POA value of region a and $POD_{a,b}$ is the POD level used to search region a by resource b, and is defined below for ground searchers performing a sweep search.

$$POD_{a,b} = site \times area_a \times resource_b \times team_{a,b} \times (c_1 d_{a,b}{}^3 + c_2 d_{a,b}{}^2 + c_3 d_{a,b} + c_4)$$

where

$d_{a,b} = people_{a,b}/$width of search area — representing the inverse of the searcher spacing (or distance between sweeps) for resource b in area a;
$people_{a,b}$ = the number of resource b performing the sweep in region a;
c_1, c_2, c_3 and c_4 are the coefficients of the equation.

The $site, area_a, resource_b$ and $team_{a,b}$ variables comprise further multiplicative factors affecting the POD level. These factors include the experience and speed of the searchers; the subject's detectability; terrain, weather and light effects; the distance sound travels; searcher fatigue; the impact of controlling large number of searchers; and the quality of management and communication. Sweere [155] envisions that terrain and vegetation effects could be analysed by software routines which "analyse digitized topographic maps and choose the appropriate values for each area."

The best placement of resources ($people_{a,b}$) is found by creating one POS equation for each search region, given the resources available to be used. Adding these equations gives an overall POS equation for the operation at the current point in time. A Lagrangian function is then created by incorporating a number of constraint equations, but we omit the details here. Sweere considers such constraints as the speed of the search resources, the minimum number comprising a team, and limits on the number of resources, including radio operators, and the time they are available for. Sweere solves a sample problem on MAPLE consisting of four search regions and ground resources performing two types of search pattern.

Sweere recommends the utilization of an initial heuristic phase to select starting points so as to reduce the running time involved in solving problems consisting of a larger number of regions and resources. He also suggests that determining the best three resource distributions would allow the search manager to select what he considered to be the best resource allocation — allowing him to consider these solutions in the light of other decision factors and on the weight of his own experience. In this respect the solution obtained by the method does not guarantee an optimal resource allocation. Sweere considers that this tool is best used in large searches where computer support people are available. He recommends that the solution be updated in response to changing POA values "after a significant number of areas have been searched and/or during a lull in the operation."

Sweere hopes that the future will see an algorithmic approach to mathematically defining POA values for a search region (based on subject stereotypes), and that this would be incorporated into search software with an optimization approach to resource allocation.

Sweere proposes a mathematical framework for resource allocation which incorporates search pattern, environmental and human factors that impact upon subject detection. These are incorporated into the solution procedure along with practical contraints. Like Colwell, in his search priority method, Sweere incorporates the access and search times required to achieve a given level of coverage directly into the objective function. The framework proposed is well considered in its inclusion of realism in a way that utilizes stereotypic information that is able to be gathered primarily from past experimentation and experience. However, it would appear that no scientific detection model, as such, is incorporated within the polynomial functions. Also, the solution approach grows in processing time with each additional set of variables required, resulting in a process that will become unmanageable for problems of a realistic size.

3.8 Clue Detection

Searching is becoming increasingly clue oriented. As one subject can leave thousands of clues, it is a more efficient procedure to search for these clues than it is to search solely for the lost person. Gathering clues to the subject's past movements or intentions enables informed calculations to be made as to possible positions, behaviour, and the current mental or physical state of the subject. Clues can reduce the size of the search area by providing physical evidence of a previous position, along with additional information such as a direction of travel.

Clues can take many forms: physical disturbances, scent, found belongings, travel intentions recorded in hut log books, visible lights or smoke, and character behaviour observed by other people. Such clues are used to build up a profile of the lost person that is continually updated as new information is gathered. Clues detected are rated on their relevancy to the subject and hence to the search itself, and are used to construct a time sequence of movements and events. Often search dogs and trained trackers will be employed if available, and if the weather and other conditions are favourable. It is important to prevent the search area from becoming contaminated by the presence of too many people who may obscure clues or create confusion [103]. A trained searcher can often ascertain the approximate time that a clue was left by considering such factors as its location and past weather conditions.

One search approach utilized is that of *binary search*. Binary search attempts to eliminate regions that the subject has not passed through. *Perimeter cuts* are carried out along search boundaries in an attempt to detect any signs of disturbance — entry or exit points — which may have been made by the subject, and any clues which may have been left behind. Often the technique will cut across the subject's proposed route at right angles to determine whether or not they have passed that way [103]. Cuts are also performed around specific signs of the subjects past positions. *Track traps* are used in a similar way. Ground is cleared at specific points, with these points being checked at intervals to determine if anyone has passed through the area, and in which direction. Recently developed devices can be placed to record the actual time of these events [81]. These methods are used to eliminate large regions from initial searching by confining the presence of the subject to a particular, bounded region.

Particular *hot spots* where physical clues are often detected more easily are in wet mud, especially along stream and river banks; alongside the subject's car (if parked at the entrance to the region of interest); around obstacles across the path of traversal; and sandy areas (Emergency Management Ltd. Introductory Track and Clue Awareness Course, May, 1996 and [170]). If a clue is detected in these places, an area is marked off around it to avoid distubance, and a more thorough investigation of the site is conducted. In particular, a thorough examination should be conducted of any hut or campsite that the searchers encounter.

Unfortunately time and adverse weather destroy clues, so quick action is required. The probability of detection of clues is also influenced by the level of clue awareness training undertaken by the searchers and the searchers' level of fatigue. Some clues can be detected more easily if viewed from a particular direction, and the use and angle of light can be effective in detecting things previously unseen (Emergency Management Ltd. Introductory Track and Clue Awareness Course, May, 1996). Due to this fact night searching can often be more effective than daylight searching, as light, and the direction it is cast, can be more easily controlled by a torch beam.

The *Relevance Of a Clue (ROC)*, in the current search is used as an input into redetermining the POA of a search region upon clue discovery [77]. ROC is given as a percentage, subjectively defined by search personnel. To be used as an accurate planning tool as much information as possible about the clue must be gathered. The ROC value is then used to update the POA value of the region that the clue was found in, viz:

$$POA^* = ROC(1 - POA) + POA$$

3.9 Search Methods and Tactics

Weddle [170] advocates that there is "no one "Best Method" ... a combination of techniques is typically used when conducting a wilderness search."

LaValla and Stoffel [103] define a search action plan as needing to be dynamic, and they cite their preference for only one plan, which is updated for each operational period. Hill [88] states that such an action plan "provides a framework for search strategy and tactics", which encourages proactivity rather than reactivity, and anticipates and plans for unforeseen events. Hill advocates the use of a preplan to guide decisions and actions needed to respond to such events.

LaValla and Stoffel advocate 12 hour shifts from 6am–6pm, when the night shift comes on to search the next shift through to 6am. There should be an overlap (approximately one hour [88]) between shifts to transfer information to oncoming teams such as new tasks, *etc.* LaValla and Stoffel recommend planning at least 12 hours in advance and partition the search operation into specific phases. The first phase consists of the initial response to regions of high probability — here speed is considered the crucial objective. The next phase of the operation counts speed and efficiency as priorities and targets regions of high probability. This phase usually comprises the first 12 hour shift with the second shift looking at new high probability regions, and regions previously searched. This second shift is focussed primarily on efficiency and thoroughness. Subsequent shifts consider regions of high and low probability simultaneously.

LaValla and Stoffel [103, page 55] consider that initial priority should be made to investigate "the potentially most life-threatening or serious" possibilities, with less urgent responses to other possibilities. Search objectives will define an acceptable level of search thoroughness in relation to the time available.

3.9.1 Passive and Active Search

LaValla and Stoffel [103] advocate the combined use of a passive and active search mode. A passive search involves the utilization of tactics aimed at confining the subject to a particular area. It uses methods of attraction such as sirens and smoke to alert the lost person of the direction of help, if they are able to make their own way out. The active search mode is where the searchers get out into the field and try and locate the subject. Decisions as to the most appropriate balance and use of the two modes need to be based on the perceived urgency of the circumstances surrounding the operation, the available search resources and any hazards.

Currently within New Zealand SAR, active rather than passive search is strongly advocated with a "go out there and get 'em" mentality. However, some confinement techniques are used in the physical placement of search teams in active search [165].

The active mode of searching consists of three methods, executed in the following order:

1. A hasty or reconnaissance search, quickly responding to regions of high probability where speed is the objective. Also referred to as a Type I search.

2. A fast, systematic search of segments with high probability where efficiency is the objective. Also referred to as a Type II search.

3. A slower, highly systematic search where thoroughness is the objective. Also referred to as a Type III search.

We now examine each of these three search methods in more detail.

3.9.2 Reconnaissance Search

The Reconnaissance Search is the initial search phase and its purpose is to either find those lost, or if unsuccessful in this, to narrow down the search area by either eliminating regions already searched, or by discovering information as to probable locations or movements of the subject. If time factors are against a thorough planned reconnaissance search, then decisions must be made as to what level of initial search will be undertaken and in what form.

LaValla and Stoffel [103] stress the importance of clue-consciousness in the initial search phase with a preference for sending in searchers who are trained in this, together with trackers, dogs, and aircraft — if the circumstances allow. Generally it is assumed that the subject is responsive at this stage. In particular the physical search concentrates on the position where the subject was last seen or the path they were known to have taken, checking specific locations and routes *eg.,* tracks, streams, huts, water sources, campsites, ridges, advantage points, and local hazards. Generally the path taken is that of least resistance.

Within New Zealand, helicopters are used to fly in and check huts in the area. Helicopters or fixed-wing aircraft will search regions of high probability if the conditions permit, making a return pass along the initial path to maximize the possibility of an individual or group attracting attention [66]. Teams are placed into areas where the most serious harm to the lost person could occur. Often a public backlash results if the person is later found close to a path and the searchers did not search the most "obvious" locations first [165]. Attempts are also made to minimize the physical searching effort of teams, such as that involved in searching up and down valleys. One way to achieve this is to split a team of competent searchers so that one half searches up one valley and the other half up the next, with both meeting at the top to continue searching from that point [165]. Sweere [156] recommends that, when searching trails, searchers should be transported to "the outermost edge" of the search region and then search inwards from this point.

"Rapid search groups" are initially deployed by Perkins and Roberts [137] as personnel arrive at the scene. They are instructed to search along known routes, probable exit routes and locations of high probability. This initial searching employs the critical separation technique and follows terrain features to reduce navigational demands [135].

Syrotuck [159] also advocates the use of 2–3 searcher teams for the reconnaissance phase of the search and indicates that for a moving subject, likely to be following a track or stream, it may be advisable to send teams in from either end in the hope of more quickly detecting the subject.

Within New Zealand it is considered that "except where timing is a critical issue adequate reconnaissance will be a key factor towards the successful conclusion of a search", [121, page 2.1]. The time taken for this phase will be dependent upon the urgency of the operation.

3.9.3 Efficient or General Search

This search approach is also used under the assumption that the subject is still responsive and may be the first method employed if the area to be searched is heavily vegetated [103]. An efficient search is described by LaValla and Stoffel [103, page 101] as one "using techniques that produce high probabilities of detection per search-hour of effort." The methods are applied to specific regions, highlighted from the reconnaissance search or other information, as having the highest priority. The regions are then searched using a sweep search technique with a wide spacing between searchers. A sweep search is performed by a line of searchers moving in parallel, with a grid search being produced when these searchers complete one sweep and return in the opposite direction, offset from their original track.[5] The form of search undertaken is dependent upon the denseness and type of terrain composing the search area, as well as the size of the object being searched for. In New Zealand's virgin bush, if there exist a number of ridges and valleys these are searched by moving from one ridge, down to the valley and up to the next ridge. Searchers will be guided in their search by compass bearings, grid references and/or features of the terrain.

Current Police practice in New Zealand is to allocate search regions by overlaying grids upon topographical maps or aerial photographs. Boundaries are defined by natural boundaries such as ridges and streams. If no natural boundaries occur, artificial boundaries are constructed as the region is searched, by laying down tape, twine or spray-painting trees. Careful instruction is needed to ensure that region boundaries are searched in the instances where neighbouring search regions are searched by two differing search teams. In New Zealand operations, search boundaries are usually only marked in thorough searches. However if several teams are searching in close vicinity they may independently decide to lay markers to indicate where they have searched. These may take the form of signs on tracks such as stones or branches [165]. Sweere [156] recommends marking regions with survey tape including a marker which informs of the area covered, the time, leader and search spacing utilized.

Syrotuck [159] points out that spacing between searchers can only ever be approximate

[5]This definition of a grid search differs from the traditional grid search in New Zealand used to describe the search of crime scenes or scenes of major disasters, such as aircraft crash sites. This search divides the area into grids, which are marked out and searched meticulously by an individual searcher.

as it is difficult to maintain a fixed spacing accurately in the field, especially for larger distances. This means that POD values can also only ever be considered as approximate, and consequently the measure for POS. However, he considers that "the system tolerates errors in spacing that are most likely to happen ... and deviations of 30 ft (9.14 m) less or beyond do not seriously affect the overall outcome of success."

Syrotuck further addresses some of the practical problems faced in search operation planning. In particular he recommends that sweep teams should contain groups of 10–15 searchers (any larger number and a straight line cannot be easily maintained) spaced across the base line of the search region. He recommends that their departure be staggered with the outside searchers of each team marking their path with biodegradable material, for the following teams to follow. Colwell [21] recommends that "multiple, small-area search zones" be employed to attain reliable coverage. He gives as an example, strips of a width equal to three times the required searcher spacing. Colwell also recommends that teams of three search resources be despatched with staggered starting times, following a specified compass bearing. He states that "multiple sweeps may be performed simultaneously, preferably at right angles to each other."

3.9.4 Thorough Search

LaValla and Stoffel [103, page 102] promote this phase only as "a last resort", as it has a significantly lower cost/benefit return than other search methods and generally destroys clues. The techniques used in this phase are a closed grid or sweep search, requiring a larger number of personnel spaced closely together. Hill [88] states that such a search is "slow, inefficient, and hazardous to the lost person's health. However, it may be the method of choice when it is suspected that the subject may be unconscious or otherwise unresponsive, such as with Alzheimer's patients and small children." Generally non-thorough methods are employed when time is critical or the available search personnel are small in number, relative to the area required to be searched [103].

3.9.5 Specific Search Techniques

Within New Zealand another search technique employed is that of *Double Strip Searching*, depicted in Figure 3.1. This method is employed during the efficient phase of a search and searches a region in strips by sweeping twice over the region from two different angles.

The double sweep action allows for a "technique (that) is very efficient in that it incorporates a second complete search of the scene from a different angle of observation", [121, page 4.2].

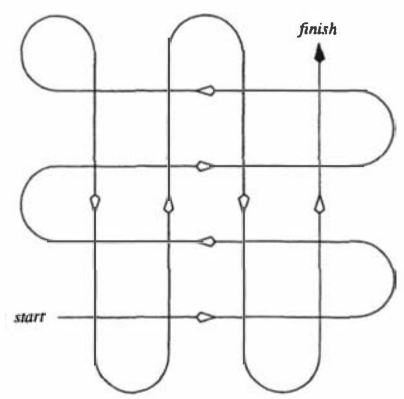

Figure 3.1: Double strip search.

Jennex [92] describes how aircraft searches utilize a variety of search patterns dependent on the conditions and the subject of the search. In particular a *Sector Search* pattern is utilized when searching for missing persons. The sector search is centred on the point of highest probability and is conducted in the formation of three equilateral triangles that are centred on this point. Such a search is illustrated in Figure 3.2.

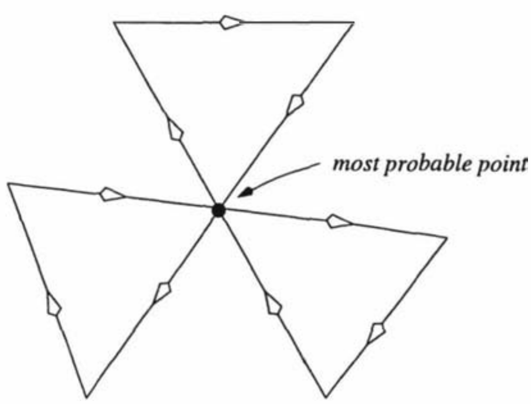

Figure 3.2: Sector search by an aircraft.

Each turn is 120 degrees deviation to the right of the current path and the distance of each path movement (representing a triangle edge) is found in relation to the subject — this is usually 4.8 km. If the search is conducted without success the pattern is offset by 30 degrees and repeated. Two highly trained spotters are employed and the airspace covered by one search pattern is 80 km.

Syrotuck [159] adapted the search method of *Repeated Expansion*, used by the USA Coast Guard, to land searching. The method is centred on a specific point of reference, *eg.*, the subjects predicted position, and consists of multiple sweeps over an extending area. The immediate area is searched and then this area is re-searched, expanding the initial area outwards to achieve an incremental area expansion with each re-searching of the previous region. A repeated expansion search is depicted in Figure 3.3 for three searches.

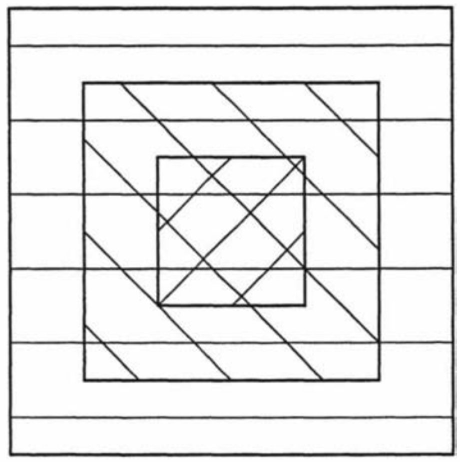

Figure 3.3: Repeated expansion search after three searches.

A variety of searcher spacings and direction may be used for the search of each expanding area and sweeps can be started shortly after one another for speed, if resources are available. Syrotuck [159, page 27] gives the main advantages of this method as the accommodation of increasing numbers of personnel which caters to "the natural buildup of personnel as the search progresses." He considers that the method is also economical with respect to searcher numbers and time, and plans ahead for an increasing search area which is usually necessary as the operation progress. However, Syrotuck notes that the technique may not be suitable for all terrain types.

3.9.5.1 The Sound Sweep

Colwell [20] introduced the concept of a sound sweep in 1993 as an effective and simple search technique. In a sound sweep the searchers, equipped with a radio, compass and whistle, maintain a line with a set spacing between each searcher. This spacing, to achieve a desired POD level, is determined from field trials conducted over the given terrain. Often the spacing will be such that searchers will be out of visual contact with other searchers. The searchers move forward at a constant speed for a set time interval. This time interval is derived from field trials in that terrain and season, and corresponds to the maximum distance that can be travelled which results in 100% success of a searcher hearing a subject's voice-response. At the end of the time interval searchers are stopped by a coordinated signal via radio from the search base and together create a noise, *eg.*, a whistle blast. The searchers then stop to listen for any possible subject response before continuing forward along a pre-assigned compass bearing. For increased POD values additional sweeps can be made at right angles to previous sweeps. The technique is dependent upon maintaining clear radio contact throughout the search period.

The sound sweep search method utilizes what has been seen to hold true in experience, that audible contact is usually established between the subject and searchers before visual contact is made, as long as the subject is responsive. Colwell [20] states that:

> "Experience on searches has frequently indicated that in many, if not the majority, of cases where persons are lost in forested terrain these subjects are heard by the rescue teams long before they are seen."

Canadian field trials conducted in forested terrain by Colwell and search volunteers, found that this search method successfully located 'victims' and allowed 3–5 times the searcher spacing than that required when searching visually. Colwell claims that the area coverage by sound sweep in a sub-alpine forest in winter, is up to 23 times greater than the historical field trials by Wartes of visual grid searching methods [25]. Colwell applied these findings to show that such an efficiency saving equated to the physically searchable area being up to 5 times greater than current visual sweep search methods. Hence fewer searchers would be required to achieve the same POD level. This then opens up options to the search manager to either expand the search area, or utilize the additional manpower by deploying excess searchers to increase the POD in other search regions.

Night trials of the sound sweep have shown that this technique is very practical for this environment, allowing time-critical searches to proceed throughout the night. Colwell [25] also advocates a similar method for initial reconnaissance trail searches, with the search team being prompted by radio as to when to create their audible signal. Colwell states that this method, "for a normal voice-response person, approximately doubles to triples the width of the trail-searched 'corridor' compared to standard visual searching." He provides data that supports a 100% POD on the actual trail, decreasing to 20% POD at the edges of the search corridor either side of the trail. Field data are given for corridor widths over differing terrain. As searchers usually perform this period of hasty searching at faster speeds, the timing between whistle blasts is less, *eg.*, 20 seconds for dense coniferous forest. This can be viewed as an inconvenience by some searchers, as stopping to detect any response also reduces the team's speed — a key attribute of this initial search phase.

Colwell [22] states that when an initial sound sweep of an area has been conducted and the operation reaches a stage where it is assumed that the subject is unresponsive, the initial sound sweeps are reassigned POD values relating to visual sweeps.

This search method has suffered criticism primarily on the grounds that not all those who are being searched for are responsive to sound, for reasons such as being: unconsciousness; a fatality; a person who is hard of hearing; or a child who is afraid that if found they will be in trouble. In defense of these criticisms Colwell [20] gives search statistics from Vancouver, Canada, which show that more than 90% of those lost were found alive within 1 or 2 days, and "were usually responsive to sound." Colwell [25] notes "that sound sweeps should be conducted within the first few days, and preferably within the first 48 hours, of search notification." There has also been some international skepticism of Colwell's experimental field trials by practitioners who refer to Heisenberg's Principle, which states that the mere act of measuring affects the measurement [80].

New Zealand trials of the sound sweep have seen a mixed response from searchers, ranging from positive to negative. In particular, some searchers complain of ringing in the ears after the combined team's whistle blast so that it is not possible to hear any response that a subject may make (Colwell [25] recommends plugging ears during the whistle blast). There have also been suggestions for a more practical prompting device that does not rely upon radio communication, *eg.*, an egg timer.

3.9.6 Night Searching

Within the New Zealand context, night searching is generally performed only in cases of extreme urgency, such as the lost person being a small child or having known life-threatening injuries — particularly if they are lost in adverse weather conditions. Searching is conducted by torchlight and usually on tracks or stream beds to reduce the danger to searchers [165]. It is also recognized as an effective technique in searches carried out on ski-fields as, once the mountain has been vacated, sound usually carries well.

In the USA, however, night searching is seen as a valid tool in itself. One reason for this is that the lost person will often remain stationary throughout this time enabling searchers to constrict the search area. Weddle [169] states that "while I can well accept that there may be some situations where darkness adds to the risks sufficiently to rule out night-time operations, I've yet to encounter such a situation."

Gordon [79] also strongly advocates this approach for a number of reasons. These include:

- maximizing the time available to search during the long nights over winter;
- that the subject is usually immobile, alert and motivated to actively assist the searchers;

Table 3.2: Matching of sweep search technique to subject type.

Subject category	Search type
Voice-responsive to sound.	Sound sweep
Subject wearing normally coloured outdoor clothing. Subject likely to shelter beside a tree or bush under winter conditions.	Standard sweep
Adults wearing high visibility clothing. Subject likely to move into open areas under winter conditions.	High visibility sweep
Unconscious person. Subject wearing camouflaging colours. Infants or toddlers in dense forest.	Low visibility sweep
Subject presumed to be a fatality.	Body sweep

- a response which is critical in urgent cases;
- that there is less contamination of the area, as untrained searchers and members of the public are unlikely to be in the region; and
- provision of a continuous momentum to the search operation, throughout a 24 hour period.

The main search resources used at night are track-and-clue aware searchers, and search dogs. Helicopters can also be used advantageously, in favourable weather conditions, as most carry specialized infrared night vision and heat sensory equipment [163]. Similar equipment and technology is available to ground searchers. Weddle [169] stresses the importance of training searchers at night and ensuring that sufficient light sources, including replacements, are carried. Sweere [156] considers the greatest difficulty of night searching to be that of accurate navigation and recommends that searchers should allow 20 minutes for their eyes to adjust before searching is commenced.

With the increased availability and development of night vision technology and specialist training, it would appear that this view of night searching being both practical and vital will become more widely accepted within the New Zealand SAR environment.

3.9.7 Search by Subject Category

Colwell [21] defines five subject categories and the type of sweep search appropriate for each. These categories and recommendations are recorded in Table 3.2.

The high visibility, standard, low visibility, and body sweep searches are all visual searches and are characterized by decreasing searcher spacing.

3.9.8 Re-searching an Area

In New Zealand it is considered that a thorough search of assigned search regions should be conducted as regions "will not be revisited unless there is a special reason", [66, page 37]. When re-searching does occur, different search techniques or search resources are applied, and preferably with a different search resource starting from a new angle.

3.10 Rescue

Factors which must be considered in rescuing a victim include their condition (which should be stabilized if possible); the current weather and limitations of rescue resources, including the number of available personnel and their physical and mental condition; the time of day; and the easiest route which can be followed. LaValla and Stoffel [103, page 128] state that the victim should be moved out in the "shortest time consistent with safety and simplicity." A deceased victim should only be removed from the scene if this causes no risk to the rescuers.

If a helicopter is unable to execute the pick-up of an injured victim, a stretcher carry is used to transport them out of the search area. If the distance involved is not far and the terrain is good, this can usually be executed by four people. However, in New Zealand if a long carry is required a Tararua stretcher and rope system are used, requiring 32 people forming two teams of 16 [165]. This is a very slow process, especially over rough terrain where pulley systems may be required to manoeuvre around obstacles such as waterfalls. Additional experienced people are required to clear a path before the carriers. LaValla and Stoffel [103] recommend that 6–8 people should carry a stretcher for periods of no longer that 20 minutes before being relieved and that three teams of lifters is a minimum requirement. Feeney [50] recommends that a long carry is executed by three teams of 8–12 stretcher carriers, who each carry for 10 minutes before rotating.

Rescue by helicopter, where possible, is preferable for the victim and at approximately $500 for an average rescue is an efficient use of resources compared to the time spent in a stretcher retrieval [165].

3.11 Search Suspension

In New Zealand the term suspension is used in preference to termination as it is less final to relatives. Under suspension, if further information comes to light, searching will be recommenced in some form [165].

The decision to suspend an operation when attempts at locating the subject have been unsuccessful is usually decided by group consensus after considering a number of factors. Such factors include [103]:

- is the subject most likely to be living or deceased?;
- is the subject most likely to be still within the search area?;
- the level of coverage of the search area, including the number of search regions re-searched;
- pressures from the family and media;
- predicted weather;
- the presence of any unresolved clues;
- pressure on search resources, *eg.,* another incident; and
- the cost of the operation.

In summary, such a decision is both time-and clue-dependent. Possibilities for a limited continuous search of the area are often evaluated, *eg.,* intermittent flights over the region. Future training exercises for the area may also be scheduled, missing person notices are posted, and the media is utilized to alert people who may be in, or travelling into, the search area. If suspension of the operation is decided upon, a post-operation critique should be conducted as soon as possible, evaluating the decision-making processes and results.

Gallas [66] gives examples of New Zealand survivors who were lost for some days in adverse conditions, including one tramper who survived for 29 days before being found.

3.12 The Global Positioning System

The Global Positioning System (GPS) has been described as "a huge step forward in the development of navigation", (cited in [123]). This is particularly due to their size and user-friendly operation, in addition to their proven reliability and accuracy when used by the USA and French military in the Gulf War [91].

A GPS receives signals from multiple satellites and can give the user's position in real-time, at any time. As well as informing the user of their actual position, it is also capable of giving information on their current course, speed, and computing a course to any given destination — warning the user if they deviate from this. GPS units can also be programmed with specific coordinates [91]. These may relate to hazard areas or other locations such as an injured victim. The unit audibly warns the user if they are within a set distance of these locations. The only restriction on the use of a GPS is that the sky must be visible for a reading to be obtained. Programmes are currently available to convert a GPS reading (latitude and longitude), to a map reference or a map grid, for greater accuracy (New Zealand National Search and Rescue Course, April 1996).

GPS units are accurate to approximately 60–100m when in motion ([91], New Zealand National Search and Rescue Course, April 1996). The size of the error is introduced by the United States Department of Defence to ensure that civilians do not use them for military purposes. Hill and Bower [91] do not consider this a concern in relation to their application to SAR as "studies have shown that even a 100m error is tolerable for most navigational functions."

As portable GPS units decrease in size and price, becoming more affordable for the average tramper and hunter to carry into the bush, land SAR itself could change focus and role in many instances. This has already been seen in marine SAR when lost craft have carried satellite detectable, Emergency Position Indicator Radio Beacons (EPIRB). The role of the searchers then becomes one of detecting the beacon's signal and locating the vessel by aircraft. The following comments made in relation to the usage of electronic emergency beacons in marine SAR would seem to be directly applicable to the use of beacons and cell phones in land SAR:

> "Although the relative number of SAR incidents where significant searching is required may go down, it is highly unlikely that the need for searching can be completely eliminated in the foreseeable future. Searching will remain an essential element of SAR for some time to come", [150, page 1-2].

A firm based in Christchurch, New Zealand has been researching and developing a personal locator beacon to be carried by those on outdoor recreational activities. The Human Emergency Location Positioner (HELP) can be activated manually or remotely if the person is unconscious or disabled. If remotely activated and the holder is not in need of assistance a reset button can be pressed to communicate this fact. The activated beacon can be detected by aircraft or from the ground. Schemes where it is possible to rent such devices may see them more accessible for the average budget, however, there will always be those who do not take such precautions and who get into difficulties.

It has recently become popular to carry cell phones into the bush for use in emergencies. There have been a number of cases now where through accident or misfortune people have been in difficulty and have used their cell phones as a means of communicating to emergency personnel, with a successful outcome. However, due to terrain features and the range of cellular networks such contact cannot always be established. Some of the same disadvantages exist as with marine beacons, including that they may not be where they are needed, they may not survive in working order, being battery powered they have a limited life, and they may not survive an accident [150].

GPS units can also be carried by individuals to relay their exact positions back to a centralized base. This is currently being implemented by the New Zealand Police Department in respect to the movements of their patrol cars [87]. Fischbeck [52, page 45] predicted such real-time location information for routing applications as a progressive area for Operations Research, with information being " relayed back to a Geographic Information System from a service vehicle in the field allowing for optimal routing."

There is becoming an increased interest in utilizing GPS units in SAR as a means of improving the efficiency of the operation. By requiring searchers to carry a GPS unit, more accurate positional information would be provided and be available to search management to aid in future decision-making. This is highlighted by Hill and Bower [91] who envision that GPS will be used in land search operations not only as a navigational aid but "a much more comprehensive application of GPS, with full integration into search management." This would include the ability of the GPS units to relay the locations of search resources directly to computer equipment at the search base, enabling translation to graphical displays — visualizing the movements of each search resource — and providing analysis information after the conclusion of the operation. Colwell [24] determines that the use of GPS in updating team locations will enable "more precise route and grid searching and much more accurate debriefings." Hill and Bower consider that, in addition to removing much of the current effort and lack of precision in updating the positions of searchers, victims and clues, navigational errors which presently occur in determining directions to particular search regions and maintaining sweep direction coordinates will be avoided.

A concern highlighted by Hill and Bower [91] that requires further research is the determination of "whether GPS accuracy is affected by wet, heavy foliage or other possible sources of interference, such as narrow ravines" — conditions found in many SAR operations.

3.13 Computerized Search Planning

3.13.1 Marine SAR Computer Software

One of the first computerized planning systems developed for SAR was the *Computer-Assisted Search Planning (CASP)* system developed for the United States Coast Guard (USCG). Richardson and Discenza [143] detail this system which became operational in 1974 and was designed upon a modular system to enable the later addition of further enhancements.

The CASP system utilizes Monte Carlo simulation, in a multiple scenario approach, to generate the initial subject location probability distribution and to update this distribution to incorporate drift. Subject motion is assumed to be Markovian. A hybrid approach utilizing Monte Carlo simulation of the subject's movements and analytic methods to compute detection probabilities was favoured as many mathematical models "tended to force the facts to fit the mathematics to an undesirable extent", [143, page 662]. The system utilizes an *M-Beta* detection model with uncertainties in navigation being modelled by assuming that "each sweep is a random parallel displacement from the intended sweep", [143, page 668].

CASP generates initial subject location probability distributions from three scenario generators, where the generator chosen depends on information known of the subject's movements. A weighted scenario approach is utilized over k scenarios having subjectively assigned normalized weights. The first of the three generators is *Position*. *Position* is used when there exists information as to the whereabouts of a subject's position at the time of the distress. The subject's initial position and displacement are modelled by probability distributions with displacement ranges input by the user. When an initial position is not known, but the general region that the subject may be located within can be determined, the generator *Area* is utilized. The *Area* generator determines the location probability distribution of the subject based upon a uniform probability density over a convex polygon.

The third generator *Trackline* is utilized when there exists known information on the track taken by the subject. It assumes a definite point of departure and an intended destination. *Trackline* simulates possible tracks taken, by sampling from three circular normal probability distributions placed at the initial, mid-point, and end point of each track segment. When distress is most likely to have occurred at a specific point along the track, *e.g.*, due to bad weather being experienced at that point, then the probability distribution can be adapted to have a higher density in that area.

Search effort allocation is optimized by CASP by utilizing two programmes; *Map* and *Multi*. *Map* assumes an exponential detection function and does not impose a constraint on the type of search pattern. Under these assumptions, Stone's [152] adaptation of the Charnes-Cooper algorithm is utilized. The procedure *Map* combines cells with equal probability density to form a collection of equi-density search regions. Regions are then ordered with respect to probability density and the region with highest density is searched first. E_k is defined as the total search effort which should be expended before searching the kth region and is given as:

$$E_k = E_{k-1} + (\ln d_{k-1} - \ln d_k) \sum_{m=1}^{k-1} A_m$$

for $k \geq 2$, with $E_1 = 0$, d_k = the probability density of region k and A_k= the area of region k. The procedure quickly defines the area to which to apply search resources.

The second optimization procedure *Multi* takes into consideration further operational constraints when multiple search resources are allocated to search the same area, with each resource uniformly searching a non-overlapping rectangle. The dimensions of the optimal search rectangle are calculated with the associated probability of detection based on a normal subject location probability distribution. These are evaluated over the differing assignment possibilities with the assignment being selected which gives maximum probability of detection without overlap. If more than one resource is assigned the same rectangle to search, the rectangle is subdivided so that it can be uniformly covered.

The information generated by the CASP system is output in the form of a sequence of probability maps for each time period. The maps are displayed two dimensionally by a grid representation with the subject location probability displayed for each grid. Equal probability contours can then be drawn upon the map to enhance visualization or symbols can be used to represent probability ranges to enable a fast appraisal. The CASP system is also capable of outputting an ordered list of the highest probability cells along with the amount of search effort which should be assigned to each cell.

The current planning system used by the USCG is the Improved Search Planning Method (ISPM), [150]. The system has enhanced flexibility to handle differing types of subject location probability distribution, two types of search conditions (ideal and poor), and differing levels of available search effort. The ISPM takes advantage of the advances in computing power to create optimal search plans via an iterative algorithm which increments the size of the search area until the POS declines. At this stage the algorithm goes back two steps, halving the increment size and proceeding. These steps are repeated until the size of the increment becomes very small [150].

The USCG approaches the decision of search task allocation based on the assumptions of equally spaced, parallel sweep searches over one contiguous area with a uniform distribution of search effort.

3.13.2 Land SAR Computer Software

Syrotuck [159] in his 1975 paper discusses in concept form what he foresaw as the future utility and usefulness of computerized search planning. In particular he considered features of the CASP system and its possible application to land SAR. Syrotuck discussed a system which would ideally have the ability to: incorporate all initially known subject details; allow statistical generation of the victim's possible locations and actions; develop and output scenario descriptions and applicable probability distributions; generate maps; output detailed search plans; utilize the particular terrain and weather conditions present; optimally plan — utilizing available resources; update plans as new information comes to hand; summarize information and subsequently update a statistical data base at the conclusion of the operation, to enable "the computer to gain 'experience' ", [159, page 33]. He pointed out that the "experience" gained would be greater if it were a centralized system used by many agencies. Syrotuck considered that the main stumbling block towards developing such a system would be the cost of designing the system and staffing it for the 24 hours a day, 365 days a year, that would be necessary.

Syrotuck proposed a map output consisting of a grid of numbered reference squares, referenced by longitude and latitude. The concept was that the computer would generate several search plans for possible victim positions. Correction factors would be incorporated for the effect of differing terrain, visibility and searcher fatigue on POD calculations, and different parameters would also be used for night searching. Since this time a number of software solutions to aspects of the land SAR operation have been developed, of which we will now overview a selection.

One of the most comprehensive programmes to be developed, and currently under further development, is the *Search Manager Program* designed by Colwell of SAR Technology Inc., Canada. This programme assists search management by providing a set of tools to enhance good decision-making in a SAR operation. The programme calculates such things as search priority, (theoretical) search area size based on travel speed, grid search logistics, and cumulative POD. It can also incorporate local area and statistical information, and it facilitates consensus decision-making and the construction of a subject profile. The programme is designed to supplement, not replace, experienced search personnel. For accuracy, search managers are required to gather data specific to their region for input into the programme.

PODSheet is a set of Microsoft Excel spreadsheets initially developed by Carnes and then further modified by Cooke [14]. The spreadsheets execute search planning calculations from data input by the user. In particular PODSheet determines the following: initial POA values via the Matson Consensus method; the number of personnel required to execute a grid search of given input parameters; the expected POD from a grid search of given input parameters; and updates POA values after the completion of a search or upon the discovery of a clue. PODSheet utilizes the critical separation theory of Perkins to determine POD values. Carnes and Cook specify that the POD value resulting is the expected minimum value in light of Perkins and Roberts' field results. PODSheet does not have the ability to keep the records generated throughout the operation and it is assumed that users are familiar with POA and POD theory, with only minimal assistance being provided in terms of warning messages and some recommended input values.

The electronic SAR database *SARDAT*, developed by Downs and Dolan for the San Diego County Sheriff's SAR Bureau [43], manages historical data from previous search operations. The information stored in the database is provided in a form which enables search managers to access information pertinent to the operation that they are currently involved in.

SARDAT manages data to store informal local knowledge and experience in an easily accessible form. The data maintained includes subject profiles, resource and weather information. The database allows data queries to access relevant information in a timely fashion. SARDAT also provides documentation for a search operation in progress, adding search information to the database as it is generated. Downs and Dolan state that their intention is for SARDAT to be used as a predictive tool for search operations and to be used as "a tool in helping to suggest solutions to ongoing search problems." SARDAT can be used alone or in tandem with *SARMAP* to relate subject profiles with local terrain. SARMAP is a computer programme developed by Dolan which provides electronic topographic maps "combined with user-defined overlays."

In 1995 Handel [86] developed the *Relative Urgency Response Factors (RURF)* software which runs on Windows and is an adaptation of Wade's method to measure the relative urgency of a search situation. The software is designed to assist search management in evaluating urgency at the interview phase of the search operation "when initial information is sketchy or inaccurate." The programme also guides management to an appropriate search response in light of the information gained.

Tristram [165] favours a greater use of computers in New Zealand SAR. This includes the application of simulated computer programmes for training purposes, generation of computer tasking worksheets and the use of a modem to relay such data to remote locations *e.g.*, a laptop computer in a field vehicle.

Colwell [24] emphasizes the need to integrate search variables into "a single, cohesive plan." He foresees that the emergence of new search software will result in the use of "integrated search mathematics within planning software with the ease that we currently use the basic calculator", with the next step being an integration of these with graphical software. He identifies desirable features of such software to include: the update of team locations via GPS; the ability for management to 'draw' upon a software map; search calculations including time, manpower requirements and POD; and probability areas being graphically defined by statistical and local information, input by the user.

Colwell [24] concludes that such software, along with more detailed lost person behaviour analysis,

> "may transform the art of SAR into a genuine science. When the potent combination of science and technology is blended with effective, quantifiable field skills we will have produced a powerful new technology that will significantly improve our capability to find the subject, and therefore, to save lives."

We consider that such a system will have the capacity to not only aid decision-making for a particular operation but to also simulate differing scenarios, outside actual operations, as a learning tool. Such simulations would consider differing behaviour and actions of the lost subject, differing terrain, weather conditions, and available search resources. Any such computerized system would need to be user friendly and be able to provide real-time data. The system would also need to produce successful outcomes for search personnel to place faith in its use.

Future possibilities also exist in utilizing parallel processing applications to solve search problems. Stone [154, page 505] considers that parallel processing "holds much promise for improving the speed and capability of search planning algorithms" as many of these algorithms contain "natural parallelism."

3.14 Limitations of Current Land SAR Approaches

Colwell [26] in reference to such concepts as POA, POD and POS, states that:

> "Each of these concepts have improved the theoretical effectiveness of the search process. However the implementation of these techniques into actual search operations has been fairly slow, as certain aspects of these techniques have not been well matched to the realistic constraints of actual search conditions."

Colwell outlines particular limitations which impact on the realistic utilization of POA values. These include: being unable to divide difficult terrain into realistic search regions; the large number of search regions required for larger search areas; and, difficulties in determining the size to make each region.

Further limitations of current planning approaches to land SAR are included in the following list.

- A lack of research into the detection functions of search resources searching for a subject or clues. This is needed to give meaningful POD values for a wide spectrum of search environments and search objects.

- No correction factors for differing terrain and search conditions.

- A lack of rigour in search task allocation. Methods are needed which provide guidance in the allocation resources with respect to set operational objectives and dynamic planning is required to respond to events which occur.
- A lack of utilization of available computing power to cater for changing information such as that resulting from clue discovery. Real time adaptation of search tasks and resource paths is required.
- An apparent lack of real-time analysis of search options for the current conditions.
- No consideration is given to searching performance as affected by searcher fatigue.
- No modelling of subject responsiveness over time and incorporation of this into search planning.
- A lack of realistic search time calculations, especially in a form which allows for ease of calculation of shortest time paths.
- A lack of computer use in modelling the search terrain with incorporation of access times and subject movement data, including visualization capabilities and the simulation of possible subject movements.

Frost [62] hopes that in the future a "practical search planning methodology based on search theory" would be developed for SAR operations. He considers that such a project, to be both scientifically rigorous yet practical, would need

> "a development team whose collective talents and knowledge covered the entire spectrum from the most mathematically esoteric aspects of search theory to the most practical aspects of planning and conducting search operations. Such an undertaking would also require significant amounts of time and resources. Tasks would include:
>
> - designing, developing and conducting sweep width experiments in inland environments;
> - testing and evaluating different search tactics; and integrating material and knowledge from many diverse areas of expertise into a clear, concise, coherent, practical, and scientifically valid set of guidelines for planning searches."

The last of the identified limitations, that of the need for an advanced computer model of the search terrain, is addressed in the following chapter. In particular, we examine the digital terrain models currently in use and their applicability in providing a model of the search area which provides adequate data for search management.

CHAPTER 4

Modelling The Search Terrain

The physical terrain over which the search is conducted requires a representation which is computationally manageable for real-time path planning and which is able to realistically capture the nature of the topography. As O'Rourke [128, page 270], we desire a model where "details are abstracted away from a real-life application to produce mathematically 'cleaner' versions of the problem" under the criterion that "if the abstraction is performed intelligently, the theoretical explorations have practical import."

To this end we turn to developments in Geographical Information Systems.

4.1 Geographical Information Systems

The rapidly expanding and influential area of Geographical Information Systems (GIS) is forging new links among a diverse range of disciplines, including Operations Research (OR) and Management Science (MS). GIS are versatile and powerful tools which "focus on spatial entities and relationships, together with specific attention to spatial analytical and modelling operations", [111, page 17]. The systems can maintain storage and retrieval for not only geographical data but historical, statistical, economic and social data, related to specific geographic regions. Hence, they provide a platform for many diverse applications, including problems requiring multiattribute decisions.

Potvin *et al.* [139] designed a computer system named ALTO which allows the user to test and evaluate heuristic solution strategies for complex vehicle routing and scheduling problems. ALTO is built upon a flexible 'general heuristic' which is capable of generating specific heuristic procedures — including those existing in the literature as well as user-defined approaches. This framework enables a dynamic approach to developing heuristic methods for problems where there is "an inability to deduce the kind of strategy that will perform well given the (problem's) characteristics", [139, page 451]. ALTO provides a graphical, interactive interface where the user is able to explore the effects of different approaches to identify promising avenues for future development. The authors consider ALTO to be the first step towards the creation of an expert system for this field of problems.

GIS can be linked with fundamental OR/MS techniques, including shortest path, vertex and arc routing algorithms, to develop powerful routing systems as seen in GeoRoute [101]. GeoRoute, inspired by the ALTO system [139] and designed by the Centre de Recherche sur les Transports de l'Université de Montréal in cooperation with their clients, provides software to solve complex routing problems over street networks. It incorporates an application-dependent and graphically interactive approach, and has since been released by Kositzky & Associates Inc. for use on the Windows platform [3, 85]. The designers [101, page 83] state that the "main innovation of GeoRoute is the use and manipulation of the structure of the network", the idea being "quite new in the GIS and operations research community", where a trade-off between visual and structural properties is often seen. Additional recent vehicle routing software that utilize GIS interfaces are surveyed by Hall and Partyka [85].

Emergency Information System (EIS) software is a leading crisis management tool developed in the United States to manage information in crisis situations. Situations that EIS has been used in include such events as toxic chemical spills, flooding, aircraft crashes, bombings, earthquakes and terrorism. EIS offers real-time decision support, powerful communication and visualization capabilities, and integration of geographical data. In particular the EIS/InfoBook brought out in 1995 combines EIS with GIS as a joint result of EIS International collaborating with the Environmental Research Systems Institute.

EIS is linked with greater terrain mapping capabilities in a specific Search and Rescue (SAR) operation application by Ketcham and Ketcham [98]. Ketcham and Ketcham linked EIS with digital orthophotographic map sets of high risk wilderness areas in Whatcom County, Washington, USA. They also incorporated aerial photographs to enable easier visualization of specific terrain features, particularly any that may have altered since the map's production. This inclusion enabled: the maps to be annotated with search information as it came to hand; map sections to be printed and distributed to searchers on the ground; searched areas to be visually classified; and searchers' locations to be identified with the aid of GPS units on the ground and plotted on the maps. The incorporation of the maps with the EIS software allowed search coordinators to track resources, and redirect them as necessary, from laptops at the scene. Ketcham and Ketcham state that "this application is extremely versatile so it can be adapted to fit any of the unique circumstances posed by each SAR mission."

> "Traditionally, SAR management has been of a more reactive nature, modifying its approach days into a mission after consolidating notes and opinions. Now, SAR coordinators can take a more preemptive approach as well as track up-to-the-minute search progress", [98, page 16].

As geographic information and data bases become more readily available — especially via the information highway — the full potential of such powerful systems will become a reality. In the USA there is already movement in this direction with the current TIGER files[1] and the "national spatial data infrastructure" [52, page 45] called for by Vice President Gore in 1994. Currently New Zealand terrain elevation data is available in a 20m contour set owned by Land Information New Zealand and distributed by Terralink. The copyright restrictions on this data are due to be lifted by the government in the near future. Pairman, of Landcare Research New Zealand, states the organization's intention to generate a nationwide digital elevation model, at a detail level of a 25 × 25m grid, from this data set early in the year 2000 [129].

Fischbeck [52, page 45] predicts that

> "within the next few years, a flood of data will be available that will allow for even more creative research opportunities for those merging OR/MS with the GIS technology."

It is this form of geographical data stored in spatial network structures that lends itself to the SAR application. By enabling OR techniques to be performed upon data structures which permit the handling of real-time routing requirements and yet also accurately represent reality, the realistic modelling of search terrain and the coordination of effort allocation can occur.

We consider now the specific constructs, known as Digital Terrain Models, which are used within GIS to model terrain.

4.2 Digital Terrain Models

A Digital Terrain Model (DTM) is differentiated from a Digital Elevation Model (DEM) in that it not only digitally represents the elevation of the land surface but also represents other topographical features. The DTM allows visualization and data storage of the shape, and attributes of the land surface to facilitate: interpolation of the terrain

[1]Topologically Integrated Geographic Encoding and Reference (TIGER) files that consist of USA Censor data containing geographical information on the entire country.

behaviour not explicitly represented; interpretation of available information; and identification of features such as water networks. DTMs are also useful for classifying the terrain into multivariate characteristics, for example, terrain density [171].

A DTM is suited to a variety of applications due to its flexibility and adaptability, however, the usefulness of any DTM is limited by its accuracy. This is dependent upon the sampling method used to determine the initial data points, as well as the quality of the terrain data utilized. In most cases DTMs are generated and then examined for their accuracy of fit to the land surface in question. If the fit is not considered to be "good", *i.e.*, the approximated surface does not fall within a specified tolerance level, further points are sampled from regions of ill-fit until the error of the fit meets the tolerance level. A computationally more expensive approach is to model all available points, selectively removing points until the tolerance level will be violated by any additional removals [106].

DTMs are generally formed of simply[2] connected surface models [171] and are described as being 2.5-D [171, 162], as there exists only one z coordinate for each (x,y) coordinate pair.

Weibel and Heller [171, page 272] state that:

> "A variety of data structures for DTMs has been in use over time. Today, however, the overwhelming majority of DTMs conform to one or other of two data structures: rectangular grid (or elevation matrix), or TIN (Triangulated Irregular Network)."

We now compare the characteristics of these two DTMs to determine the one best suited to modelling terrain for SAR operations.

4.2.1 Triangulated Irregular Networks and Rectangular Grids

Rectangular Grids (or Uniform Sampling Grids) approximate the topography by overlaying a grid structure onto the terrain. Each grid square is assigned a height value equal to the elevation value of the contour of the surface region enclosed by that grid square. Hence, the grids may be discontinuous at their perimeters, and approximations are necessary to smooth these inconsistencies when forming a path that moves from one grid square to an adjacent grid square.

The Triangulated Irregular Network (TIN) is a DTM where "2-D objects (triangles) are, strictly speaking, embedded in a 3-D space", [67, page 125].

A TIN geometrically partitions the terrain into triangles by a triangulation generated over a set of representative data points. These points are given as an (x,y) coordinate pair

[2] Simple in that non-adjacent segments do not intersect [128].

with an associated spot height and are selected to provide an economical representation, *i.e.*, one which utilizes the least number of points yet still retains the required accuracy level of terrain information for the application at hand. To form such an accurate representation, a TIN must be created from points sampled from critical physical features, with areas of varying terrain being sampled more heavily. Critical physical features include local minima and maxima, ridge lines and watersheds. As the selected points are irregularly positioned, an irregular network is created.

Lee [105, page 414] defines a TIN as follows:

> "A TIN model approximates a topographic surface by connecting a set of irregularly spaced elevation vertices into triangular facets. The triangles share edges and vertices to exhaust the space as if they were a triangular mosaic."

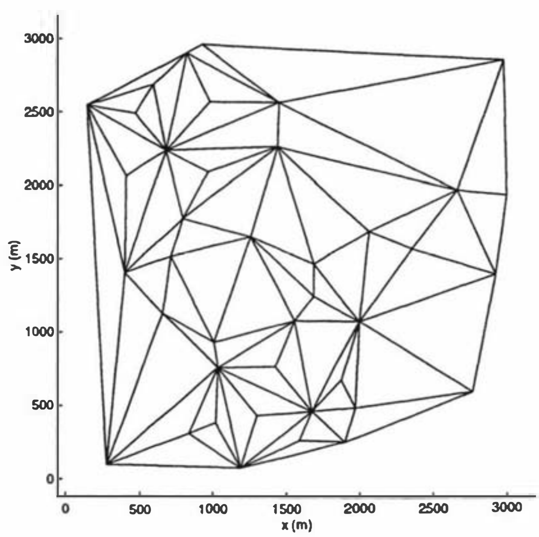

Figure 4.1: Triangulated Irregular Network (viewed from above).

It is often advantageous to construct TINs from regular square grid DEMs as the elevation data of the gridded DEMs is generally widely available at reasonable cost. Hence, TINs can be constructed for specific terrains for particular applications, based on either a required tolerance level in the elevation difference of the approximated surface to the original, or a pre-specified TIN size.

A number of procedures exist to construct TINs in this way, four methods in particular are examined by Lee [106] — the skeleton, filter, hierarchy and drop heuristic methods. The methods all select 'surface-significant' points from the grid DEMs to form the vertices and edges of the TIN but each differ in their definition of 'surface-significant'.

The skeleton method is a two phase approach of Fowler and Little (cited in [106]), which begins by selecting significant structural points to create an initial TIN. The TIN then has support points added to it, to obtain a surface approximation within the given tolerance bounds. The initial points are selected relative to their neighbouring points as being locally minima, maxima, potential ridge or channel points. The method is recognized as a complex approach, requiring line-thinning methods to reduce the points required to model ridge and channel lines. Significant storage is required. The skeleton method may also find peaks in flat areas where noise is present, and will select a large number of significant points in rugged areas.

The filter method of Chen and Guevara (cited in [106]) defines 'very important points' as those whose elevation when approximated by their eight grid neighbours are the most different from their actual elevation. The least significant points are then discarded in such a way that the process terminates when the point set remaining is the desired size or the surface approximation falls within the tolerance target. These points form the vertices of the TIN. The method visits each point only once and is therefore both fast and relatively uncomplicated in its approach. Only local information is considered and the method does not guarantee a fit to the terrain which is globally best.

The method of De Floriani *et al.* (cited in [106]), the hierarchy method, is a triangulation approach which uses triangular patches to approximate the terrain surface in successively finer detail. The method has the advantage of a hierarchical data structure and guarantees that a prespecified elevation precision will be met. The approach begins with the grid being subdivided into two triangles; these triangles are then recursively subdivided into three triangles on the point within the triangle which has the largest elevation difference between its actual elevation and that interpolated from the triangle. The recursion ends when the precision level is achieved or a TIN of required size exists. The main drawback of the hierarchy method is that it tends to create long-thin triangles.

The final method, which Lee [106] considers, is the drop heuristic method which Lee himself developed. This method is the only one of the four approaches which is based upon optimization. The objective is to determine the best set of points from the grid DEM to triangulate, so that the resulting TIN has the minimum elevation difference to the real surface. All the points are initially used to form the TIN with points then being dropped if they result in the least elevation difference when removed. This procedure continues, using the Delaunay triangulation to reconstruct the TIN after the deletion of the point, until the tolerance level or number of required points is achieved. As the method assesses all remaining points at each iteration, it is computationally expensive.

Lee tests the latter three methods against one another to assess their individual and relative performance. He concludes that the drop heuristic method generally performs better than the other two approaches, but has the greatest computational requirements. The filter method was the fastest approach of the three and the most suitable for local elevation changes, while the hierarchy method produced the most efficient data structure. Lee noted some distortion occurring in the TINs created by the hierarchy method due to its sampling approach. All methods resulted in better surface approximations when there was a greater number of points in the final TIN.

As a TIN is formed by a number of carefully selected coordinates, it is more economical in its representation than the large number of fine Rectangular Grids needed to accurately represent the same varying topography. This is due to the uniform sampling structure of the grids.

In a TIN the faces of the triangles give an approximate portrayal of the terrain features by capturing the slope of the terrain. The triangulation maintains the relational accuracy of the sampled points explicitly within the computational storage of the structure. Proximity information such as neighbouring triangles and vertices are recorded, allowing straightforward retrieval of this information for algorithmic queries. TINs therefore require greater storage requirements than the Rectangular Grids, whose proximity information is implicitly represented by their structure.

Jones *et al.* [94] introduce a storage scheme termed the Implicit TIN which is computationally more efficient than the traditional TIN storage structure, in that it stores only the vertices of the triangles and any linear constraints such as roads, ridge lines, fences or rivers. The authors apply the Implicit TIN to the application of a multiscale database where TINs of varying levels of detail are retrieved and reconstructed for specific queries. A spatially indexed, hierarchical quadtree structure is used to store the vertices and linear features. While the Implicit TIN structure is efficient in storage terms, there remains a significant computational effort to retrieve and reconstruct the TINs in response to each query. Hence, a trade-off between storage costs and retrieval time exists.

While computationally more complex to handle than Rectangular Grids, TINs have a number of advantages over such a matrix structure. One advantage is that the TIN model contains no vertical discontinuities between neighbouring triangles, and paths between edges and/or faces in the TIN can therefore be found without another level of approximation. TINs are also advantageous in that triangles, by their very nature, are more effective at representing non-horizontal planes occurring naturally in the terrain, than are flat grids. The TIN is able to model non-convex surfaces more realistically as "most interesting terrain is not convex, ... since it will usually have many mountains and valleys", [119, page 180].

DTMs allow data structures to be associated with each triangle face, recording terrain attribute information such as bush coverage, and natural or man-made features.

The accuracy of the TIN model can be improved by incorporating additional terrain phenomena. Laurini and Thompson [102, page 247] state that:

> "a more realistic representation will be achieved if the spatial data units recognize natural surface changes in slope, at peaks, pits, passes, ridge lines, saddle points and course lines or discontinuities, rather than just be fitted arbitrarily."

In particular, the explicit modelling of structural features such as roads, rivers, ridges, tramping tracks and fence lines can be represented. Such features, and changes in terrain morphology and vegetation, can be modelled in the TIN by means of a *constrained triangulation*. In a constrained triangulation these features are represented as edges of the triangles, with the triangulation being formed around these prescribed edges. Modelling such features by Rectangular Grids is not possible without extending the initial model [171].

A TIN provides a means of interpolating characteristics of those points not sampled by fitting a suitable linear or polynomial function to each triangle face. Hence discontinuities between triangle faces are easily incorporated. Goodchild and Lee [76] show that a TIN is considered a good model for interpolation as there is a uniquely appropriate way of fitting a plane to the triangle face, unlike a Rectangular Grid where uniqueness does not exist. It is also possible to string contours through the triangles based on spot height information and subsequent interpolation within each triangle, as outlined in Gold and Cormack [71] who achieve this by utilizing a contour tree. The tree is found by taking advantage of the structure of the triangulation and its spatial ordering properties which allow it to be traversed in a tree order.[3] The authors [71, page 148] state that "the intrinsic concept of ordering within a triangular network should be of value in many other applications."

Telcik [162] defines a *weighted-TIN* as a TIN which has weights, relating to the cost of traversal, assigned to each triangle and edge. Telcik uses this construct in solving the cross-country problem. The problem incorporates routing over both fixed paths (road networks), and non-fixed (off-road) paths, in order to obtain the total route of least cost. The use of a TIN in representing the terrain enables paths to be determined along the edges of the triangles (roads) and across the face of the triangles (off-road). The TIN is also advantageous in other practical aspects. This can be seen in Goodchild and Lee [76, page 178] who state that the TIN is "particularly suited to the problem of visibility coverage" in its surface representation and discrete set of surface peaks (triangle vertices).

A disadvantage of the TIN construct is that it may produce adjacent faces which

[3] Any triangulation can be represented as a binary tree when using a fixed viewpoint [71, 128].

are sharply angled. A large number of triangles (which are often distended) may also be required to model the transition between two areas of different relief. A TIN is also incapable of directly representing the presence of caves or overhangs in the terrain, due to its 2.5-D nature. To incorporate such features special extensions to the model would be required. However, caves are not found in predominance in many regions of New Zealand and if people were to go missing while caving then models of the caves would be utilized rather than the above-ground terrain. In general, overhangs are not significant in number or size as to warrant special inclusion. Hence, there is not a great loss of accuracy and usefulness of the DTM in excluding these particular features for the application of modelling the search terrain.

As Weibel and Heller [171, page 274] point out "no data structure is clearly superior for all tasks of digital terrain modelling." Nevertheless, in deciding upon a suitable physical model to represent the terrain of the search region, we have selected the geometrical partitioning of the TIN for the advantages outlined above. We consider that for this application the TIN is particularly suitable for the following reasons: its continuously connected triangle faces; realistic visualization; ability to include existing physical features; accurately model the variability of the terrain; efficiently interpolate data; and provide explicit proximity information in its computational structure.

A TIN representation is also attractive in its closeness to the visual guides presently used by searchers in the field. Searchers today currently navigate and plan their routes from both topographic maps of scale 1:50 000 and aerial photographs of the area. A TIN can be considered as supplying similar visual information as the aerial photographs but with more of the physical information found in a topographic map. A TIN is, however, only a computationally tractable approximation of both and Weibel and Heller [171, page 275] caution "that any product which is derived from a TIN ... will be heavily dependent upon the quality of the TIN."

To construct the TIN, which is the most suitable for the modelling of the search terrain, the most appropriate triangulation still needs to be determined. Laurini and Thompson [102, page 251] state that "a triangulation that produces Delaunay triangles is generally the preferred relatively straightforward method, producing triangles with a low variance in edge length" and Jones *et.al.* [94] consider it "common practice" to generate a Delaunay triangulation when no linear features exist. The following section outlines the Delaunay triangulation and its properties, describing why we feel that this particular triangulation best suits our purposes.

4.2.2 Delaunay Triangulation

The Delaunay triangulation is the straight line dual of the Voronoi diagram, assuming that no four points within the Voronoi diagram are cocircular. Under these conditions the triangulation produced is unique [16, 104] and non-degenerate [40]. The Voronoi diagram[4] is the locus of points which do not have a unique nearest neighbour [128]. Voronoi introduced the ideas behind the Delaunay triangulation for a lattice representation of sites in 1908, with Delaunay extending this in 1934 to sites which were irregularly positioned [7]. Both the Voronoi diagram and the Delaunay triangulation are tessellations; sets of non-overlapping polygons which cover a plane without gaps.

The Delaunay triangulation is planar and constructed such that every vertex within the triangulation represents a site in the Voronoi diagram and each triangle face corresponds to a vertex in the Voronoi diagram. The edges in the triangulation are orthogonal to their Voronoi edges but do not necessarily cross those edges [7, 128].

The Delaunay triangulation is of order $O(n)$ in the number of its triangles and edges, as implied by duality [104], and the triangulation consists of [94, 104]:

- $(3n - H - 3)$ edges; and
- $(2n - H - 2)$ triangle faces.

where H = the number of convex hull vertices and n = the number of vertices in total. The convex hull of all the Voronoi sites forms the convex hull of the Delaunay triangulation.

Lee and Schachter [104, page 237] state that:

> "The Voronoi and Delaunay tessellations have been extensively used to model spatial patterns in a wide range of fields including astronomy, biomathematics, computer science, geography, meteorology, metallurgy, numerical analysis, and packing and covering."

The Voronoi diagram represents the proximity of sites to one another; this is also translated to its dual. If two sites in the Voronoi diagram are closest neighbours (Thiessen neighbours), then they are represented as an edge in the Delaunay triangulation. Due to this property, a Minimum Spanning Tree over the Voronoi sites need only be constructed using edges from the Delaunay triangulation. Two other subgraphs of the Delaunay triangulation are the Relative Neighbourhood Graph and the Gabriel graph.

[4] A Voronoi diagram is also known as a Dirichlet tessellation or Blum's transform [7], and Voronoi regions are also referred to as Thiessen polygons, Wigner-Seitz zones [7], or 'proximal polygons' [140].

The Relative Neighbourhood Graph is the graph constructed by connecting two vertices by an edge, if and only if they are at least as close to one another as they are to any other point. The Gabriel graph is constructed by forming an edge between two vertices if a disk with the diameter of that edge contains no other vertices. This equates to a graph of the Delaunay edges which cross their dual Voronoi edges [7]. There is also evidence that suggests that the Delaunay triangulation in the plane is Hamiltonian with high probability [7], *i.e.*, there exists a cycle that passes through all its vertices. Such a cycle is of special interest to a SAR application if the vertices of the triangulation represent key points in the terrain, *eg.* advantageous viewpoints or huts.

In particular, Whitney shows that any four-connected triangulation[5] is Hamiltonian, while Kantabutra shows that a degenerate Delaunay triangulation[6] may not be Hamiltonian and Dillencourt shows that a non-degenerate, two-connected Delaunay triangulation may not be Hamiltonian [39]. Whether every three-connected Delaunay triangulation is Hamiltonian remains an open problem [39]. Dillencourt [40] also shows by example that, if a Delaunay triangulation is non-degenerate and Hamiltonian, the triangulation does not necessarily contain a minimal length Hamiltonian cycle. He shows that this is also true of a minimum weight triangulation.

Due to its dual relationship with the Voronoi diagram, the Delaunay triangulation possesses a number of interesting and useful properties. The most significant of these is that referred to as the "empty circle criterion" and is illustrated in Figure 4.2. This criterion states that a circle drawn through the two end vertices of any triangle edge will contain no other vertex within the triangulation. This is equivalent to stating that the circumcircle of any triangle in the triangulation will contain no other vertex. The criterion maximizes the smallest angle of any triangle, within that triangulation. A triangulation for which this holds true is defined as locally equiangular [147].

Sibson, in 1977 [147], showed that the Delaunay triangulation is the only triangulation which is locally equiangular; this is further shown to be equivalent to the triangulation being globally equiangular by Edelsbrunner in 1987 (cited in [7]). Jones *et al.* [94, page 43] describe this when they point out that the Delaunay triangulation "is characterized by providing the set of most equiangular triangles, a property which is desirable when interpolating within triangles."

Interpolation of the elevation value of a point falling within a triangle is determined from the elevation values of the vertices of that triangle. As "a Thiessen polygon can be used to define the region of influence of any point in an areal context", [94, page 43], its

[5] A four-connected triangulation is one where each vertex is connected to at least four other vertices in the triangulation by an edge.

[6] A Delaunay triangulation is non-degenerate if its Voronoi dual has no more that three Voronoi regions meeting at any point, otherwise it is degenerate.

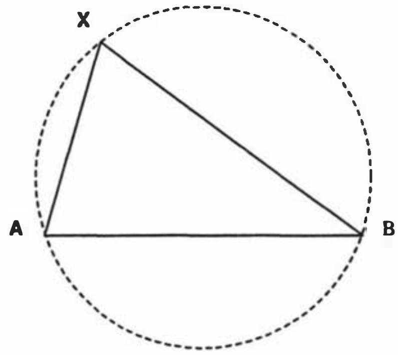

Figure 4.2: Empty circle criterion.

dual is useful for interpolation purposes. Under equiangularity the interpolated values are more likely to be closer to the true point values as the interpolation is computed over those points spatially closer in reality. Laurini and Thompson [102, page 250] state that for this reason it is "especially important to avoid long narrow triangles." By selecting the Delaunay triangulation, bounded errors on interpolation can be ensured [110, 140].

The Delaunay triangulation, by the property of equiangularity, avoids the creation of narrow triangles with extreme angles. The triangles are described by O'Rourke [128] as being "especially good partitions" for this reason. The Delaunay triangulation is described by such words as "stable" [71] and "unique" [16, 104]. If only traversal of the triangle edges is to be considered, the Delaunay triangulation does not minimize the total edge length but "it is however, close to optimal on the average," (Lingas, cited in [7, page 358]).

While the properties of the Delaunay triangulation, specifically the circle criterion, hold in 2-D, they do not necessarily transfer to 2.5-D, as the triangulation is calculated only on the consideration of the (x,y) values of the terrain points. Hence the maximization of the minimum angle may not be achieved for the spatial triangle when triangle edge lengths are determined under the 3-D Euclidean distance measure and the equiangularity of the triangles may not hold. Additionally, the adjacency information stored within the triangulation is made partially redundant. This is seen in the specification of closest neighbours (as measured by shortest paths) as traversal time is dependent not only on physical distance but also on the terrain type, and other factors such as the prevailing weather. However, despite the increase in dimension, the Delaunay triangulation does retain an important quality as defined by a measure of "roughness".

The Sobolev seminorm (cited in [7]) is used as a measure of roughness for a triangulation performed over a set of points with (x,y) coordinates and an associated height. The seminorm is defined as:

$$\sum_{\Delta \epsilon T} | \Delta | (\alpha_{\Delta}^2 + \beta_{\Delta}^2)$$

where

$| \Delta |$ is the area of the spatial triangle Δ

α_{Δ}, β_{Δ} are the slopes of the plane containing Δ

(The spatial triangle is the triangular surface in 2.5 D defined by the triangulation.)

Rippa in 1990 (cited in [7]) showed that this measure is minimized for the Delaunay triangulation. Aurenhammer states:

> "This result is somewhat surprising since the Delaunay triangulation — although itself clearly being independent of the height at each site — optimizes a quantity that depends on these heights" [7, page 374].

It is for this reason that the Delaunay triangulation is especially well-suited for use in terrain modelling and TIN construction in particular.

As a TIN can be represented by a number of triangulations, one triangulation must be chosen and used consistently — we have selected the Delaunay for the reasons outlined above and because it is widely used and accepted for terrain modelling.

The actual approach required in modelling a real terrain would require the utilization of a method that allows for alterations to any initial TIN representation, to ensure that the surface fit is within a pre-specified tolerance level. This in itself is a greater research area which falls outside of the scope of this thesis. (It is our intention that all heuristic methods developed within this thesis are applicable to a TIN constructed using any triangulation, and are not specifically constrained to the Delaunay triangulation in particular, although some alterations to storage structures may be necessary).

Lee and Schachter [104] consider an application to visual flight simulators which utilizes a piecewise planar surface terrain model. The authors [104, page 220] conclude that the Delaunay triangulation is "an excellent choice" for "fitting triangular faceted surfaces to digital terrain data" as it has good visualization capabilities due to the minimum angle of the triangulation being maximized, and the triangulation is computationally efficient to construct.

In conclusion, the Delaunay triangulation is particularly suitable for the application of modelling terrain as it: produces triangles which are generally more representative of the actual topography, produces a good visualization of the land surface when linked by computer graphics and provides greater accuracy in interpolation.

4.2.3 The Constrained Delaunay Triangulation

As introduced previously, a constrained triangulation is one formed upon a number of prescribed edges which represent linear structures. These structures are generally physical but may also include non-physical boundaries such as political electorates. These linear features become edges in the triangulation with the remaining triangle edges being formed around them in such a way as to construct a triangulation. A constrained Delaunay triangulation[7] is one which is formed by adding edges to the prescribed edges, so as to obtain a triangulation which matches as closely as possible to the Delaunay criteria. An example of such a triangulation is illustrated in Figure 4.3 (taken from [16]). Chew [16, page 100] states that "a constrained Delaunay triangulation provides a natural way to retain the boundary information while producing a good triangulation."

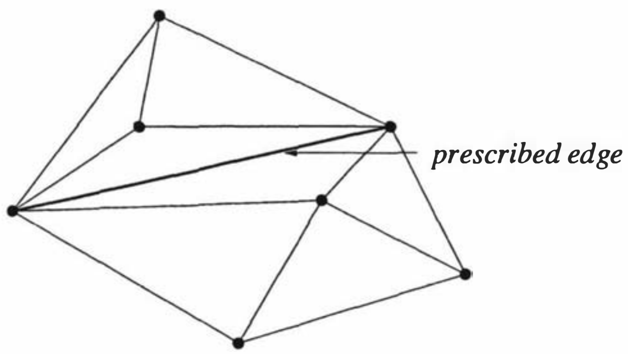

Figure 4.3: Constrained Delaunay triangulation.

In the constrained Delaunay triangulation, the empty circle criterion is relaxed to the criterion that a circle drawn through the two end vertices of a non-prescribed edge will only contain a vertex if a line, drawn from that interior vertex to at least one of the end vertices, crosses a prespecified edge [16]. Lee and Lin (cited in [7]) show that a constrained Delaunay triangulation still possesses the property of equiangularity.

While the constrained Delaunay triangulation is no longer the straight-line dual of the Voronoi diagram, its dual is a type of Voronoi diagram which may overlap itself [16]. It also exibits similar spatial adjacency properties and has the Constrained Minimum Spanning Tree as a subgraph. The Constrained Minimum Spanning Tree is analogous to its unconstrained companion but contains a set of prespecified edges which may not be crossed by any other edges in the tree.

[7] A constrained Delaunay triangulation is also known as an obstacle triangulation or a generalized Delaunay triangulation [16].

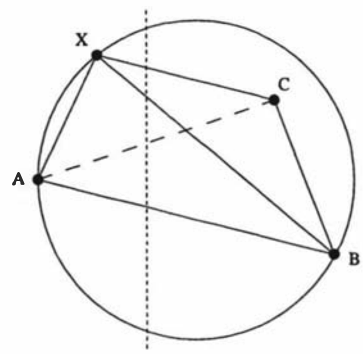

Figure 4.4: Lawson's local optimization procedure.

4.2.4 Construction Algorithms for Delaunay Triangulations

We now consider a selection of construction algorithms for the Delaunay triangulation — both the constrained and unconstrained versions.

4.2.4.1 Unconstrained Delaunay Triangulations

Lee and Schachter [104] present two construction algorithms for the Delaunay triangulation. The first is a divide-and-conquer technique with the asymptotically optimal running time of $O(n\log n)$. The algorithm sorts the vertices in ascending order of (x,y) coordinates. The vertices are then divided into two subsets and their Delaunay triangulations are constructed via the empty circle criterion. The triangulations are then merged using the common lower tangent of the two.

The second algorithm has a worst case running time of $O(n^2)$, although the authors state that it is comparable to the first algorithm in its average case performance. The algorithm is simpler in approach to the first, but is limited to a rectangular region. Here the Delaunay triangulation is constructed iteratively by first partitioning the rectangle into bins, ordering the vertices within each bin and then inserting each vertex into the triangulation, based upon this ordering. The insertion process utilizes Lawson's local optimization procedure (cited in [104]) to ensure that the resulting triangulation is Delaunay. This procedure swaps the diagonal of any quadrilateral in the triangulation if by doing so the minimum angle is maximized. This is accomplished by determining the circumcircle of one of the triangles forming the quadrilateral. If the fourth vertex of the quadrilateral falls inside the circle, the diagonal of the quadrilateral is swapped with its opposite diagonal[8] as illustrated in Figure 4.4.

To globally meet this criterion, the diagonal swapping process may spread outwards

[8] Only required exchanges are executed to avoid possible cycling at degenerate instances [147].

over the triangulation, but the process will always terminate (as shown by Sibson [147]). Lee and Schachter consider that this iterative algorithm is especially suited to the application of terrain fitting as additional data points can be easily incorporated to create a surface fit closer to the actual terrain surface.

Lee and Schachter's iterative method is extended to an arbitrary convex region by Macedonio and Pareschi [110], who explore the utilization of a Delaunay triangulation applied to terrain modelling, especially the applications of approximating the volume of a given surface, and the interpolation of elevation data. They use, as their inputs, irregularly placed points with corresponding deposit thickness or elevation values. The authors' iterative approach to constructing the initial triangulation departs from Lee and Schachter's algorithm only in their initial step, where they first construct the convex hull of the region. All remaining points are iteratively added to the triangulation based on the bin ordering approach of Lee and Schachter.

4.2.4.2 Constrained Delaunay Triangulations

Chew [16] presents an $O(n \log n)$ divide-and-conquer algorithm to construct the constrained Delaunay triangulation upon a rectangular region. This region is vertically divided into strips with the constrained Delaunay triangulation being found in each strip. Adjacent strips are then linked together and the process is repeated until the entire region is triangulated.

An alternative $O(n \log n)$ algorithm is proposed by Jones *et al.* [94] who first find a Delaunay triangulation over all the vertices and then insert the prescribed edges. This insertion procedure is carried out by deleting the edges of the triangulation which intersect the prescribed edges, and then retriangulating the two polygons which are subsequently formed on each side of the those edges.

Lee and Schachter [104] propose a method to determine a Delaunay triangulation from a planar set of points and line segments (where the set of line segments is sparse) for application to terrain fitting. The point set is defined as the set of local extrema of the terrain surface and the endpoints of ridge segments. The method ensures that the local extrema are vertices of the final triangulation and that the ridge segments are represented by edges of the triangulation. For any point in this set that lies within the smallest circumscribing circle of a ridge segment, a normal projection is constructed onto this segment, generating a projection point set. These projection points are added to the initial point set and a Delaunay triangulation is constructed over the point set. The terrain surface is represented by an elevation value being associated with each of the points in the triangulation.

4.2.4.3 The Selected Construction Algorithm

Within this thesis, artificial computer-generated topography data is utilized which contains characteristics of possible real terrain scenarios. The choice of using artificial topographies over real terrains was made, in part, due to the large variability of topography throughout New Zealand, even within particular regions. By choosing to perform experimentation over terrains which have been generated in a controlled way, problem characteristics relating to different terrain types can be isolated in such a way that the results obtained are more meaningful. Furthermore, in utilizing artificial topographies we are able to exploit the most desirable characteristics of both the unconstrained and constrained Delaunay triangulations by assigning specific linear features to a selection of triangle edges in the final unconstrained Delaunay triangulation, thus obtaining good terrain representation with inclusion of such features.

We have chosen to construct the TINs of search terrains using the Delaunay triangulation. The particular construction algorithm we follow is the algorithm of Macedonio and Pareschi.[9] This approach was selected as it lends itself well to terrain modelling as shown by both Lee and Schachter and the applications presented by Macedonio and Pareschi. As the TIN is constructed only once at the outset of the modelling process, there do not exist any pressing time factors which would necessitate the preference of an $O(n\log n)$ algorithm and in any case the selected algorithm is deemed as performing competitively in most instances. Macedonio and Pareschi provide their C code listings and it is these which we have utilized along with their computer storage representation. We have implemented additional array representations for those topographic features not considered by the authors.

We forsee that in application to SAR operations, TINs would be constructed for all land surfaces falling within each SAR team's assigned area, ahead of any search being instigated.

4.2.5 Projection to a Planar Graph

A TIN can be easily and naturally transformed into a planar (triangulated) digraph by depicting the vertices of each triangle as the vertices of the graph, and the triangle edges as the connecting directed edges. We define the resulting digraph as the *Search Region Graph*. Instead of regarding the edge costs as isotropic we have developed a more realistic costing which reflects the height differentials and varying terrain difficulties present.[10] Hence, an edge cost is determined in both the uphill and downhill directions for each

[9] We assume that the search region can be defined as a convex area.

[10] The costing process is outlined further into this chapter.

directed edge, giving the required digraph property. The triangular regions defined by the edges are also assigned traversal costs, defining a weighted-region graph, similar to those addressed by Mitchell [119] and Telcik [162].

4.3 The Topography

The features of the search terrain are represented on the TIN by a classification structure on the edges and faces of each triangle. In our model the vertices of the convex hull of the triangulation are assumed to be accessible by 4-wheel drive vehicle.

Each vertex within the triangulation has a height characteristic by definition of its corresponding elevation (z-coordinate). To realistically represent the terrain, the triangle faces are assigned density classifications categorizing the average bush cover over the terrain represented by that face. The following classification system is employed:

1. no bush cover
2. sparse bush cover
3. moderate bush cover
4. thick, tangled bush cover or thick scrub

As we are concerned only with computer generated topographies, we have assigned each edge a specific feature type, rather than employ the constrained Delaunay triangulation to explicitly model the natural and man-made features. Each edge is identified as either a formed track; a ridge; a stream bed or river;[11] or, if the edge does not fall within any of these categories, the edge is classified as being untracked terrain.

Additional classifications are added to represent the level of traversal difficulty of each edge type. The classification system parallels the bush cover classification system applied to the triangle faces, such that a classification of 2 under either system defines a similar level of traversal difficulty.

1. a well-formed track
2. a moderately upkept track or a predominantly shingle stream bed
3. an overgrown track or a rocky stream bed
4. untracked or a river

[11] We assume that stream beds are traversable in their length, and that both streams and rivers are crossable in good to moderate weather if there has not recently been any heavy rainfall.

Each untracked and ridge edge is assigned a density classification which is the average of the density classifications of its two adjacent triangle faces. The bushline is defined at 1200–1300m to mimic the New Zealand bush conditions [75]. The TIN generation process and classification distributions employed in creating a test terrain environment are described fully in Chapter 12.

4.4 Cost Structure

The cost of traversal over any sector of the TIN can be approximated by the height differentials, spatial distances, terrain density, and weather and light factors present.

Telcik [162], in the costing of a weighted-TIN for the cross-country problem, attempts to incorporate the physical terrain characteristics represented by each triangle and their respective traversal difficulty. He uses the simplification that traversal in either an uphill or downhill direction is identical. Allowances are made however to represent impassable regions as obstacles, with a prohibitively high cost. This classification is also assigned to areas with a gradient too steep for the vehicles to handle easily.

In an attempt to be as realistic as possible in assigning cost structure, we employ a more terrain-dependant cost structure, foregoing Telcik's simplification of identical uphill and downhill traversal times. We consulted current New Zealand texts on bushcraft and outdoor skills for realistic guides to walking times for this country's conditions. The New Zealand Mountain Safety Council [75] gives the following approximate traversal times for a party of average fitness and experience who are each carrying full packs:

- 4 to 5km per hour on a smooth wide track;
- 2km per hour on a rough track;
- 300m height gain per hour when walking uphill; and
- 400m height loss when going downhill.

The Council also indicates that longer times will be recorded for: larger parties navigating obstacles, the use of sidle tracks[12] over ridge or valley tracks, and the navigation of overgrown tracks.

Graeme Dingle [41], a well known New Zealand outdoor adventurer, gives the following rough guide for a tramper in New Zealand bush country conditions:

- speed up to 5km per hour on a good track and relatively level ground;

[12] Tracks which wind around.

- speed up to 3km per hour on untracked, but open and relatively flat bush;
- speed up to 600–1,000m per hour in thick, tangled bush;
- 300m height gain per hour of ascent; and
- 600m height loss per hour of descent.

Dingle discounts the British use of the well known Naismith's rule in calculating travel times, as "generally inaccurate in New Zealand bush". He qualifies the above guide by stating that "there is no end to differences in rates between trampers of various levels of fitness and determination!"

Searchers themselves will generally have fitness levels above average; however, as we are not explicitly regarding the factor of fatigue in this initial model, we will base their traversal times on those of the average tramper. For the purposes of the model we amalgamate the times given in the two frameworks, with some factorization of ascent and descent values.

- Terrain density measure of 1:
  - 5km per hour on relatively level ground
  - 400m per hour of ascent
  - 600m per hour of descent
- Terrain density measure of 2:
  - 3km per hour on relatively level ground
  - 300m per hour of ascent
  - 500m per hour of descent
- Terrain density measure of 3:
  - 2km per hour on relatively level ground
  - 200m per hour of ascent
  - 400m per hour of descent
- Terrain density measure of 4:
  - 0.6km per hour on relatively level ground
  - 100m per hour of ascent

– 300m per hour of descent

We define "relatively level ground" as less than 500m ascent or descent over 10km.

These traversal speeds are used when searchers are moving from one point to another and not searching at the same time. We refer to such traversal speeds as 'access only speeds' or, alternatively, 'deadheading speeds', in-line with arc routing literature which refers to an arc being 'deadheaded' if it is traversed without service. Research by Perkins and Roberts [137] has shown that searchers who are physically visually searching will take **twice as long** to traverse the same distance than if they were not searching. (This equates to a POD rate of 75–80% for experienced searchers who are spaced at critical separation and are utilizing the purposeful wandering technique.) Hence we model traversal speeds over the TIN for resources who are actively searching as half that of the above speeds. We also assume the same search speeds regardless of the search method in use. These speeds do not have to be maintained over long intervals as resources are rested regularly. It is envisioned that, for application to real search scenarios, search speeds would be found by field tests for particular search regions.

4.4.1 Forbidden Regions

Any edge in the TIN which intersects or follows a non-traversable natural formation, such as a deep river or swamp, can be represented as a forbidden edge. This is realized by assigning a very large traversal time (∞) to all such edges, ensuring that they are never chosen as part of a feasible path.[13] Terrain types requiring specialist skills for traversal may also be represented as regions forbidden to those searchers lacking the required skill attribute.

4.5 Visibility and Sound Measures

Visibility is a key element when attempting to model terrain. The modelling problem addresses such questions as; 'when located at a particular point, how much of the surrounding area is visible from that point?' The accuracy of the response is directly related to the accuracy of the terrain model employed. Visibility is defined by Goodchild and Lee [105, page 413] as "the location and size of area which can be seen from any given viewpoint." However, Goodchild and Lee caution that no precise definition of visibility exists for real terrain due to the surface involved and the presence of obstacles such as trees. They also state that errors in elevation will result in large errors in visibility.

[13] Alternatively, these edges could be deleted from the graph entirely if only edge traversal of the TIN was being considered.

Goodchild and Lee [76] consider the aspect of visibility when employing a TIN representation of the terrain to identify suitable locations for such utilities as fire towers, used to monitor forests. The authors utilize the approximation that the surface within each triangle face can be represented by a plane passing through its three vertices. Visibility is modelled as a property of each triangle and Goodchild and Lee consider a triangle to be visible from a point on the TIN if and only if its three edges are fully visible. An edge is defined as being visible from the viewpoint if a line connecting the two vertices of that edge is totally visible. Vertices are visible to one another (inter-visible) if a line connecting them "lies entirely on or above the surface", [76, page 178]. The visibility function is a Boolean function and the triangles adjacent to the viewpoint are defined as visible.

Goodchild and Lee [76] present an algorithm to determine the subset of triangles visible from each vertex of the TIN with a worst case scenario of the complexity $O(n^3)$. "The algorithm is executed fully for every vertex, as there appear to be no simple theorems which would allow information about one vertex's visible area to be derived from another", [76, page 180]. Lee [105] later modifies this visibility algorithm by simplifying it and increasing the computation speed required to reach a solution. The modified algorithm sorts all edges of the TIN in ascending order of their distance from a given viewpoint. Each edge is then tested for its visibility from the viewpoint by calculating whether the edge is 'blocked' by any other edge positioned closer to the viewpoint. A visibility matrix between triangles and viewpoints is then obtained.

We consider a simplified visibility model which is not as dependent upon the elevation accuracy of the TIN model and which takes into consideration the vegetation of the terrain. We model the visibility over the terrain as a characteristic of the triangles and edges of the TIN, and directly correlate visibiity to the terrain density classifications of these entities. We utilize Perkins' [137] definition of the "limit of visibility" to define a *visibility measure* for a ground search resource. The visibility measure quantifies the distance to the limit of visibility under differing terrain and search conditions. Hence Perkins' critical separation distance can be expressed as *2* × *visibility measure*. We also extend the definition to include a measurement of how far sound travels through differing terrain in an analogous fashion, defining this measurement as a *sound measure*.

In particular we stipulate that a searcher positioned on an edge can not see beyond that edge, unless they are situated on one of its end vertices. In this position the searcher can see down any adjacent edges, to the limit of the visibility associated with each edge. The visibility of the searcher on an edge is also restricted in that he can only see into the triangles adjacent to his current position, to the limit of the visibility measure of those triangles.

A searcher positioned on the face of a triangle can see only across the face of that triangle to the limit of the visibility measure of that triangle, *ie.*, the searcher cannot see into an adjacent triangle. We justify this assumption by proposing that in reality the angle between adjacent faces may often be convex, preventing visibility into the adjacent triangle face. Similarly, for simplicity we assume that a searcher can not hear sounds beyond the edge or triangle they are positioned upon.

While true 3-dimensional visibility is not modelled, the restrictions were considered to be reasonable by those experienced in search and rescue operations [81]. This is because searchers are generally assigned to search only the terrain closest to their position and, as such, their concentration of view is on this region. Additionally, there exist many cases where those being searched for may only be visible from a close proximity, especially if unconscious, injured or hostile weather conditions prevail.

Visibility and sound measures are assumed to be constant throughout any triangle or edge, and all searchers are assumed to have identical visual and hearing capabilities. As terrain density classifications for edges are defined with consideration to the two adjacent triangles, sudden changes from very high visibility measures to very low visibility measures at the boundaries of the triangles are eased.

Wartes' (cited in [137]) field trials in moderately dense underbrush give a 50% POD level when searchers are spaced at 100 feet (30m). Utilizing the theoretical 50% POD level attributed to critical separation by Perkins [137] we can conclude that the visibility measure for this terrain is approximately 15m. Goodman and Cowan [78] give values of critical separation over three differing classifications of search terrain. These result in a visibility measure of 25.4m for easy search terrain, 12.7m for moderate search terrain and 7.11m for difficult terrain. No specific descriptions of these terrain types are provided.

Colwell's [21] field experiments in dense coniferous forest in summer for a standard visibility sweep give a spacing of 47.2m for a 50% POD level. This equates with a visibility measure of 23.6m. A sound sweep conducted over this terrain resulted in a spacing of 151m for the same POD level indicating a sound measure of 75.5m, approximately three times the visibility measure.

Colwell [25] notes that while performing a sound sweep search "there is a high probability of searchers finding a missing person at three to four times the searcher spacing required for a normal, visual grid search", however we have chosen to err on the side of caution and factor the sound measures at three times that of the visibility measures. Colwell [21] also conducts sound sweep experiments which compare detection by a normal and quiet voice response. Examining the results, the spacing to detect a quiet voice response is, on average, a factor of 0.548 that of the spacing to detect a normal voice response. We utilize these factors to approximate sound measures for each response category.

To determine visibility measures for each of the four vegetation types in the model, we infer distances from these international field data measurements, as we were unable to find data specific to New Zealand conditions.

The visibility and sound measures utilized in this study are detailed in Table 4.1. We define these measurements as those holding for ideal search conditions and for standard visibility sweep searches.[14] In real application, tests would be conducted in the field to determine accurate visibility and sound measurements for the specific terrain at hand over a variety of environmental conditions.

Table 4.1: Visibility and sound measures.

	Terrain Density Classification			
Detection Method	1	2	3	4
Visual detection	50m	25m	15m	5m
Sound detection-normal voice response	150m	75m	45m	15m
Sound detection-quiet voice response	82m	41m	25m	8m

4.6 Weather and Light Conditions

Under New Zealand bush conditions "the weather can change dramatically in just a few minutes" and "wind and cloud conditions can change dramatically in just a few metres across an exposed windswept ridge", [75, pages 34–35]. In particular, weather in the high country of New Zealand has its own peculiarities — often covered by cloud, and experiencing hail, thunderstorms, snowfalls and blizzards. The New Zealand Mountain Safety Council [75] give the air temperature at the bushline to be about 8°C colder than that at the sea level, stating that the air temperature drops by 6.5°C–10°C per 1000m altitude. The snowline in winter will generally occur at an altitude of 1000–1500m [75].

To model the weather conditions and light levels under which the search is being conducted, we implement the following four descriptive levels:

- level 1 — perfect weather and light conditions
- level 2 — drizzling rain
- level 3 — steady rain, blustery conditions or snow

[14] If the subject is unconscious or a fatality, a low visibility sweep or body sweep, at a smaller spacing would be necessary to achieve a 50% POD level. From Colwell's data 50% POD is achieved at a spacing of 37.1m (a factor of 0.8 of a standard sweep) for a low visibility sweep and at 40.9m (a factor of 0.9 of a standard sweep) for a body sweep [21].

- level 4 — storm conditions

The level of visibility is dynamic in nature — varying in dawn, dusk and night conditions, and decreasing with deteriorating weather conditions. To model this dynamic property of visibility we develop a visibility factor which correlates to the present weather and light conditions in the following way:

- level 1 — visibility factor 1.0
- level 2 — visibility factor 0.8
- level 3 — visibility factor 0.6
- level 4 — visibility factor 0.4

In the previous section, the visibility measure is defined as the distance a searcher can see under perfect conditions for each terrain type. These conditions relate to a visibility factor of 1. Visibility factors for all other weather and light levels are then fractions of this measure. For example, a searcher moving in steady rain (level 3) through terrain of density classification 2 (Table 4.1), will have their visibility measure of 25m updated to

$$0.6 \times 25 = 15m$$

which equates to 60% of the visibility measure under perfect conditions.

Light conditions throughout dawn and dusk are modelled as being one level more difficult than the current visibility conditions, and visibility at night is considered to be one level more difficult again. Sound measures are also assumed to decrease with changing weather and light conditions in proportion to visibility measures. When visibility measures are altered, sound measures are adjusted to measure three times the visibility measure for a normal voice response and three times 0.548 times the visibility measure for a quiet voice response. If a resource is searching a region when a visibility change occurs the distance at which the team of searchers is spaced must also be updated to achieve the same level of POD.

Mitchell considers the uncertainty prevalent in terrain models with respect to traversability, for example, changes due to weather conditions. Mitchell [119, page 197] proposes that it should be possible to update traversal probabilities in the light of known information:

> "A truly 'intelligent' planner should have information about the kinds of circumstances which potentially affect different types of terrain and be able to reason about it."

With decreasing weather and light conditions, traversal times are affected as searchers' visual capability is decreased and the ground beneath becomes more difficult to move over and through, creating conditions which are not only more physically demanding but also more mentally demanding. Within the model we utilize a traversal factor, analogous to the visibility factor, to update traversal times. These factors are arbitrarily defined as:

- improvement of weather and/or light 1 level — traversal factor 0.8
- decrement of weather and/or light 1 level — traversal factor 1.25
- decrement of weather and/or light 2 levels — traversal factor 1.5
- decrement of weather and/or light 3 levels — traversal factor 1.75

To update traversal data for the new conditions, the appropriate traversal factor is multiplied to the current traversal times. Within the simulation model the number of levels by which the weather or light alters at any stage is only one level. However, as the traversal costs input into the simulation are expressed for ideal conditions (visibility level 1), if the starting weather level is level 4 a decrement of 3 levels is observed. This is the only time where an instantaneous level change greater than one occurs.

When level 4 weather conditions are experienced, some presently traversable regions may become flooded, snow-bound, or impassable for some other reason such as a fallen tree. In such instances these regions will have their traversal cost updated to infinity (∞), to ensure that no searchers are assigned to these regions as part of any feasible search pattern. We assume in the model that only edge regions representing water features may become flooded. In these conditions the level of urgency of the search is greater, as the risk of exhaustion, hypothermia and loss of life increases. Such conditions may require the withdrawal of searchers from the search region for their personal safety. Within the simulation model, we arbitrarily reverse the status of flooded regions to being traversable again, three days after an improvement in weather conditions (to level 3) occurs.

4.7 Search Regions

We consider each edge and triangle of the TIN as an individual search region, except for those edges which are typed as untracked. As these edges represent boundaries between triangles we do not view them as linear features requiring searching. The number of search regions is then given by $5n - 2H - 5 - untracked\_edges$, where $n =$ the total number of vertices of the TIN and $H =$ the number of convex hull vertices.

From this point we will refer to each search region as either an edge region or a triangular region.

Having developed a model of the search terrain we now model the path taken by the subject over this terrain, including a model of clue placement. The visibility model is expanded upon to define a model for the detection of both the subject and the clues formed by him, when searching is conducted by human ground resources. An automated approach to the update of POA values when clues are detected is also developed.

CHAPTER 5

Detection And Clue Modelling

5.1 Detection Models

Frost [62] notes that "almost all detection profiles are at their highest close to the searcher's actual track and decline in some fashion as distance from the searcher's track increases". We now examine a selection of lateral range curves for visual search resources with the intention of developing a visual detection model to simulate detection in a land SAR operation.

5.1.1 The Definite Range Detection Model

The *definite range* model of detection or 'clean sweep' [100] is the simplest detection model and is defined as:

$$\begin{aligned} p(x) &= 1.0 \quad -r_{def} \leq x \leq +r_{def} \\ p(x) &= 0.0 \quad \text{otherwise} \end{aligned}$$

where $p(x)$ is the probability of detecting a search object located at a distance x from the sensor and r_{def} = the definite range over which the sensor can detect. The lateral range curve for the definite range detection model is depicted in Figure 5.1 (adapted from [150, Figure 4-5]).

For this detection model, r_{def} is equal to r_{max}, the maximum range of detection, and the sweep width, $W = 2r_{max}$. While the model itself is not realistic for most sensors, theoretically the definite range model is a useful model of detection as it defines an upper bound on the probability of detection and provides an alternative definition of sweep width which can be defined as:

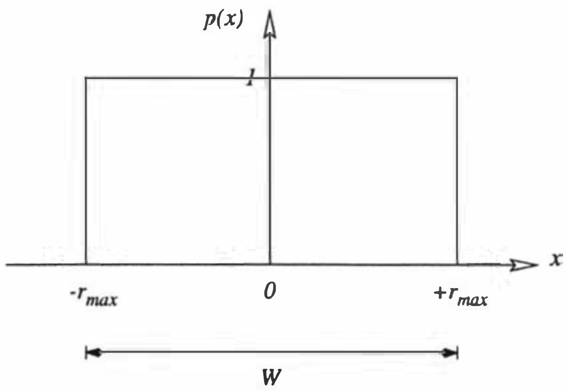

Figure 5.1: Lateral range curve of the definite range detection model.

> "the width a definite range sensor would have to sweep in order to detect the same number of objects per unit time in a uniform distribution of search objects" [150, page 4-5].

The definite range model is a special case of the *M-Beta* detection model, which is described below.

5.1.2 M-Beta Detection Model

The *M-Beta* detection model is defined as:

$$p(x) = M \quad -r_{max} \leq x \leq +r_{max}$$
$$p(x) = 0 \quad \text{otherwise}$$

where $r_{max} = \beta/2$. The lateral range curve is depicted in Figure 5.2 (adapted from [150, Figure 4-7]).

The sweep width is equal to:

$$W = M \times \beta$$

5.1.3 Inverse Cube Detection Model

The *inverse cube* model is an empirical model developed by Koopman during the Second World War for the US Navy, to visually detect enemy warships from an aircraft by sighting the ship's wake [100, 150].

The instantaneous probability of detection, γ, is approximated by

$$\gamma = \frac{kh}{r^3}$$

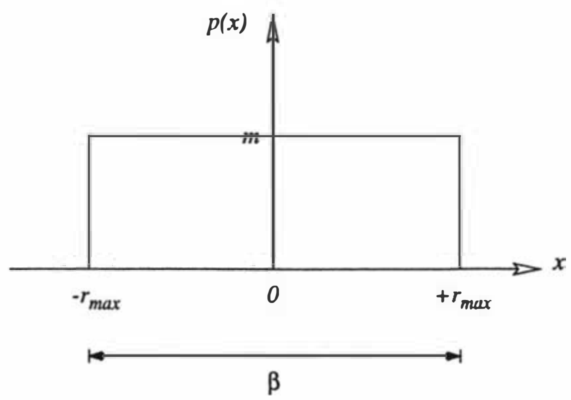

Figure 5.2: Lateral range curve of the $M - Beta$ detection model.

where k is a proportionality constant consisting of search factors and an area dimension, h is the height that the searching aircraft is flying above the ocean, and r is equal to the horizontal range between the aircraft and the ship's wake.

The lateral range function is given as:

$$p(x) = 1 - e^{\frac{-2kh}{wx^2}}$$

where w is the relative speed, or in terms of sweep width (W) as:

$$p(x) = 1 - e^{-\frac{W^2}{4\pi x^2}}$$

The graph of the lateral range curve in ideal search conditions is illustrated in Figure 5.3 (adapted from [150, Figure 4-10]). In theory the maximum detection range of the inverse cube model is infinite.

Frost cites the main advantage of the inverse cube detection model in modelling human sensors in marine SAR is that "it is based on a physical model, however crude or inaccurate, of visual detection under operational conditions" [150, page 4-11]. The inverse cube model has been used as the basis of detection calculations in marine SAR planning for over fifty years, even though not all the conditions under which the model was developed held for those searches and the model has never been verified. Frost also notes that "under ideal conditions involving parallel sweep searches for typical SAR search objects of moderate to large size, the inverse cube model often predicts the POD surprisingly well. However, for poor conditions or small search objects, the inverse cube model is a poor predictor of POD, being generally too optimistic" [150, page 4-12].

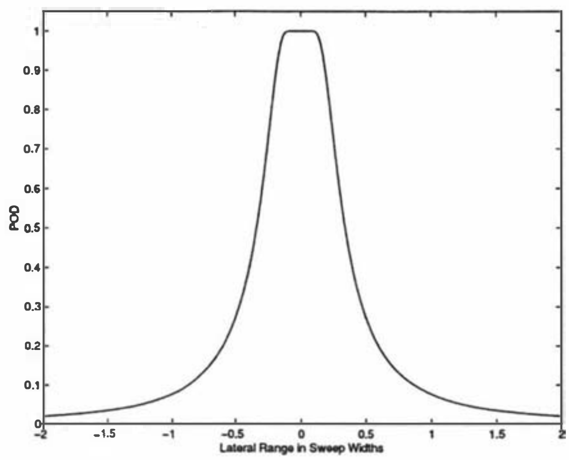

Figure 5.3: Lateral range curve of the inverse cube model of detection under ideal search conditions.

5.1.4 Random Search

The *random search formula*, or the *exponential detection function*, was also developed by Koopman [100]. The random search formula defines a lower bound on the probability of detection, and is used in this capacity to predict the POD of search resources in poor conditions by the marine search planning method, ISPM [150]. (The ISPM recommends the use of the inverse cube model under ideal search conditions.) The POD of a random search is derived to be:

$$POD = 1 - (1 - \frac{WL}{nA})^n \quad \approx \quad 1 - e^{-\frac{WL}{A}} \text{ when } n \text{ is large}$$

where L represents the length of the sensor's path, A is the size of the search area, and n is the number of equal length portions which the sensor's path is divided into.

Hence $POD = 1 - e^{-C}$ when n is large, with $C = \frac{WL}{A}$ being constant and consistent with Section 2.7.

Random search assumes "that search effort is randomly distributed over the search area according to a uniform distribution and that each small increment of search effort is positioned independently of the past search effort" [153, page 218].

Additional assumptions behind this derivation include: that the search object is stationary and has a uniformly distributed location probability density; that detection occurs within the definite range of W/2 either side of the sensor's path; and that the length of any portion of the sensor's path is small relative to the total path length and "decidedly larger" than the detection range [150]. Hence, the probability of non-detection over the total path is the sum of non-detection in each of the n independent path portions ($(1 - \frac{WL}{nA})^n$).

Stone [153] shows that these assumptions differ from a sensor following a path determined by a random walk. Following a random walk may give a lower POD than the random search formula.

Stone also notes that, although no search situation usually meets these underlying assumptions, the random search formula "provides a reasonable and conservative estimate for the detection function for a wide class of searches" [153, page 218]. However, he shows that when the sweep width is uncertain this is no longer the case and the sensor's detection ability is not approximated well by the exponential detection function.

Due to its dependence on C, the random search formula is "valid for virtually any reasonable detection model and certainly for any where $p(x)$ is maximum at $x = 0$ and decreases monotonically as $|x|$ increases" [150, page 5-5]. Different detection models with a uniform effort distribution hence perform equally for equal values of C "whenever significant random variations in the search parameters are present" [65].

Randomness in a search operation can be caused by either random behaviour by the search subject, navigational error, or stochastic search factors such as weather changes. Frost notes that "almost all humanly planned endeavours are subject to unavoidable random factors and are rarely completed exactly as intended" [150, page 5-3].

5.1.5 Comparison of Classical Detection Models

For *M-Beta* models the POD value for a parallel sweep can be computed as

$$\begin{aligned} POD &= C & \text{if } C < M \\ POD &= 1 - (1 - M)^{\frac{C}{M}} & \text{if } C \geq M \end{aligned}$$

For the inverse cube model the POD for a parallel sweep search is determined by

$$POD = erf(\frac{\sqrt{\pi}}{2}C)$$

where erf is the error function.

When the different detection models are graphed displaying POD against coverage, for parallel sweep searches, Figure 5.4 is obtained.

It can be seen that the definite range detection model forms an upper bound while the random search model forms a lower bound. Koopman observes that "all actual situations can be regarded as leading to intermediate curves" where "the inverse cube law is close to a middle case, a circumstance which indicates its frequent empirical use, even in cases where the special assumptions upon which its derivation was based are largely rejected" [100, page 79]. Positioning sensor tracks closer than W provides no additional benefit for a definite range sensor as the maximum POD level has already been reached. As the

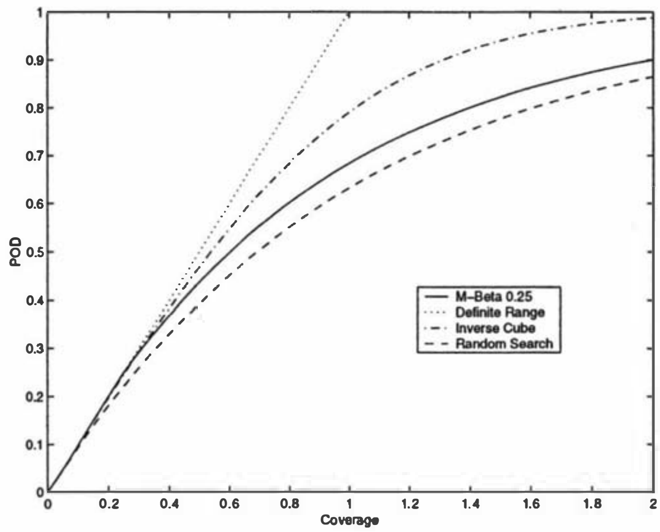

Figure 5.4: POD vs. coverage comparison of classical detection models.

value of M decreases for $M - Beta$ models, efficiency is lost and the curve approaches that of the random search curve [150].

"The amount of search effort required to achieve a certain POD with a parallel sweep search is highly dependent upon, and sensitive to, the nature of the sensor's lateral range curve" [150, page 5-13]. Diminishing rates of return for expended effort are noted for all but the definite range detection model.

It is recommended in marine SAR [150] to utilize the random search POD curve when the sensor is following a random path, with randomness represented in terms of "the size of the probable error of position relative to the search object" [150, page 5-14], or when the POD curve of the lateral range function is close to that of the random search POD curve [150].

Generally a lateral range function becomes "less peaked" when search conditions deteriorate, decreasing the sweep width. This is true for the inverse cube detection model. Greater research is called for by Frost in determining the POD curves to be utilized under differing searching conditions (rather than the current two choices in marine SAR — the inverse cube or random search models) due to the sensitivity of the level of search effort to changes in that curve [150]. However, Frost states that, due to other uncertain factors affecting search performance, "a case may still be made for the random search POD curve being the most realistic representation of the POD values which should be expected under operational conditions" [150].

5.1.6 Critical Separation Detection Model

We now express the visual detection model presented by Perkins [137] to define the theory of critical separation, in search theory notation. This model has the lateral range curve depicted in Figure 5.5.

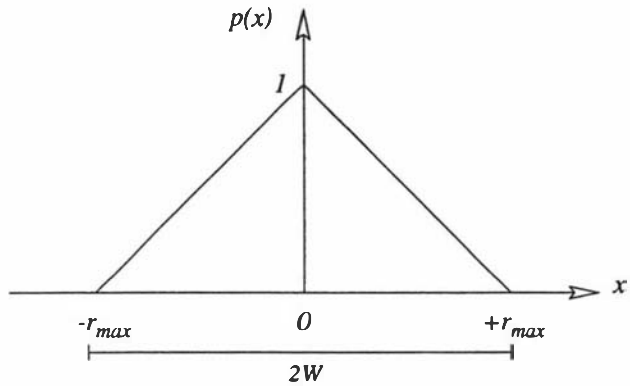

Figure 5.5: Lateral range curve for the critical separation detection model.

Here r_{max} is equal to the visibility limit of the search resource (defined as the "visibility horizon" by Perkins).

The lateral range curve has some similarities to the inverse cube model of detection under ideal conditions, in that it defines a detection probability of 1.0 if the search object is in the path of the sensor, diminishing towards a zero probability of detection as lateral range increases. The curve, in some respects, could be viewed as a 'linearization' of the inverse cube model, being slightly less optimistic of detection over small values of x but more optimistic of detection for larger values of x.

The lateral range function is described as:

$$p(x) = \frac{-|x|}{r_{max}} + 1.0 \quad \text{if } -r_{max} \leq x \leq +r_{max}$$
$$p(x) = 0 \quad \text{otherwise}$$

with the sweep width, $W = r_{max}$.

Perkins notes that the slope of the function is steeper in more difficult terrain and search conditions. This agrees with the decreasing sweep width relationship to deteriorating conditions noted in [150], however, Perkins does not propose that the lateral range function becomes 'less peaked'. He instead assumes that the model will return a POD of 100% at $x = 0$, regardless of terrain or conditions. As Perkins' model is for ground searchers, it is realistic that if the subject lies on the resource's path detection will occur, unlike the inverse cube model which represents detection of a ship from the air. Perkins [137] defines *critical separation*, for search resources conducting a parallel sweep search, to be that spacing which results in an average POD of 50%. Diagrammatically this is represented in Figure 5.6.

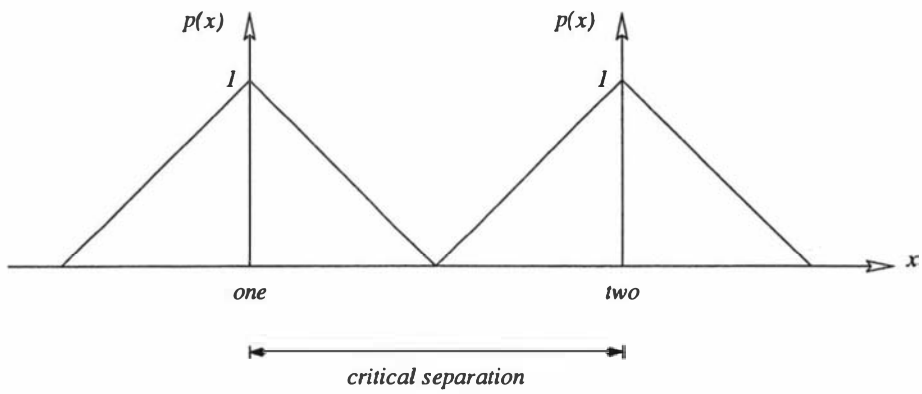

Figure 5.6: Two searchers spaced at critical separation.

Hence, the critical separation distance measure is equivalent to $2r_{max} = 2W$, and the coverage obtained at this spacing is $W/2W = 0.5$.

When the POD obtained by this detection model (for parallel sweep searches) is graphed over differing levels of coverage (and track spacings), the detection model is identical to the definite range model up to a coverage of 0.5, as depicted in Figure 5.7. For greater coverage the detection model lies below that of the definite range model but above the random search model, falling below the inverse cube model for higher valued coverage.

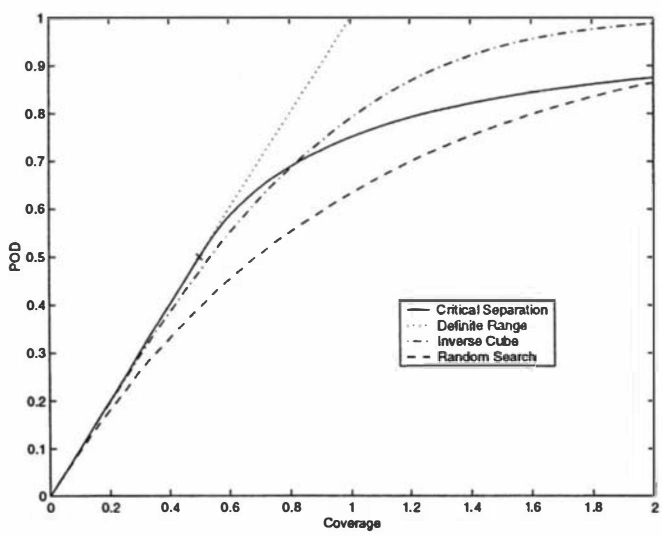

Figure 5.7: POD vs. coverage comparison of the critical separation detection model.

In terms of searcher spacing in parallel sweep searches, the equivalence of Perkins' formula for POD is:

$$POD = 1.0 - 0.25S \quad \text{if } S \leq 2$$
$$POD = \frac{1.0}{S} \quad \text{if } S \geq 2$$

where S is the track spacing in units of W.

Coverage can be expressed as a function of POD as follows:

$$C = POD \quad \text{if } POD \leq 0.50$$
$$C = \frac{1}{4(1-POD)} \quad \text{if } POD \geq 0.50$$

It can be seen from Table 5.1 that doubling the search effort over a region previously covered to a 0.25 coverage returns double the coverage (0.5) and double the POD (0.5). However, to double the search effort again to achieve a coverage of 1.0 does not double the POD. The resulting POD is 75% *i.e.* an increase of a factor of 0.5. Therefore, in terms of the level of POD returned for the amount of effort given, a coverage of 0.5 at a spacing of Critical Separation (CS) is most efficient, confirming the conclusion that Perkins reached via his utilization formula [137].

Table 5.1: Coverage and effort values for different searcher spacings for the critical separation detection model.

Spacing	Coverage	Search Effort	POD	POD Increase
4W(2CS)	0.25	0.25A	0.250	
2W(CS)	0.50	0.5A	0.500	0.250
W(0.5CS)	1.00	A	0.750	0.250
2/3W(1/3CS)	1.50	1.5A	0.833	0.083
W/2(0.25CS)	2.00	2.0A	0.875	0.042

5.1.7 Detection Model for a Land SAR Operation

Frost [61] states that, "I think a key factor in planning and evaluating searches, whether for subjects or clues, is to have an objective handle on their respective detectabilities (sweep widths in the search theory world) so more objective POD estimates can be made from which to compute POS, etc.". Frost [59] questions the validity of generalizing the results of field experiments on detection to real search operations. In particular, Frost considers that results obtained are likely to be "significantly biased toward the high end of the scale." He defends this statement from the position that search resources in field tests have a high expectation of detecting search objects and are often shown the object that they are to detect prior to searching, creating an "intense search image" which acts as a filter to the "visual stimuli" that they encounter as they search [56, 59]. He also criticizes the measurement of POD in field experiments as the proportion of finds rather than the number of detection opportunities compared to those found [56]. Frost states that a similar bias was detected by the USCG when attempting to translate the use of empirical data collected from exercises, to normal operations [56].

Frost [56] calls for more research into defining PODs for land SAR based upon objective, rather than the current subjective measures, observing, in particular, detection over differing conditions and search methods. Tate [56] points out the difficulty of classifying detection probabilities for terrain and vegetation which varies greatly. In particular he states that "we could spend a lifetime just cataloguing different levels of visibility in a year in just this county and our data would probably be useless to anybody else more then 50 miles away." Fowler's response [61] is to propose an experimentation process which first establishes which variables of land searches "have an appreciable effect on sweep width and POD."

In light of these opinions and the fact that there is little research to date into the detection capabilities, and lateral range curves in particular, of ground search resources in New Zealand conditions, and indeed worldwide, we have chosen to utilize the *critical separation* detection model proposed by Perkins [137] for a number of reasons which we will proceed to outline.

To begin with we considered published search detection models. We discounted the possibility of utilizing a definite range or M-Beta detection model, considering these models to be unrepresentative of human searchers. While the inverse cube model of detection has proven sufficient for planning marine searches under optimum conditions, the model itself was developed for search conditions where the search subject was large and to be detected from the air - conditions which are not true for land SAR. However, the idea of the probability of detection monotonically decreasing with increasing $|x|$ is intuitively appealing for detection by the human eye and is also seen in the critical separation detection model of Perkins.

The random search model is suitable when search conditions are unfavourable, and is utilized in this capacity by the USCG. In [150] a case is made for using this detection model to cater for the uncertainties present in search operations. However, being a lower bound on the probability of detection, the random search model is too pessimistic under ideal search conditions utilizing experienced searchers, especially when searching the linear features of the TIN model. As we can control, via simulation, the stochastic factors encountered in a search operation (such as weather changes), and we assume a stationary object (non-random behaviour), identical, experienced searchers (less likely to make navigational errors), and homogeneous terrain over each edge and triangle region in the TIN, the advantages of the random search model are outweighed by its disadvantages. As the critical separation model is easily adjusted for differing vegetation or search conditions by altering the visibility horizon of searchers (and hence the sweep width of their search), stochastic search factors can be easily incorporated into the detection function.

While Perkins and Roberts [137] note additional POD gains through the use of purposeful wandering, we do not directly model this; instead we calculate POD values based on the theoretical critical separation lateral range function. As it is difficult in most real terrains for a team of ground searchers to remain at a constant spacing, we consider that such navigational inaccuracies may off-set these observed gains along with other operational factors which Frost cites in his case for the use of the random search model. The theoretical POD values of the critical separation model are also in line with existing field data from other sources such as Wartes (cited in [137, 159]) and Colwell [21, 25]. In relation to Wartes' data the theoretical values exactly equal the experimental data as is evident from Table 5.2.

Table 5.2: Wartes' field data in critical separations.

Wartes Visual Data		Critical Separation	
Spacing (feet)	POD(%)	Spacing	POD (%)
100	50	1CS	50
60	70	0.6CS	70
20	90	0.2CS	90

Colwell's published experimental POD data for grid searches [21, 25] were obtained by determining the percentage of search objects found by a line of grid searchers sweeping through an area, excluding multiple object finds. A curve-fit algorithm was then applied to the results to reduce the scatter originally present [29]. When comparing the POD values observed by Colwell in experimental searches against the theoretical POD values determined by Perkins' critical separation detection model, a relatively close correlation is observed. In particular, the correlation is closer for visual searches in winter and especially in dense coniferous forest, although a close correlation is also evident for a high visibility sweep in summer.

The comparison data for visual grid searches conducted in dense coniferous forest in winter [21] are displayed in Tables 5.3 through to Table 5.6. We omit the tabulated comparisons for searches conducted in open sub-alpine forest in winter and in dense coniferous forest in summer. The critical separation measure (CS) is set at the spacing of searchers that achieves an experimental field POD of 50%, with all other spacing measurements being referenced from this.

Table 5.3: High visibility sweep data for dense coniferous forest in winter.

Colwell's Observed Data		Critical Separation Theoretical Data	
Spacing(m)	POD(%)	Spacing(CS)	POD(%)
19.1	100	0.16	92
28.9	95	0.23	88
38.2	90	0.31	84
47.5	85	0.39	81
58.6	80	0.46	77
66.3	75	0.54	73
76.1	70	0.62	69
86.3	65	0.70	65
97.3	60	0.79	60
109	55	0.87	56
123	50	1.00	50
139	45	1.13	44
162	40	1.32	38
250	35	2.03	25

Table 5.4: Standard visibility sweep data for dense coniferous forest in winter.

Colwell's Observed Data		Critical Separation Theoretical Data	
Spacing(m)	POD(%)	Spacing(CS)	POD(%)
11.2	100	0.10	95
23.3	95	0.21	89
33.4	90	0.31	85
42.7	85	0.39	80
51.5	80	0.47	76
60.2	75	0.55	72
68.9	70	0.63	68
77.9	65	0.71	64
87.4	60	0.80	60
97.5	55	0.89	55
109	50	1.00	50
122	45	1.12	45
140	40	1.28	39
173	35	1.59	32
240	30	2.20	23
257	25	2.36	21
267	20	2.45	20
275	15	2.52	20
282	10	2.59	19
287	5	2.63	19

Colwell's data tends to be slightly more optimistic than that given by Perkins' model when searchers are spaced closer than 0.5 critical separations (coverage greater than 1.0) in dense coniferous forest in winter. This might indicate that Perkins' model is too pessimistic over small x, however, this pattern is not seen in summer conditions (except in the high visibility sweep), where the opposite tends to be true.

While Colwell's visual data sets for dense coniferous forest in winter correlate well with Perkins' model over all types of grid search, the two search types that result in the highest correlation with Perkins' model for the other two search conditions (dense coniferous forest in summer and open sub-alpine forest in winter) are the high visibility and standard sweeps (results not shown).

The data gathered by Colwell for sound sweep searches over these different search conditions is not as close to Perkins' critical separation detection model results as the visual data but it does, nevertheless, follow it reasonably closely (on average the error is not more than 10%), particularly in winter conditions and for a quiet voice response as illustrated in Table 5.7.

It would appear that Perkins' detection model represents visual detection in the field fairly accurately, particularly in search conditions epitomized by dense coniferous forest in winter. Colwell describes these conditions as Pacific West-Coast Dense Coniferous Forest of elevation 1200m with a "2 metre firm snowpack with no bush showing. There was fresh snow on the trees, it was moderately windy and snowing heavily most of the time" [21]. From these field observations it may be that Perkins' model is better suited to difficult visibility conditions as described above, or to those situations were the subject is conscious and is wearing clothing which is easily visible (standard or high visibility sweeps).

When plotting Colwell's observed POD data against searcher spacing, the curve "consists of a decreasing slope, followed by a 'plateau region' at lower PODs and wider searcher spacings" [29]. However, Colwell later showed that these plateau regions are a product of the way in which the trials were executed and are not accurate in their implication of greater efficiency for wider searcher spacings.

To arrive at this conclusion, Colwell [29] mimicked the search patterns used in these field experiments on Excel spreadsheets and calculated PODs "from the number of searcher spacing 'tracklines' that actually intercepted (i.e., saw) the target objects". The results revealed that plotting POD values against searcher spacing resulted in the same POD values being obtained for discrete ranges of searcher spacings. Colwell notes that fitting a curve through these "POD plateaus" gives a similar curve to those published and considers that this effect was missed initially due the 'noise' present in the experimental field results. Colwell concluded that the cause of these plateaus was geometric in nature and due to the number of searchers which could be placed along a fixed-length baseline, being equal over a range of searcher spacings. He notes that a linear graph is obtained when plotting the number of searchers against the POD values.

Colwell notes that the interpretations which he made in [21] of multiple sweeps at wider search spacings being more efficient were therefore incorrect and that future field experimentations should be conducted by spacing search objects at varying distances from the searchers' tracklines to avoid the results being skewed by the number of searchers searching. Such a method would facilitate the construction of a lateral range curve for a search resource under the given terrain and environmental conditions.

The data originally published is, however, "reliable, reproducible and can be used to estimate expected PODs" [29] by other resources searching an area under the same conditions, and, in particular, using the same fixed baseline lengths. As no field experiments exist to date which sample detection counts with respect to distance from the search resource, or as the percentage of detection opportunities, Colwell's published field tests comprise the most relevant detection information available. As these results compare favourably with the theory of critical separation over most searcher spacings and grid sweep search methods, we consider Perkins' model to be an appropriate one to use to model detection for the purposes of this simulation. In real applications, it would be desirable to conduct experimentations (in line with Colwell's and Frost's recommendations) prior to search operations to determine both approximate lateral range curves for search resources and their traversal speeds over different search methods, terrain and environmental conditions.

Additional advantages of the critical separation detection model are that it is used in land search in the United Kingdom and it is calculated in the field for the conditions at hand, being re-evaluated as those conditions change. Even though it is not a strict representation of reality — Carnes and Cooke [14] state that "in fact it is only when the distance I'm looking gets very close to the visibility distance does the Probability of Detection start to fall off rapidly. Thus the function should not be a linear one" — it bears an intuitive resemblance to human detection of a search object.

The critical separation model is easily adapted to the TIN model as we explicitly model the same type of vegetation in each search region, hence all that is needed to apply the model is a visibility horizon measure to be allocated to each region under differing search conditions. Such visibility measures and correction factors for differing search conditions were outlined in Chapter 4. In particular we define the maximum lateral range distance of a search resource (r_{max}) as its *visibility measure*. When considering sound searches, we view these in the same way as visual searches but we define r_{max} to be the maximum distance at which a subject can be detected by sound alone.

The factors impacting POD which Sweere [155] defines include: the speed of the searchers; their skill; the terrain; the weather; the light; the distance sound travels; the fatigue of the searchers; the detectability of the subject; and the quality of management and communication. Of these factors, the visibility and sound measures address the terrain characteristics, the distance sound travels and the detectability of the subject (which is also impacted by the responsiveness factor); the weather and light conditions prevailing are then addressed by correcting these distance measures by the multiplication of correction factors. We assume that all searchers have identical skill and we do not directly model fatigue, instead limiting continuous search periods to six hours.[1] The speed of the searchers is derived with respect to the vegetation and terrain gradient, and then adjusted for differing weather and light conditions, and whether the resource is searching or accessing a region. We do not directly consider the quality of management and communication within the model, assuming that this is of a high quality.

5.1.8 Detection Along an Edge

Edges of the TIN represent streams, rivers, ridge lines, tracks and vegetation breaks. Hence, the physical features the edges represent vary in width. While a track may often be only 1m wide, a river may be considerably wider. In the computer simulation model, we assume a constant physical width for all edge types which is wide enough for a single searcher to travel over, and a team of searchers to traverse in single file. In the case of an edge representing a river, we assume that a team does not walk through the river itself but along its bank. The bank selected is the one on the same side of the river as any other search regions that the resource has been assigned to search, in order to minimize the number of river crossings.

Visibility in front of the search resource, as he moves along the edge, is given by the visibility measure assigned to the edge, based on its terrain classification. Visibility into the triangular regions adjacent to the edge, as the searcher scans to either side, is given by the visibility measure assigned to that triangular region. As a team of ground searchers searching along an edge moves in single file, the lateral visibility range for each searcher is vm_{tri1} into triangular region $tri1$ and vm_{tri2} into triangular region $tri2$. If the terrain classifications of the two triangular regions are different, the resulting lateral range curve for each resource is non-symmetric as portrayed in Figure 5.8.

While it may be possible to space the searchers in a team across an area centred on the edge, it is not feasible in all instances due to the terrain involved; hence the edge, being the primary search region, is concentrated on by all team members. In addition,

[1] Fatigue could, alternatively, be incorporated by a further correction factor.

Table 5.5: Low visibility sweep data for dense coniferous forest in winter.

Colwell's Observed Data		Critical Separation Theoretical Data	
Spacing(m)	POD(%)	Spacing(CS)	POD(%)
10.1	95	0.10	95
22.3	90	0.22	89
32.4	85	0.31	84
42	80	0.41	80
51.4	75	0.50	75
60.9	70	0.59	70
70.4	65	0.68	66
80.5	60	0.78	61
91.1	55	0.88	56
103	50	1.00	50
116	45	1.13	44
132	40	1.28	39
151	35	1.47	34
212	30	2.06	24
260	25	2.52	20
277	20	2.69	19
288	15	2.80	18

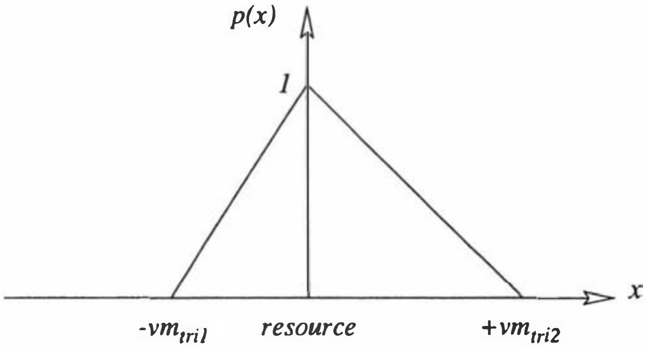

Figure 5.8: Non-symmetric lateral range curve.

Table 5.6: Body sweep data for dense coniferous forest in winter.

Colwell's Observed Data		Critical Separation Theoretical Data	
Spacing(m)	POD(%)	Spacing(CS)	POD(%)
11.8	100	0.16	92
16.3	95	0.23	89
21.1	90	0.29	85
26.1	85	0.36	82
31.3	80	0.43	78
36.9	75	0.51	74
42.9	70	0.60	70
49.3	65	0.68	66
56.1	60	0.78	61
63.7	55	0.88	56
72.1	50	1.00	50
81.6	45	1.13	44
92.7	40	1.29	39
106	35	1.47	34
125	30	1.73	29
157	25	2.18	23
256	20	3.55	14
284	15	3.94	13

Table 5.7: Quiet voice response sound sweep data for dense coniferous forest in winter.

Colwell's Observed Data		Critical Separation Theoretical Data	
Spacing(m)	POD(%)	Spacing(CS)	POD(%)
15.0	100	0.11	95
34.1	95	0.25	88
48.2	90	0.35	83
59.5	85	0.43	79
69.9	80	0.50	75
79.9	75	0.57	71
89.9	70	0.65	68
100.0	65	0.72	64
111.0	60	0.80	60
124.0	55	0.89	55
139.0	50	1	50
160.0	45	1.15	43
235.0	40	1.69	30
263.0	35	1.89	26
275.0	30	1.98	25
284.0	25	2.04	24

the adjacent triangular regions can be visualized to a certain extent as illustrated by the lateral range curve in Figure 5.8. In instances where the bank of a river is being negotiated, the actual distance that a searcher can see into the terrain represented by the triangular region on the other side of the river is reduced by the width of the river *i.e.*, $vm_{tri2} - riverwidth$. While this is what would occur in practice, we do not directly model this as no width is assigned to the river edge in the model. Instead we assume that the visibility horizon to the right of the search resource is equal to the visibility measure assigned to the terrain on the searcher's right.

This is the case when search resources are actually searching the edge. When resources are traversing the edge at a faster speed, merely as a means of access to another location, no sighting of terrain adjacent to the edge is achieved. It is assumed that the resources search only the edge itself without taking the additional time required to search to the left and right of their position. The time to traverse the edge by a team of searchers is assumed to be approximately the same time as required by one searcher. In deteriorating weather or light conditions the time required to traverse an edge is increased further so that the same level of searching can be accomplished.

We assign a POD level of 100% to each traversal of an edge by a search resource, regardless of whether the edge itself is being searched or used only as access, and independent of the terrain and search conditions prevailing. The reasoning for this is that if negligible width is assigned to the edge, then the lateral range $|x|$ approaches 0 where $p(x) = 1.0$. If a subject lies on the path, there is indeed a very high likelihood that at least one searcher in a team will detect them as they move very close to their position, even if they are unconscious or a fatality. Colwell [28] also proposes that a default POD of 100% be allocated to a trail search by a team. He states that "the probability of detection is most likely to be a very high value, approximately 100% or very close to it, along the trail itself. A team searching the trail may have a slightly higher POD than a single searcher so we may therefore assume, for most practical purposes, a default POD value of 100% along the trail."

Assigning a POD level of 100% to edge traversals effectively means that in the case of a stationary subject, once an edge has been traversed no increase in POD_{cum} is possible by further searches. Additionally, it becomes more efficient to allocate search resources to non-traversed edges as means of access to search regions, provided the increase in time over the shortest available path does not outweigh the overall gain in coverage.

When edges are searched, a portion of each triangular region abutting that edge is visualized up to the visibility horizon of the searcher, *i.e.*, from $x = 0$ to $x = vm$. However, in our model we assume that the width of a triangular region is significantly greater than the visibility measure of the region due to the equiangularity of the Delaunay triangulation. Hence, the physical coverage of a portion of a triangular region that results from an edge search is so insignificant as to be of no value. As a result of this assumption, we consider every edge to be traversed at an access only speed which results in a POD of 100% along that edge with no searching of the adjacent triangular regions occurring during this traversal. Hence, as the geographic regions represented by the triangular regions are only covered if the actual region is searched, search objects located within one of these regions can only be detected from searching within the region itself.

5.1.9 Detection Assumptions of the Simulation Environment

Within the simulation environment for the SAR model, we represent a search resource, whether it be a team or an individual, as a single point moving over the TIN. When the team comprises a small number of individuals, we consider that this is a reasonable assumption due to the relatively large geographic region of the TIN. It is an even better approximation if a small spacing between searchers is utilized. We also approximate the actual movement of resources over a triangular search region, as will be detailed in Chapter 8. This is due to the computational complexity involved in directly modelling the resources' trajectory, when in our initial model we only desire to approximate the time spent searching a region. Hence the detection function which we employ is also approximate.

Detection is considered in terms of group detection rather than individual detection. Hence, the distance between the search resource and the search object is calculated as the distance between the object and the point representing the resource. We assume that the point represents the centre searcher in a team of ground searchers. When searching a triangular region, the time to detection is based on the fraction of the region which has been covered. This fraction is equal for all individuals of the team, hence a point or non-point representation of the team would give the same results.

The SAR model assumes detection of the subject (visual or sound detection) occurs if a uniform, randomly generated number between 0.00 and 0.99 falls below the POD_{cum} value resulting from the search. Detection occurs at a point 0.5 times the visibility or sound measure away from the point representing the search resource (or at the time the search task commences if this distance is greater than the distance between the entry vertex and the subject's position). This distance was selected as it is the lateral range at which the probability of detecting the subject under the critical separation detection model is 0.50; it can be regarded as the 'average' detection distance given by the lateral range curve. This 'average' value was chosen because the SAR model developed does not determine the path of the search resource and the subject's position to a detail fine enough to be able to determine the exact distance between the two at the point of detection.

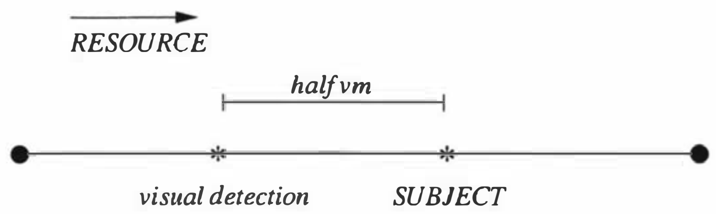

Figure 5.9: Distance representation of the visual detection of a subject.

A distance representation of visual detection is illustrated in Figure 5.9. The time from commencing the search task to visual detection, t_v, is:

$$t_v = max\{0, fraction \times search\_hours - (0.5 \times visibility\_measure/\text{deadhead speed})\}$$

where the variable $fraction$ represents the fraction into the search region at which the subject is located and the deadhead speed is the 'access only' speed specified for the region.

The time to sound detection, t_s, is similarly calculated as:

$$t_s = max\{0, fraction \times search\_hours - (0.5 \times sound\_measure/\text{deadhead speed})\}$$

We assume that the search resource which detects the subject is also the closest in time to the physical location of the subject. It may be possible that, if the subject can be detected from a vertex, a path through another vertex may provide another resource a quicker path to the subject, even if that resource can not yet sight the subject. This occurrence, however, is likely to be rare, and the additional time needed to re-route and communicate such a path to the second resource is likely to outweigh the initial cost savings.

5.1.10 Responsiveness Factor

We model only one type of sound detection — that of detecting a normal voice response. Hence we consider two levels of responsiveness in the subject which will influence the success of sound detection — unresponsiveness and a responsive subject capable of a normal voice response. We assume that the probability of a conscious subject becoming unresponsive is greater for an injured subject or one who has experienced extreme weather conditions, and that this probability increases as the duration of the search lengthens. In the simulation model a Boolean variable *responsive* has its value initially determined by

Algorithm 5.1 function *responsiveness_level*()

```
if number days ≥ 4 or worst weather level = 4 then
    if subject injured then
    prob ← 0.80
    else
    prob ← 0.40
end
else
    if number days ≥ 2 then
        if subject injured then
        prob ← 0.50
        else
        prob ← 0.20
    end
end
rand ← uniform random number downloaded from input file
if (rand < prob) then
    // subject is deemed to be unresponsive
    responsive ← FALSE
end

end
```

the parameter $START\_RESPONSE$, the variable is then arbitrarily set to FALSE under the conditions outlined in Algorithm 5.1. If the subject is unconscious at the beginning of the search operation we assume that they remain unresponsive for the entire operation.

In application, the effect of the responsiveness factor is that when sound sweeps are conducted at a spacing set to detect a normal voice response, sound detection will occur only if the subject is responsive and a uniform, randomly-generated number falls below the POD_{cum} value derived from that search spacing. We assume that sound detection can occur only in triangular regions; we do not consider sound detection along edges of the TIN, or under circumstances of visual searching where search resources are not following sound sweep procedures of producing an audible noise then waiting for a response.

5.2 Modelling the Path of the Subject

We represent the path of the subject in terms of both the regions that they move through, and the entry and exit vertices of each of those regions.

Two different types of subject path generation are considered. The first is a random path generation; the second is a path generation in line with the POA values embedded in the TIN structure, *i.e.*, the POA value assigned to each region of the TIN to model historical POA values. The specific type of path that is generated is determined by the parameter $PATHTYPE$, this being set to *random* or *poa*. At each vertex of the path a decision is made, according to the selected generation heuristic, to determine which of the potential neighbouring regions the subject next moves into. Only vertices of the TIN are considered as decision points and untracked edges are not considered viable regions for allocation. The subject's path is generated from a given vertex on the TIN ($SPATHSTART$) and is a minimum time in length, this being specified by the parameter $SPATHLENGTH$ and given in hours. Regions are added to the path until this minimum time bound is exceeded. The subject is then positioned inside the last region of the path at a specified fraction into the region from the entry vertex, a. This fraction is given by the parameter $LOST\_FRACTION$. The total time that the subject has spent traversing the path is updated to account for this last fraction of movement.

If the last region on a subject's path is a triangular region, we approximate the physical location of the subject by the centre of the triangular region. The x and y coordinates of this centre are given by averaging the x and y coordinates of the three region vertices. The z coordinate of the centre point is interpolated from the plane of the triangular region.

A confusion factor is also incorporated into the path generation of the subject to account for disorientation of the subject or poor navigational skills. This factor is set as the model parameter $CONFUSION$ and is defined as Boolean. If the parameter is set equal to FALSE the subject moves into a new region at each decision point, but if set equal to TRUE the subject can move back into a previous region of the path with a probability set at 0.333.

The time that the subject takes to move through a region i, entering at vertex a and exiting at vertex b, is approximated by:

$$time_i = \frac{distance_{a,b}}{speed_{i,a,b}} \qquad \text{if region } i \text{ is an edge}$$
$$time_i = \frac{urand \times distance_{a,b}}{speed_{i,a,b}} \qquad \text{if region } i \text{ is a triangle}$$

where *urand* is a uniformly, random-generated number over the range [1.2, 1.5], $distance_{a,b}$ is the length of edge (a, b), and $speed_{i,a,b}$ is the traversal speed through region i with respect to the gradient between a and b.

When the subject's path is generated randomly, the next region that the subject moves to is selected randomly, as is the exit vertex of that region.

When the search path is generated in line with historical POA values, the region chosen at each decision point (vertex a) is the neighbouring region with the highest POA value. A neighbouring region is defined as a region which has vertex a as one of its defining vertices. If the region selected is an edge of the TIN, then the exit vertex is simply defined to be the other defining vertex of the edge. If the region is a triangle then the exit vertex, b, is that vertex whose edge region (a, b) has the highest POA value. For example, the triangular region in Figure 5.10 is entered from vertex 2. The two edge regions of the triangle incident to vertex 2 are edge (2,4) and edge (2,6). As the POA value of the edge (2,4) is greater than the POA value of edge (2,6), vertex 4 is selected as the exit vertex.

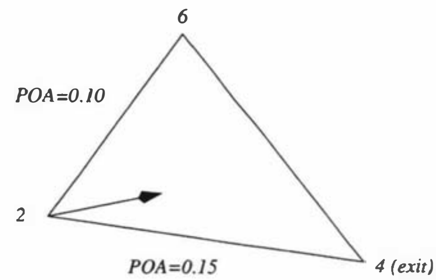

Figure 5.10: Exit vertex determination for a triangular region.

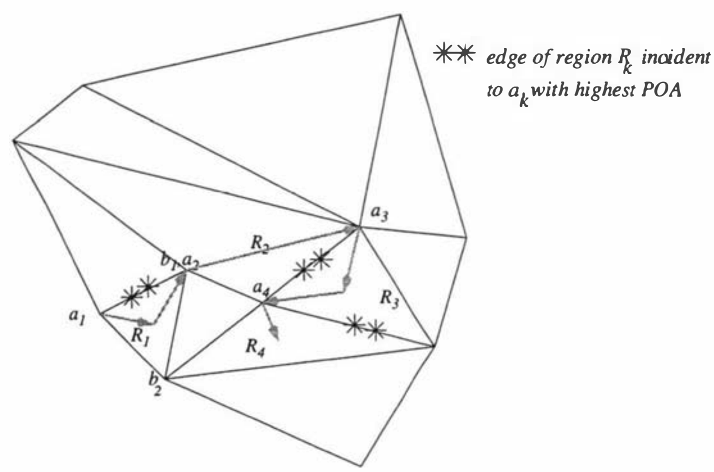

Figure 5.11: Path generated in line with historical POA values.

Figure 5.11 portrays a complete subject path generated via the POA decision criteria. In this example, the first decision point, at vertex a_1, has five neighbouring regions — three edges and two triangles — of which the region with greatest POA is selected, region R_1. Vertex b_1 is then selected as the exit vertex, as edge (a_1, b_1) has a greater POA value than edge (a_1, b_2). The region adjacent to vertex b_1 with the largest POA value is then edge region R_2, if region R_1 cannot be re-selected (R_1 may be re-selected if $CONFUSION$=TRUE). Regions R_3 and R_4 are then similarly selected before the time length of the path exceeds the specified minimum length. The path terminates with the subject being located into region R_4 at the fraction given by the parameter

LOST_FRACTION.

We make the assumption that the stationary subject is not located in a region which may flood. In accordance with this, if the last region selected in the subject's path is at risk of flooding, an alternative region which is not at risk of flooding is selected in its place. This is always possible as we additionally assume that only edge regions may flood, hence, a triangular region adjacent to the subject's current position can be selected as the region in which the path terminates.

5.3 Clue Modelling

5.3.1 Clue Placement

When laying clues along the subject's path, we consider the path as a time path moving through regions of the TIN. This time path is divided into $CLUE\_PARTITION$ equal lengths with a fixed number, $NUMCLUE$, of clues being randomly distributed over each length in accordance with an underlying uniform distribution. Each physical clue is placed in the region of the subject path corresponding to the equivalent portion of the time path. This is illustrated in Figure 5.12 for a subject path consisting of four regions taking a time of 8 hours, with 2 clues being placed in each hour period.

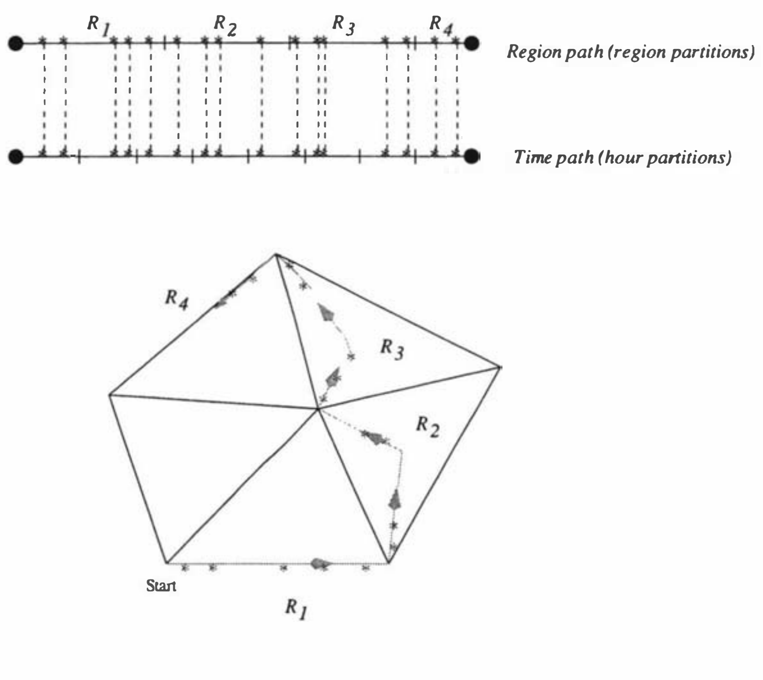

Figure 5.12: Physical clue placement.

The time placement of a clue j in period k (which begins at time t_k), is found by

$$clue\_place_j = t_k + r_k \times time\_length/CLUE\_PARTITION$$

where r_k is a random number in the range [0.00, 0.99] and *time_length* is the total time taken by the subject to traverse the path.

5.3.1.1 Hut Clues

Clues placed at any vertex of the subject's path which has a hut positioned at that vertex, are treated as a separate case. We treat such clues as a physical record of the subject's route intentions as recorded in the hut log. Intentions are assumed to the recorded 58% of the time; this figure is the average percentage of New Zealand search operations where the subject's intentions were known, for the years 1994 to 1999 [122]. The intended vertex and region path of the subject is updated when the hut is searched and the clue discovered. The probability of detecting recorded intentions is equal to 1.00.

5.3.1.2 Clue Attributes

Hill [89] states that, when differing search detectors are utilized in land searches, "the number of potential clues per square mile in which the lost subject has been approaches the seven-figure mark." Hence, instead of modelling each clue left by the subject we define a clue in the model as an amalgamation of spatially correlated clues — a subset of the thousands of clues which would be left by a subject in reality. Effectively this means that an area of trampled bush is recorded in the model as one clue rather than many, as the many small clues comprising this image can be seen instantaneously and give one and the same information.

The following information is recorded for each physical clue:

- clue type,

  1.0 hut clue with an intended route recorded which is the same as what is known,

  1.1 hut clue with an intended route recorded which is different from what is known,

  2 inorganic clue, and

  3 organic clue;

- region of the TIN in which the clue is located;

- the entry and exit vertices the subject utilized when moving through the region within which the clue is located;

- the fraction into the region, from the entry vertex, at which the clue is located; and
- whether the clue has been detected.

Of the physical clues placed, 10% are type 2 clues and 90% are type 3 clues. An example of a non-degrading clue would be an in-organic object belonging to the subject; examples of degrading clues would be footprints, and trampled or bent foliage.

We make the assumption that all clues detected in the simulation of the search operation were laid by the subject. This enables a standard information update for all detected clues without having to allow for a factor of 'noise' created by confounding clues.

5.3.2 Clue Detection Probabilities

Hill [89] addresses the application of POD in land search, particularly relating to clue analysis, from the angle of information theory. In particular he distinguishes between the presence and value of negative information, and the absence of information, stating that: "The fact that they have found no clues is an important message that can have significant informational value. Yet, because it is negative information, its significance is easily overlooked" [89].

Hill illustrates the significance of such information by calculating the number of clues which could be expected to be detected with a clue POD of 25%, using the binomial distribution. From these results he concludes that "the total absence of found clues, even by resources using a relatively low POD, is important information that we shouldn't ignore". In particular Hill proposes that the best search method is "to search each plausible segment once with clue-sensitive resources employing relatively low POD tactics before searching any segment a second time", to reduce the uncertainty of the search in line with information theory.

However, Frost argues this premise over a series of e-mails from 2 June to 10 June 1998, with a further follow-up on 28 September 1998, initiating debate on clue detection probabilities on the SAR Research and Development list. In particular Frost [57] states that "I believe the use of the binomial distribution to plan or evaluate SAR searches is both misguided and dangerous."

Frost [57] criticizes the use by Hill of modelling clue detection probabilities using the binomial distribution as applied to information theory. Frost argues that this approach is invalid as the search problem does not meet the underlying premises of the binomial distribution, namely that of requiring independent, uniformly distributed clues. As clues are laid along the subjects' path of travel, such clues are not independent and are likely to be spatially correlated. Even if clues were left randomly, Frost considers it highly unlikely that a search resource would then follow the subject's path exactly (except perhaps when established trails are followed) to meet the requirements of the distribution. Frost cautions against deciding not to re-search a region which has previously been searched without clue detection or to decide to search elsewhere before re-searching that region as "there are a number of factors affecting where and how effort should be placed to maximize the chances for finding the subject and no one of them, taken alone, is a reliable guide" [57]. He states that it would appear in reality that "clues are missed more often than they are found, even when multiple clues are present to be found" [56]. Frost also notes that:

> "There is what appears to be the very real difference between what empirical PODs say searchers should be finding and what they actually are finding in operational situations. This difference needs to be explained. Either there are typically very few 'detectable' clues present in 'real' searches, or the PODs for those that are there are much less than we are estimating, or some combination of the two" [56].

5.3.3 Modelling Clue Detection

There is very little research on clue detection probabilities and Frost [56] identifies this as a difficult area, stating that the expected number of detectable clues in a region "will probably be a very elusive value for the foreseeable future, if we can ever get any handle on it at all". In light of this we model clue detection by linking it to a search resource's probability of detecting the subject.

The position of a clue on the TIN is not modelled exactly but is instead represented as being located at a given fraction into a region from a specified entry vertex. Similarly the exact path followed by a search resource is also not modelled. Hence, it is not possible from the model to determine how close to a clue a search resource passes and hence the detection probability associated with that lateral range. If we assume that a search resource has the same type of lateral range curve, in shape, for detecting a search object the size of a human or a cigarette box, then the data which changes is the range of x and hence the effective sweep width of the resource. If we depict the lateral range of a resource for detecting a human subject and a small clue in common environmental conditions as in Figure 5.13, then the sweep width for the first curve is $rmax_S$ while, for the second curve, the sweep width is $rmax_c$ (which could also be expressed as $\frac{rmax_c}{rmax_S} \times rmax_S$).

If a search resource has a higher sweep width when searching for a subject under one set of environmental conditions than another, then the sweep width when searching for a clue in those conditions will also be higher. Illustrated another way, if resources are searching at a closer spacing, resulting in a higher subject detection level, they are also

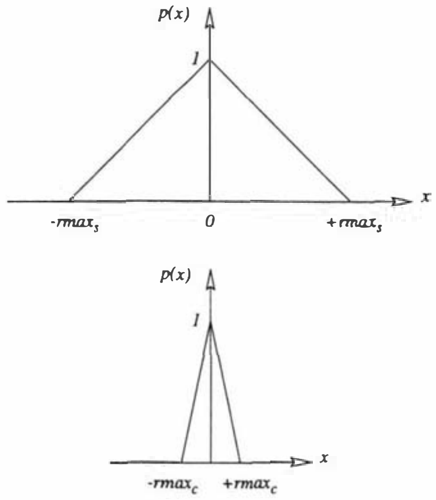

Figure 5.13: Subject and clue lateral range curves.

more likely, on average, to pass closer to a clue.

In particular, we determine the clue POD level, POD_c, of an inorganic clue as

$$POD_C = pod\_fraction \times POD_S$$

where POD_S represents the subject POD level.

Frost [56] states that "since clues other than major articles of clothing are likely to be small and often contrast poorly with the surrounding environment, baseline PODs for such clues are likely to be quite small in comparison to the PODs for the subject for the same amount of effort spread over the same amount of area under the same conditions."

While the variable *pod_fraction* could be parameterized, we arbitrarily set it equal to 0.2. Hence if $POD_S = 60\%$, $POD_c = 12\%$. This level of clue detectability is probably higher than would be encountered for most clues in reality, but it is a reflection of the modelling decision to define a clue as an amalgamation of a subset of the clues left by the subject. We also assume that search resources are trained in clue awareness.

By linking clue detectability to subject detectability the current terrain, light, and weather conditions are automatically factored in. However, for organic clues which can degrade over time detectability is adjusted to account for the worst weather level experienced, and the number of days which have passed, since the clue was formed. The clue POD level of an organic clue is then calculated as

$$POD_C = pod\_fraction \times (1 - env\_factor) \times POD_S$$

The environmental factor, *env_factor*, is arbitrarily defined as given in Table 5.8.

Table 5.8: Environmental degradation factors of clue detectability.

Condition	env_factor
Worst weather level = 4 or number days $\geq$ 6	0.6
Worst weather level = 3 or number days $\geq$ 4	0.4
Number days $\geq$ 2	0.2

For example, if $env\_factor = 0.4$ and the subject POD level equals 60% then the clue POD level becomes 7.2%.

If a sound sweep search is being conducted of a region the predicted POD_S is equal to that of a visual sweep search at one third the spacing. Hence, as we assume that there exist no clue types which can be detected by sound, leaving only those which can be detected by sight, we reduce the POD_C by a factor of one third in sound sweeping. This effectively calculates the POD_C under these conditions as a fraction of the visibility measure of the resource for that region.

5.3.4 Physical Clue Detection

When a search resource is about to enter a region, either to search it or to use it for access to another region, a procedure is called to determine if there are clues within that region and whether or not they will be detected by the resource. If a clue lies in the region, it is detected if a uniform randomly generated number between 0.00 and 0.99 is less than the POD value attached to that clue. Unlike detection of the subject, we assume that visual detection of a clue occurs at its physical location. This assumption is justified by the usually small nature of clues and their low contrast with the environment, making visual detection at any significant distance from their physical location unlikely. This is also reflected in the relatively low POD values assigned to the clues. When a clue is detected, its discovery is communicated to the management team at the base at the time of discovery, the information pertaining to the clue is analysed and the subject's location probability distribution is updated in light of this information.

All detectable clues are ordered chronologically from the entry of the search resource into the search region. As the clue placement is recorded in orientation to the vertices that the subject entered (vertex a) and exited (vertex b) the region, some calculation is required to determine the time of detection if the search resource is entering or exiting the region by a different pair of vertices. In the instance where the region is an edge of the TIN this is straightforward and the time to discovery, from entering the region, is calculated as:

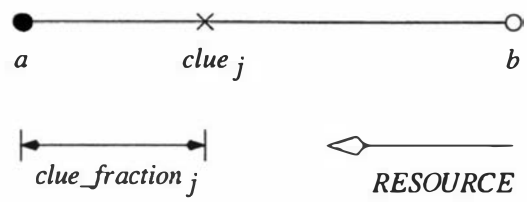

Figure 5.14: Clue position for a resource entering an edge region at vertex b.

$$\begin{array}{ll} region\_time_i \times clue\_fraction_j & \text{if the resource enters region } i \text{ at vertex } a \\ region\_time_i \times (1 - clue\_fraction_j) & \text{if the resource enters region } i \text{ at vertex } b, \end{array}$$

where $region\_time_i$ is the time required to search region i and $clue\_fraction_j$ is the fraction into region i from vertex a which clue j is positioned at. This is illustrated in Figure 5.14.

If the clue is located in a triangular search region, i, then we approximate the time to clue discovery as:

$$\begin{array}{ll} region\_time_i \times clue\_fraction_j & \text{if the resource enters } i \text{ at vertex } a \text{ or exits } i \text{ via vertex } b \\ & \text{and by} \\ region\_time_i \times (1 - clue\_fraction_j) & \text{if the resource enters } i \text{ at vertex } b \text{ or exits } i \text{ via} \\ & \text{vertex } a \end{array}$$

The detection time of the clue is approximated from the direction the subject moved through the region in comparison to the direction that the search resource is to move through the region, *i.e.*, whether the search resource follows a similar direction to the subject or not. In particular we consider whether the resource enters or leaves the region by the same vertex as the subject, or if the resource enters the region by the vertex the subject exited by, or leaves the region by the vertex that the subject entered by.

This approximation is based upon the assumption of a sweep search by the search resource which covers the terrain represented by the triangular region from entry to exit vertex, to the coverage level indicated by the POD value of the specific search method utilized. As the actual path of the subject in that region is unknown, and in particular it is unknown how the search resource's path correlates to the subject's path, it is not possible to more accurately define the time of detection (*i.e.*, the clue is defined as being located at a fraction "into the region", not at a specific set of coordinates). A visual representation of a search through a region that contains a clue is shown in Figure 5.15.

Once a clue has been detected and this fact communicated to search management, a procedure determines the time to detection of the next clue lying beyond the search

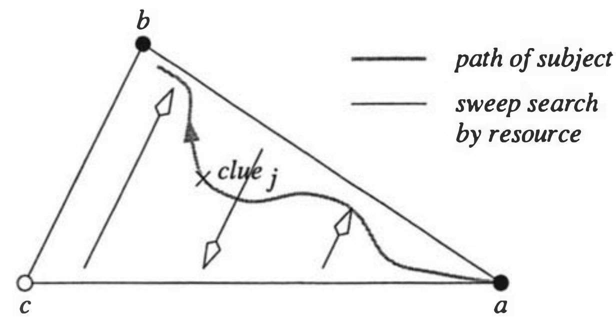

Figure 5.15: Clue position for a resource entering a triangular region at vertex c and exiting at vertex a.

resource's current position.

We make the assumptions that a clue is detected only once over the entire search operation and that any resources assigned to re-search the region where the clue was detected have already been briefed as to its location. The assumption of a once-only detection avoids the rediscovery of the same clue artificially boosting the clue count and hence the POA value of any given search region. Additionally, we assume that no clues are discovered after the detection of the subject, as resources are no longer traversing the region in a 'searching mode' but are traversing at a deadhead speed. Any clues found at this time are redundant in any case.

Whenever there is an interruption to searching, such as a rest or communication break, or the environmental factors of the search alter, the times until detection of any clues are updated to account for this.

5.3.5 Information Update

When a clue is detected, the known information relating to the subject's movements is either confirmed or contradicted. In particular, the POA values representing the subject's location probability distribution are adjusted to reflect an increase in the probability that the subject may still be located in the search region in which the clue was located, or in a region adjacent to this. When these POA values are then normalized so as to sum to one, the POA values of search regions falling outside of this region set decrease to reflect the decreased likelihood that the subject is located in these regions.

The POA value of a region in which a clue is discovered is adjusted by a linear adjustment formula in [77]. The level of adjustment of the POA values is defined by the variable ROC, standing for Relevance of Clue and is given as

$$POA^* = ROC \times (1 - POA) + POA$$

where ROC is subjectively allocated a value in the range [0.00, 1.00] to reflect the relevance of the clue to the operation.

Carnes and Cooke (in their PODSheet computer programme) [14], update the POA value of a search region by the following formula.

$$POA_i^* = POA_i \times IOC$$

where IOC represents the Influence of the Clue and is calculated as

$$IOC = e^{(potential\_clue\_influence \times clue\_probability \times -0.0866)}$$

The $potential\_clue\_influence$ is a value in the range [0,8], and represents the information given by the clue as to the likelihood that the subject is in region i. If the search manager is "virtually 100% certain" that the clue indicates that the subject is in region i a value of 0 is assigned, while a value of 8 is assigned if the manager is "virtually 100% certain" that the clue indicates that the subject is not in region i.

The $clue\_probability$ attached to the clue is a value in the range [0,4] representing the relevance of the clue. A value of 4 indicates that the clue is very likely to be good while a value of 0 indicates that it is very likely to not be a good clue.

The POA^* values are then normalized to sum to one.

In order to simulate a search operation in a way which does not rely on expert determination of clue influence or relevance at the time of clue detection, we construct rules to automatically update the probability distribution of the subject's location. These rules are based upon the number of clues detected in a search region and the direction of subject movement that is indicated by these clues. As the initial clues found within a search region have the potential to provide the greatest increase in information known on the subject's movements, the detection of further clues in the same search region is less likely to provide new information and more likely to build upon the information gained from previously detected clues. With this in mind we have developed a measure (which we term $ROC_{cum,i}$) to measure the total relevance to the search operation of all clues detected within search region i. We view this cumulative clue relevancy as increasing with each clue discovered, but increasing only to a limit to reflect the actual increase in knowledge gained with each successive clue discovery. To this end we have selected an exponential formula to update $ROC_{cum,i}$ from the previous $ROC_{cum,i}$ value.

$$ROC^*_{cum,i} = 1 - e^{-ROC_{cum,i}}$$

The initial $ROC_{cum,i}$ value is set by a parameter of the simulation, ROC_0, to enable the significance of clues discovered to vary and to monitor the impact that this has upon planning and allocation decisions, viz:

$$ROC_{cum,i} = 1 - e^{-ROC_0}, \forall i$$

When a clue is discovered in region i we update the POA value of that region to:

$$POA_i^* = ROC_{cum,i} \times (1 - POA_i) + POA_i$$

If different scenarios of the subject's likely movements have been developed then the detection of a clue may indicate the dominance of one scenario over another and the POA values of all regions can be recalculated based upon a shift in the scenario weight probabilities. In the simulation environment of the SAR operation we do not consider the presence of multiple scenarios, instead we plan on either the intended route of the subject (if known), or on the historical POA values embedded in the TIN. When a clue is detected we increase the POA value of the region in which it was detected by the formula above. If the POA of the region was initially high relative to other regions, this strengthens the initial planning assumptions; otherwise, it shifts the POA values to weight this region more highly, mimicking a scenario change.

In conjunction with increasing the POA value of the region in which the clue was discovered we also increase the POA value of adjacent regions in the direction of subject movement indicated by the clue. As clues are modelled to impart information on the subject's entry and exit vertex when moving through a region, this applies to all search regions (and as such implicitly does not include untracked edge regions) which have the subject's exit vertex in common. The regions whose POA values are increased are illustrated in the Figure 5.16 and identified with a '+'.

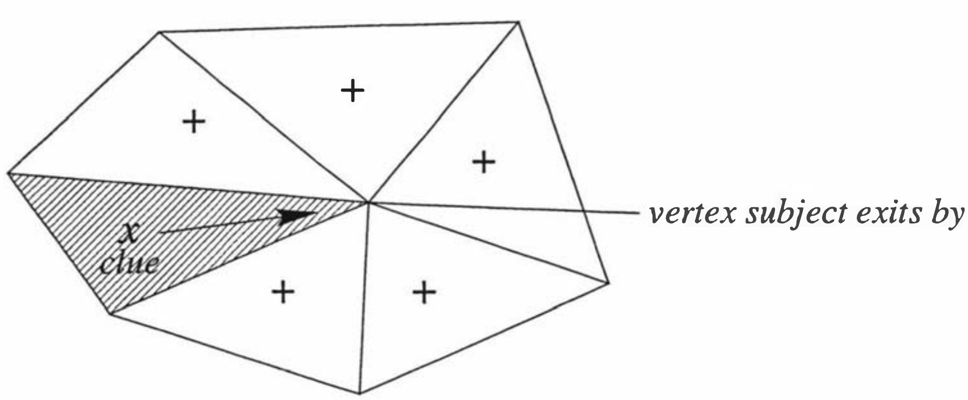

Figure 5.16: POA update upon clue detection.

We adjust the POA values of these regions to a lesser extent than the region in which the clue was located. The sum of the POA values of those regions falling in this set is increased to

$$POA\_sum^* = (0.5 \times ROC_{cum,i}) \times (1 - POA\_sum) + POA\_sum$$

The individual regions in the set then have their POA value updated, in proportion, to

$$POA_j^* = \frac{POA_j}{POA\_sum} \times POA\_sum^*$$

If the case arises where $POA\_sum = 0.0$ we consider POA_sum to equal 0.01 in the above equations, in order to increase the POA values of the regions in the adjacent set.

Additionally, in all instances where $POA_j = 0.0$ for some j, then this region j has its POA value increased to

$$POA_j^* = \frac{0.01}{POA\_sum} \times POA\_sum^*$$

When a clue is discovered in the form of a hut log, the POA values of all regions with a vertex in common with the vertex that the hut is located at are increased. In addition, if the intended route outlined in the hut log differs from the intended route known, then the POA values of the regions along, and adjacent to, the altered portion of this route are increased, while those regions comprising, and adjacent to, the previous route are decreased. Regions which were adjacent to the subject's initial intended route and which are also adjacent to the new intended route, do not have their POA values altered as the POA values of these regions were initially set to be higher at the outset of the search operation due to their close proximity to the intended route. (This process is conducted at the commencement of the search operation when the subject's intended route is known and adjusts POA values in the same manner as now described). Pictorially this may be illustrated as in Figure 5.17, with decreasing and increasing location probabilities indicated by '−' and '+' respectively. Regions whose POA values do not alter are represented by 'o'.

Search regions which comprise the altered portion of the subject's intended route have their POA updated to

$$POA_i^* = ROC_{cum,i} \times (1 - POA_i) + POA_i$$

while adjacent regions are updated, in proportion, to sum to

$$POA\_sum^* = (0.5 \times ROC_o) \times (1 - POA\_sum) + POA\_sum$$

and regions adjacent to the previous intended route and not adjacent to the altered intended route are updated, in proportion, to sum to

$$POA\_sum^* = POA\_sum - (0.5 \times ROC_o) \times POA\_sum$$

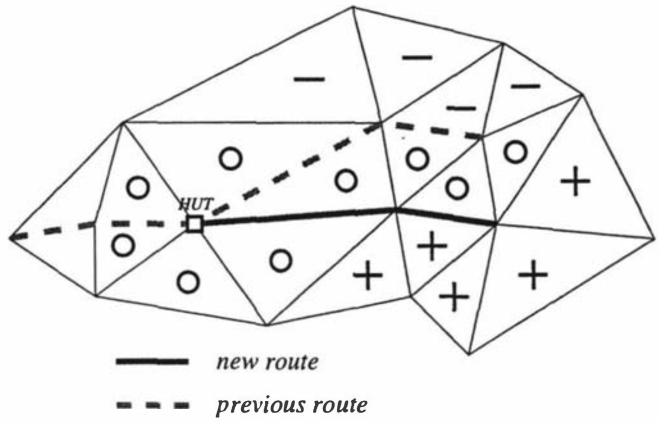

Figure 5.17: Updating location probabilities for an altered subject route.

After the POA values of regions falling in the update set are adjusted the POA values of all regions in the TIN are normalized to sum to one. However, the POA values of untracked edge regions remain at zero in accordance with the assumption that such edges do not contain the subject, being solely representative of boundaries between search regions rather than actual linear features.

A physical search model has now been developed which incorporates: the search terrain; visibility restrictions; traversal of search resources; subject movement and responsiveness; clue formation and deterioration; and a detection model for both a human subject and clues formed by him, based on Perkins' critical separation theory. Having defined a physical model over which to conduct a search we now consider the problem of resource allocation, and in particular, the specific paths followed by the resources, in the context of the current literature. We develop alternative problem formulations for the purposes of exposition, and identify the similarities of the SAR problem to existing problems as well as its unique aspects.

CHAPTER 6

Placing The SAR Problem In Context

Following current international SAR procedures we consider three phases to the SAR operation — hasty, efficient and thorough. We define the three phases over the search region graph as incorporating:

- Phase One: edge searching of required edges, also incorporating required vertices.
- Phase Two: edge and triangular region searching of selected regions of the search region graph utilizing efficient search patterns.
- Phase Three: edge and triangular region searching of selected regions of the graph utilizing thorough search patterns.

The main differences between search Phases Two and Three are the search patterns utilized. In the third phase these patterns are more thorough and result in higher POD levels. Generally this increase in thoroughness is offset by increases in manpower requirements, as searchers are spaced at closer intervals. These patterns have the lowest net benefit return per searcher hour of effort.

Within this chapter the fundamental and unique aspects of the SAR problem are explored in relation to existing problems in the literature.

6.1 Dynamism

The SAR problem is dynamic in that it evolves in real-time with the search revealing new clues, or lack of clues, to the subject's whereabouts, as terrain is searched. In addition, the problem is stochastic in that it is affected by uncontrollable elements, such as weather, and changing factors such as the number of available searchers. In a dynamic problem the time dimension is essential to the crux of the problem and information regarding future events is unknown or tentative. The problem itself may be open-ended (having an indefinite time period) and the model of a dynamic routing problem may be categorized by paths, in preference to tours, due to this. Generally events happening in the 'near-term' are considered to be of greater importance. Psaraftis [141, page 226] considers it "unwise" to commit resources too far into the future in a dynamic scenario, as "other intermediate events may make such decisions suboptimal, and because such future information may change anyway."

The objective function and time constraints of a dynamic problem may also be of a different structure to that of static problems, and are defined for subproblems (decompositions in time or space) of the overall problem. Dror and Powell [44, page 12] note that dynamic and static problems do not possess a standard formulation and are "often characterized by the lack of a well-defined objective function." Often surrogate objectives will be utilized for tractability and these may be more representative of objective functions associated with static problems. Additionally, Psaraftis [141] considers that objective functions should meaningfully incorporate known future information to some extent. Often time constraints will be defined as 'soft' constraints.[1] The author considers that meaningful performance measures for dynamic problems are throughput or productivity related.

As there are no firm guidelines on how long a SAR operation will be conducted before suspension occurs and there is no means of establishing how long it will take to detect the subject, the SAR problem is indefinite in length. The problem is time critical and future information inputs are unknown. Re-scheduling and re-assigning search tasks to resources with respect to priorities derived from new information must be incorporated within the solution structure. Due to constraints of urgency all effort must be put into the immediate time frame; hence the problem displays all of the essential elements of a dynamic problem. The SAR problem is not only a dynamic problem with stochastic elements but, due to the urgency of response required, it also displays all the elements of a real-time decision problem.

6.2 Coverage

The classical definition of coverage from search theory is the ratio of search effort to the size of the search area and is defined in Section 2.7. We introduce a new definition of

[1] Psaraftis [141] defines a 'soft' constraint as one which can be violated at a cost, and one which is subject to update and revision.

coverage, a *visibility cover*, which is analogous to the traditional definition of coverage within the Operations Research (OR) literature. This traditional definition applies to problems where locations are to be found for facilities which service portions of a network. If a portion of the network is within a certain distance of the facility it is deemed to be "covered" by the facility.

We define terrain falling within a search resource's visibility measure (vm), or alternatively within their sound measure (sm), as falling within the resource's *visibility cover* (the term 'visibility' is used interchangeably for both sound and visual searches). Hence, a search object positioned at a point on the TIN located at a distance d from the resource is *visibly covered* by that resource if $d < vm$ for a visual search and if $d < sm$ for a sound search.

As the resource passing by such a search object has a positive, non-zero probability of detecting the object, the object can be viewed as being 'covered' by that resource. An object may however need to be 'covered' in such a way on more than one occasion for detection to actually result. In this sense the visibility cover definition differs from the OR coverage definition and, unlike most traditional measures, visibility coverage also differs in its dynamism. As a search resource's visibility measure alters with weather and light conditions so does the resource's visibility cover.

In light of this definition of coverage it is possible that some edges within the search area need not be explicitly traversed, *i.e.*, visiting either or both of its end vertices may result in all terrain along an edge being visibly covered by a resource. Thus, in order for such an edge to be visibly covered it is not required that this edge should feature in one of the final search paths, only that one, or both, of its end vertices must. Accordingly, edges are partitioned into three disjoint classifications of visibility measure. In increasing level of restriction, the terrain represented by an edge falls within a resource's visibility cover if:

1. only one of the end vertices of the edge is visited;

2. both the end vertices of the edge are visited;

3. the edge is physically traversed completely.

Henceforth we refer to an edge as falling within Class 1, Class 2 or Class 3 as defined. With changing visibility conditions over the search operation it is possible that an edge may fall within a different edge class at different time points.

The definition of *explicit and implicit searching* is also utilized. An edge which is physically traversed as a component of a resource's search path is described as being *explicitly searched*, while an edge in Class 1 or Class 2 is *implicitly searched* if its required vertices are a component of a resource's search path and it is not explicitly searched.

Coverage-routing problems in the literature are now examined with respect to their applicability to the SAR problem.

6.2.1 Coverage — Routing Problems in the Literature

Initially those problems which seek a single tour or path under some measure of coverage are considered.

6.2.2 Vertex Coverage

6.2.2.1 Tour Routes

The ***Covering Salesman Problem (CSP)***, first introduced by Current in 1981 [36, 68], has the objective of finding the tour of minimal length over a subset of given vertices, such that every vertex which is not on the tour is within a predetermined covering distance of a vertex on the tour. The CSP is applied particularly to bilevel transportation networks where a service is provided at each vertex on the tour and those 'customers' off the tour must be close enough to access this service at at least one vertex on the tour. Particular applications include rural health delivery and postal services. Current and Schilling [36] also develop a bi-criterion formulation for the CSP that additionally considers the objective of minimizing the cost of stopping at each vertex on the tour.

Gendreau *et.al.* [69] further refine the CSP to define the ***Covering Tour Problem (CTP)***. Unlike the CSP, the CTP is defined over a graph consisting of a set of vertices which can be visited by the final tour — a subset of which must be visited — and an additional set of vertices which must be covered. Vertices lying off the final tour are covered if they lie within a specified distance from a vertex on the tour and the minimal length tour is sought.

Gendreau *et.al.* [68], and Current and Schilling [36], show that the CTP and CSP, respectively, reduce to the TSP when the vertices which must be covered by the tour are required to be within a distance of zero of the tour, *i.e.* the vertices must lie on the tour itself. Hence the problems are of NP-hard complexity.

The ***Geometric Covering Salesman Problem***[2] is defined by Arkin and Hassin [5] as the problem where a salesman needs to determine a minimal length tour which intersects geometrically defined neighbourhoods of a set of buyers *i.e.*, a specified set of regions of a plane. The neighbourhoods are defined by each buyer as the distance that

[2]The Geometric Covering Salesman Problem is also referenced as the Travelling Salesman with Neighbourhoods problem [5].

they are prepared to travel from their location to meet with the salesman. The authors note that the problem generalizes the Euclidean TSP where buyers are represented by single vertices; the GCSP is also NP-hard. The Geometric Covering Salesman Problem is described by Gendreau *et.al.* [68] as a continuous version of the CTP.

Two problems which are related to the CSP/CTP are the ***Median Tour Problem (MTP)*** and the ***Maximal Covering Tour Problem (MCTP)***, introduced by Current and Schilling [37]. Both problems are also shown as NP-hard extensions of the TSP which visit only a subset, p, of the vertices in the graph. (This number could be allowed to vary as an additional criterion.) Both the MTP and the MCTP are formulated as bi-criterion problems where the first criterion is to minimize the total length of the tour. In the MTP each vertex has an associated demand and the second criterion is then to minimize the total demand weighted distance that must be travelled from vertices not on the tour, to the vertex in the tour closest to each. The second criterion of the MCTP is to minimize the demand at the vertices not covered (by some prespecified maximal travelling distance) by a vertex on the tour. The MCTP can be expressed as a special instance of the MTP by editing the distance matrix.

Essentially such problems consider two decisions; namely — which vertices to include on the tour and how best to route the selected vertices. Generally a conflict exists between the length of the tour and the accessibility objectives.

The CTP is applicable to a reconnaissance search phase where searching is restricted to specified vertices, such as huts, and the implicit searching of Class 1 and Class 2 edges. In this instance the set of vertices which must be visited are the identified huts and the end vertices of the edges required for each edge to be visibly covered. The vertex of a Class 1 edge which is not included in the final tour is covered by the tour in the manner described by the CTP, as it lies within a prespecified distance from a vertex in the tour. This distance is the visibility measure of that edge. While the predetermined covering distance is one fixed measure for all vertices in the problems described in the literature, this is not translated to the SAR model where visibility measures will differ depending on the terrain, and will also alter temporally when affected by changing weather and light conditions. The CTP restricts resources to beginning and terminating their search at the same vertex, *e.g.* the search base.

The objective of minimizing the length of the tour applies well to the reconnaissance phase where speed is considered an essential factor. When the coverage of a Class 1 or Class 2 edge is further modelled by delaying the search resource at the end vertex lying on the tour, this can be viewed as the cost of stopping at that vertex. In this case the bi-criterion formulation of Current and Shilling is applicable. If an objective of maximizing POS were instead sought, then the MCTP of Current and Shilling could be adapted such that the demand attached to a vertex not covered (and hence a Class 1 or Class 2 edge not covered) was equal to the predicted POS value of the uncovered edge incident to it.

As the problems are formulated for single tours, the final tour would need to be split for multiple resource deployment or the problems would need to be adapted to produce multiple tours. The tours also route only vertices, hence they are unsuitable for the inclusion of Class 3 edges in the reconnaissance phase.

6.2.2.2 Path Routes

The problem addressed by the ***Shortest Covering Path Problem(SCPP)*** is to find the path of least cost between two specified vertices such that all vertices in the network are covered. Each vertex is covered if it is within a prespecified distance of a vertex on the path, where this distance does not have to be identical for every vertex [33]. The SCPP was first formulated by Current *et.al.* [34]. The shortest path problem and the hamiltonian path problem are both special cases of the SCPP. The shortest path problem occurs when the covering distance is significantly large and the hamiltonian path problem occurs when the covering distance is significantly small. Current *et. al.* [33] prove that the SCPP is NP-complete via a transformation to the hamiltonian path problem, indicating that the complexity of the problem can be reduced by utilizing a limited network where some arcs already exist.

The SCPP combines the theory of covering, in relation to facility location, with the shortest path problem to jointly address both network design and routing. As for the covering tour problems, the SCPP is especially applicable to bilevel routing.

Current *et.al.* [34] extend the SCPP to consider multiobjectives such as the trade-off between minimizing the path length and minimizing the maximum covering distance — effectively placing a budget constraint into the objective function. A trade-off curve can be generated for these objectives considering differing values for the covering distance.

Current *et.al.* [33] perform computational results on randomly generated symmetric networks ranging from 10 to 90 vertices, of differing densities, and conclude that the problems with an intermediate covering distance are the most difficult to solve.

Current *et.al.* [35] further extend the SCPP to the ***Maximum Covering Shortest Path Problem (MCSPP)***. The objective of this problem is to find the minimum cost path between a predetermined starting vertex and a finishing vertex, such that the selected vertices on the path maximize the total demand covered by the path. In the MCSPP a demand exists at each vertex in the graph and this demand is met if the vertex is located on the path or if the vertex lies within a specified distance of a vertex on the path. The MCSPP is hence a multiobjective problem whose objectives conflict.

The ***Maximum Population Shortest Path Problem (MPSPP)*** is defined by

Current *et.al.* [35] when the covering distance for all vertices in the graph is equal to zero, *i.e.,* the vertices must be visited explicitly by the path for their demand to be met. This is a special case of the MCSPP and has the same objectives.

Both the MCSPP and MPSPP are defined over a graph consisting of non-directed arcs. Current *et.al.* [35] emphasize the flexibility of the problem formulation which can also be extended to cater for: instances where specific arcs or intermediate vertices are required to be on the path; cases where multiple coverage of vertices is desirable; the inclusion of mandatory closeness constraints[3] which could be varied for differing vertices; and situations where varying service demands may occur at vertices — each service being represented by an additional covering objective. The authors state that minimizing coverage distance can be expressed as a third objective in the MCSPP if it is not a well defined value.

The SCPP is applicable to planning search resource paths, as paths, not tours, are specifically detailed, although the problem addressed is once again a single, not multiple, path problem. Again the notion of coverage is identical to that of the various covering tour problems and fits well with the visibility cover in the search, as defined by searcher visibility. In the formulation of the SCPP, coverage distance is also able to be varied for differing vertices. However, once again, vertices are explicitly included in the path, not a specified set of edges. Additionally, symmetric distances are properties of the underlying network for the covering path problems which are not features of the search region graph.

The MPSPP by regarding vertices as covered, only if they are explicitly present on the path, could be seen as a special case of a reconnaissance search phase where searchers concentrate their searching on those edges and vertices physically traversed, and do not slow to investigate other areas in their proximity. If conditions were so severe as to restrict visibility to the terrain directly before, and underfoot, of the searcher, this constraint would be a true representation of the problem at hand.

6.2.2.3 Tree Networks

The ***Minimum Cost Covering Subtree Problem (MCCSP)*** is introduced by Aaronson-Hutson and ReVelle [1]. The problem is that of finding the minimum cost group of arcs which form a subtree, where all vertices are within a prespecified distance from some vertex in the subtree. Hence the problem is effectively a SCPP where the path must form a subtree and the underlying network forms a tree. Aaronson-Hutson and ReVelle define direct coverage as the coverage pertaining to vertices on the subtree and indirect coverage pertaining to the cover of vertices not on the subtree. They further define two

[3] The maximum distance that a vertex can be from the path.

classifications of indirect coverage. The first classification applies to vertices which are within a specified distance of a vertex on the subtree. In the second classification, vertices are covered if they are within a specified distance to an arc of the subtree.

Aaronson-Hutson and ReVelle [1] introduce the ***Maximal Indirect Covering Subtree Problem (MICSP)*** utilizing the first classification of indirect coverage. The solution to the MICSP is a subtree which maximizes the demand at those vertices covered by the subtree (indirectly and directly) and minimizes the total cost of the subtree. Hence, a weighted trade-off occurs between total coverage and total cost.

The Minimal Length Covering Subtree proposed by Kim *et.al.* (cited in [1]) is related to these problems. The problem has the objective of minimizing the total length of the subtree so that demand vertices are covered, and the solution can include whole or partial arcs. The authors define loss as "proportional to the distance to a terminal vertex of the subtree", where this distance must be within a specified covering value for each demand vertex in the network.

Aaronson-Hutson and ReVelle discuss service applications where demand declines with distance from a subtree facility due to access inconvenience. Such declining demand can be modelled by a discounting factor. As distances from indirectly covered vertices to the closest vertex in the subtree are not directly modelled, a mandatory closeness standard is used as an upper bound on the distance of uncovered vertices from the subtree. Distance and time standards are used interchangeably. The mandatory closeness measure "serves as a proxy for an objective of minimizing maximum distances" and is introduced as a constraint on the problem.

Both the MCCSP and MICSP are solved by Aaronson-Hutson and ReVelle [1] for a special case where the underlying network is a spanning tree, thus reducing the complexity of the problem by guaranteeing unique paths between all arc pairs.

As these problems are defined over tree networks they would only apply to very special instances of the search region graph. The extended classifications of indirect coverage are useful in that a visibility measure is geometrically defined 'as the bird flies' and does not need to be defined upon an existing network *i.e.,* as the edge distance between an uncovered vertex and a covered vertex. In the SAR problem, a vertex not traversed in a search path will be visibly covered if it lies within a specified distance of a traversed edge, as long as no visibility restrictions exist. A mandatory closeness criterion could also be applied to uncovered vertices (a vertex not in the visibility cover) to ensure that no vertex was significantly far from a searched area.

6.2.3 Geometric Coverage

The ***Watchman Route Problem*** is defined by Chin and Ntafos [17] as finding the minimum cost route within a polygon such that every point in the polygon is visible from at least one point on the route. The problem is originally derived from the application of stationing watchmen in an art gallery and includes such applications as security surveillance. O'Rourke [127] considers the problem and extensions in details.

Chin and Naftos [17] show that the Watchman Route Problem over convex polygons which may contain convex, interior polygonal holes,[4] is an NP-hard problem. The authors extend this result to show that the Watchman Route Problem over simple polyhedra and over rectilinear polygons with rectilinear holes, is also NP-hard.

The concept of visibility utilized in the watchman route problem is effectively unlimited except when prevented by non-opaque barriers. In this respect it is dissimilar to the limited visibility encountered in a bush-covered terrain. In effect the problem is defined similarly, but the geometric environment that it is executed upon is significantly different in the extension of dimension resulting from the inclusion of an elevation factor. A similar objective of minimizing the route of the searcher such that all points within a subset of the TIN are visible from some point on that route can be utilized, with the additional incorporation of other criteria such as that of regions of higher priority requiring earlier coverage.

6.2.4 Edge Coverage

The ***Weighted Vertex Cover Problem*** is defined as the problem of determining a minimum weight set of vertices which cover all the edges in a given graph [4]. The vertices each have an associated weight and an edge is considered to be covered if at least one of the end vertices of the edge is in the vertex set. This problem is also NP-hard.

As the problem considers weights on the vertices rather than the edges, it is not directly related to the SAR problem. However, the notion of coverage employed here can be applied to those edges in the TIN which are classified as falling within a resource's visibility cover if the resource searches one end vertex. The following generalizations of the weighted vertex cover problem extend this coverage notion in a way more applicable to the SAR problem.

Arkin *et.al.* [4] consider two problems which generalize the weighted vertex cover problem and consider non-negative weights associated with the edges of the graph rather than the vertices. The first problem is the ***Tree Cover Problem*** where a minimum cost tree is required so that all edges have at least one vertex in the tree. If all edge

[4] Polygonal holes are defined as opaque obstacles.

weights are equal, this problem is equivalent to a minimum connected vertex cover. The authors also consider a second problem entitled the ***Tour Cover Problem***. In this problem a tour rather than a tree constitutes the cover. Repeated vertices and edges are allowed within the tour. Arkin *et.al.* show that both problems are NP-hard and do not assume that the cost matrix on the edges satisfies the triangle inequality. Approximation algorithms are given by Arkin *et.al.* to solve both the tree and tour cover problems.

The Tour Cover Problem can be viewed as a Watchman Route Problem where the watchman is required to see every edge by visiting at least one of its end vertices, and to execute this by the shortest closed walk.

In the above problems the definition of coverage has useful applications to some classifications of edges in the SAR model. Generalizations of the problems could incorporate paths rather than tours, and specify subsets of edges which need to be covered. While a tree structure does not describe an efficient search pattern for a single resource, it may provide the framework for a multiple resource deployment, with one resource assigned to search each leading branch.

6.2.5 Additional Search Considerations

6.2.5.1 Communication

The ***Multicovering Problem***, also known as the ***Generalized Set Covering Problem***, is considered by Hall and Hockbaum [84]. The problem addresses coverage in models which contain a factor of unreliability, by enforcing multicoverage. Applications of the problem include the determination of the placement of emergency service facilities, placement of radio communication equipment, crew scheduling, and the problem of 'terrain masking' encountered by military radar surveillance. For example, in the placement of radio communication equipment the solution of the problem can ensure that a particular region is covered by more than one repeater so that communication does not break down. The problem is NP-hard as it is a generalization of the set covering problem.

The main application of the multicovering problem to the SAR problem is in the area of radio communication. A solution to the multicovering problem can ensure that all regions of the terrain are covered more than once to prevent communication loss due to the uncertainty of conditions and equipment performance.

6.2.5.2 Sub-base Placement

A problem in the literature, which has application to the placement of sub-bases in the SAR operation, is the ***Multi-Weber Problem***. This problem finds n locations which minimize the summed distance from a number of fixed points, whose locations are known.

Positioning n sub-bases in this manner would ensure fast response times to any sector of the search region graph, when the set of fixed points represents the set of vertices of the graph. Rosing [144] presents a set covering formulation of this problem based on convex hulls and solves small problems to optimality.

6.2.5.3 Observation Points

Viewpoints located at key positions on the TIN could serve as useful observation points, particularly in the instance of a moving subject. Goodchild and Lee [76] formulate two coverage problems over a TIN. The first problem is a location set-covering problem with the objective of minimizing the number of facilities needed to see the entire surface of the TIN. The second problem is a maximum covering location problem and has the objective of maximizing the area covered by a fixed number of facilities. Due to the type of terrain composing a search area, the limited visibility of a search resource and the limited number of resources available, the second problem is more applicable to locating search resources at observation points. Due to the underlying subject location probability distribution it may also be preferable to position resources so as to cover regions of high POA only.

Goodchild and Lee consider only the vertices of the TIN as possible facility location candidates, as the set of vertices of the TIN are assumed to include all peaks of the surface modelled by it. This limits the possible locations of facilities, hence making the problem tractable. The authors [76, page 18] also note that "the TIN representation ensures that the field of view of each vertex is almost certainly unique, and that dominance almost never exists. The search for coverage must therefore consider all vertices."

6.3 Vertex Routing

Vertex routing problems consider the problem of finding a route through the vertices, or a subset of vertices, of a given network. Such problems have application to the SAR problem if a tour through a set of required vertices, such as huts, hazards or observation points, is desired and if a route is desired for a transportation vehicle transferring search resources to the starting vertices of their search paths, or collecting resources from the termination vertices of their path. More significantly though, the problem can be applied to a search phase where only triangular regions of the search region graph are searched and the dual of the graph is utilized such that each triangular region is represented by a vertex located at its centre of mass. When considering a region as a vertex, its spatial quality must be accounted for; hence the time to travel between triangular regions is small in comparison with the time required to traverse the region face. This application necessarily requires careful thought as to an appropriate definition of inter-region traversal as well as methods of traversing the triangular region in a searching pattern.

6.3.1 Single Route Problems

The most commonly utilized vertex routing problem is the ***Travelling Salesman Problem (TSP)*** where a salesman must visit a set of cities represented by vertices via the least cost route, beginning and ending that route at the same vertex [130, page 496]. The TSP is an NP-complete problem. Many variants on this problem have been proposed over the years; we now examine some of those relevant to the SAR problem.

Golden *et. al.* [73] present the ***Time-Constrained Travelling Salesman Problem*** where each pair of vertices has not only a travel time associated with it but also a net profit. The problem seeks to find a subtour over the graph which begins and ends at a given origin vertex, maximizes profit and has a length which does not exceed a given time constraint.

A general ***TSP with precedence relations*** is considered by Lokin [109] where conditions exist to specify which vertices must be visited prior to other vertices. In this version of the problem the starting and end vertex of the tour is not required to be the same. The objective is to minimize the cost of the resulting path or tour.

The ***Clustered Travelling Salesman Problem*** [109] addresses the problem of visiting a cluster of vertices contiguously, when the vertices requiring visitation are initially clustered into subsets. This problem is then extended by Lokin [109] to one where precedence relations exist between the vertex clusters specifying the order of visitation. This priority version of the TSP is addressed by Fiala Timlin and Pulleyblank [51] who consider an application of precedence constraints to the scheduling and routing of a helicopter which services oil platforms. The oil platforms are grouped into priority classes where all platforms in a given priority class must be visited before any platforms in a class of lower priority are visited. The objective of the problem is to minimize the total flying distance of the helicopter such that all required platforms are visited. The particular problem considers pick up and delivery at some platforms, and is extended to consider a capacity constraint on the helicopter.

The ***Multiobjective Vending Problem*** is presented by Keller and Goodchild [97]. This problem is formulated such that each vertex has an associated reward and a cost is incurred with each inter-vertex traversal. The problem seeks to determine a tour through a subset of vertices with maximum reward and least cost, and hence is a multi-objective problem. A similar problem is defined by Malandraki and Daskin [112] as the ***Maximum Benefit Travelling Salesman Problem***; in this problem a vertex may be utilized more than once, but no additional benefit is gained.

The ***Orienteering Problem*** introduced by Tsiligirides [167] can be viewed as a special case of the Multiobjective Vending Problem when there exists an upper bound upon the cost of the tour. Hence the problem seeks the tour which maximizes the total prize value within the cost constraints, which has particular application to score orienteering.

The particular problem of searching triangular regions in the SAR problem can be seen as being related to the Time Constrained TSP and the Orienteering Problem when constraints upon search path duration are considered. If these constraints are relaxed and considered as an objective, the problem is similar to the Multiobjective Vending Problem. If benefit is defined in terms of the POS value predicted from searching each region, then the SAR problem seeks to maximize this benefit as for the Orienteering Problem. Unlike the Maximum Benefit TSP, additional benefits can be obtained with successive searches of a region. The SAR problem also differs from the Time Constrained TSP variant in that the benefit is associated with a vertex rather than a pair of vertices.

If the SAR problem is formulated in terms of precedence constraints, such that some search regions must be searched prior to others, the precedence versions of the TSP can be invoked. In particular, if regions are clustered into sets of similar priority, a clustered TSP with precedence constraints exists. When k search resources are available, a useful approach could be to cluster k tasks of similar duration and priority together, and then find a solution to the TSP with precedence relations for each resource, selecting a vertex from each cluster to allocate to each resource.

6.3.2 Multiple Route Problems

A multiple version of the Orienteering Problem is addressed by Butt and Cavalier [13] as the ***Multiple Tour Maximum Collection Problem (MTMCP)***. The objective of the problem is to determine a set of tours which maximizes the total value of reward collected from vertices, within a given time limit. The problem can be seen in such applications as score orienteering conducted by teams or in vehicle routing where a reward is obtained from each customer visited. The MTMCP is NP-hard as the Vehicle Routing Problem (VRP) and the TSP are special cases of it.

The MTMCP is most relevant to the problem of search coverage of the triangular regions when the dual of the search region graph is utilized. The maximization of reward within the MTMCP is analogous to the maximization of POS in the search, with the reward at each vertex being equal to the POS value predicted from the search of each region. Due to resource and time constraints it is usually not physically possible to search the entire search area, hence a selection of the vertices must be chosen for searching, as in the MTMCP. The SAR problem has a similar objective of maximizing POS in a resource constrained situation.

Multiple tours are required, one for each resource, and the problem is time constrained. The time constraint is seen in the objective of finding the subject as quickly as possible and also in the duration of each search resource's path as a precaution to prevent fatigue. Time constraints in the SAR problem can then be considered in both planning shifts for resources and in terms of total searcher hours available for a set phase of the operation. Tours are appropriate if a resource begins and ends its route at the search base or some sub-base located on the TIN. However, the MTMCP does not order traversal on precedence or priority criterion, and is static, not dynamic, in nature.

6.4 Arc Routing

Arc routing problems involve the traversal of arcs in a network where vertices in the network are used merely as entry and exit points for arc traversal. Arc traversal may start or finish at an intermediate point on the arc, and routes may be generated by creating cycles or paths. Routing may be restricted to subsets of arcs when not all arcs require service. While there exists extensive literature in this area, not as many results are available in comparison to the widely researched vertex routing problems. This lack of attention is largely due to the fact that, as Lapalme *et.al.* [101, page 85] state, "arc-routing problems are particularly difficult". Some practical applications of this field include postal delivery services, rubbish collection, and the operation of automated guided vehicles in factories [10, 151]. Arc routing problems have application to the SAR problem in the reconnaissance search phase if edge only traversal is specified.

6.4.1 Single Route Problems

The most common occurrence of arc routing is called the ***Chinese Postman Problem(CPP)*** and is attributed to Kwan Mei-Ko, a Chinese mathematician. The CPP requires a route to be found through a network of edges, such that all edges are traversed at least once and the total distance traversed is minimized. The postman begins and ends at the same vertex in the graph. The case where all edges of the graph are undirected is referred to as the undirected CPP, the instance where all the edges are directed is the directed CPP, and where only some edges are directed the problem becomes a mixed CPP.

The undirected CPP can be solved in polynomial time as shown by Edmonds and Johnson [46]. The directed CPP results when arcs can only be traversed in one direction, *eg.,* one-way streets; when all arcs are to be traversed, this is simpler than the case where the graph is undirected, and may also be solved in polynomial time. However, it is possible that no solution for the directed CPP may exist. This will occur when there exists a subset of vertices in the graph, which have no arcs between the vertices within the subset and any other vertex not in that subset.

In some instances the graph on which the CPP is defined will be mixed, containing both directed and undirected arcs. Papidimitriou [131] shows that the mixed CPP is NP-complete. Papidimitriou also investigates instances where the underlying network is physically constrained due to some real-world application. In particular, he considers cases where the arcs form a planar graph with vertices of degree less than four, and all arc costs are equal to one. In these instances the problem is still found to be NP-complete. The CPP over a mixed graph may also be unsolvable as in the directed case, and for the same reason.

The ***Rural Postman Problem(RPP)***, first mentioned by Orloff [124], is the CPP where only a selection of the arcs require traversal. The best route is determined over these required arcs with the remaining arcs in the network being used solely as connections to establish the route with least cost. In instances where all the arcs are required, or a connected subgraph exists among the required arcs, the problem reduces to the CPP.

Lenstra and Rinnooy Kan [107] prove that the RPP is NP-complete over both undirected and directed networks. The RPP is also NP-complete over mixed networks as the CPP is a special case of the problem. However, Orloff [126] states that these results are misleading as the additional complexity of the RPP over the CPP is directly related to the number of disconnected components present. Hence, if the network can be transformed so as to reduce the disconnectivity of the sub-components, then the complexity of the problem can be reduced.

The ***Windy Postman Problem(WPP)*** was first researched by Minieka [118]. The WPP is a CPP where the cost of traversing an arc is dependent upon the direction that that arc is traversed; this may be due to gradients existing upon the arcs or resistance factors such as travelling into the wind. Hence, the WPP requires a decision as to which direction to traverse arcs in to achieve a route of least cost. This is a problem of much greater difficulty and as such has received less research attention. The previously examined cases — undirected, directed and mixed CPPs, can all be considered special cases of the WPP. As with the previous variations of the CPP, the objective is to determine a route over all the arcs in the network with minimum cost, such that all arcs are serviced. The resulting tour is a diorientation of a postman tour.

Minieka [118] shows that the WPP is a special case of the mixed postman problem. As the mixed postman problem is NP-complete, so too is the WPP. However, Guan [82] shows that the WPP is of polynomial complexity when the underlying network contains symmetric cycles, *i.e.*, each cycle in the graph has an equal cost, independent upon the direction of traversal around that cycle. Win [172] shows that this polynomial complexity result also holds for the special case where the WPP is defined on an Eulerian graph.

The ***Minimum Cost Eulerian Orientation Problem***, a WPP with the additional restriction that each edge must be traversed exactly once, is addressed by Guan and Pulleyblank [83], and Win [172].

Dror *et.al.* [45] extend the CPP to consider precedence relations on the arcs, *i.e.*, cases where the arcs require traversal in a specified hierarchy of priority classes. Applications of this problem include snow ploughing and waste collection. Two precedence relations are considered by Dror *et.al.*. The first is linear, such that no arc in the next priority class can be traversed before all arcs in the previous class have been traversed. This problem can be solved in polynomial time for connected subgraphs of arcs having the same priority class. The second relation is general, where the arcs are partitioned into disjoint subsets with a weak partial ordering relation upon these. Dror *et.al.* prove that the CPP under a general precedence relation is NP-complete via a transformation to the TSP.

The ***Maximum Benefit Chinese Postman Problem (MBCPP)*** is introduced by Malandraki and Daskin [112]. The MBCPP defines an extension of the CPP where a benefit is associated with each traversal of an arc in the graph and the restriction that each arc must be traversed at least once, is removed. The objective is to determine a tour which maximizes the total net benefit obtained. The authors assume that additional benefits gained by traversing an arc more than once decrease as the number of traversals increase (decreasing marginal benefits). The net cost of traversing an arc for the n-th time is taken as the negative of the net benefit. Hence the problem objective can be equivalently expressed as minimizing the total net cost of arc traversal.

If the assumption of decreasing benefits with each arc traversal does not hold, an additional constraint is added to the formulation to ensure that the arcs are selected in the correct sequence. Applications of the MBCPP cited by Malandraki and Daskin include the snow-ploughing (or snow-salting) of street networks, where additional benefit is obtained with each traversal. Benefits are taken as user-inputs and can be defined so as to weight arcs of higher priority, such as arterial streets. By removing the constraint that all arcs must be traversed, those arcs of lowest net benefit may remain untraversed while resources are applied to those arcs of higher net benefit.

Extensions to the MBCPP proposed, but not solved, by Malandraki and Daskin include: the addition of a budget constraint on the length of the tour; a bi-objective problem where a weighted objective maximizes the net benefit of the tour and minimizes the length of the tour; a multiple vehicle version where k tours pass through the depot;

a secondary objective of minimizing the variability or range of the length of k tours; and a temporal version with application to the ploughing of roads while it is still snowing, where the time between successive ploughings becomes important.

When considering edge traversal of the search region graph, the two arc routing problems which are the most related are the Rural Postman Problem (RPP) and the Windy Postman Problem (WPP). The RPP, in considering only a subset of edges in the graph which must be visited by any route, relates to a particular search phase where only edges of a given visibility class or priority rating need be searched explicitly. The WPP is applicable as it is formulated over a network where edges have a differing cost depending upon the direction in which they are traversed. In the SAR problem, if a non-level gradient exists there is a differing time taken to traverse an edge in the up-hill, compared to the down-hill, direction.

The extension of the CPP to incorporate precedence relations on the arcs is also applicable when some edges in the search region graph are identified as hazards or as having very high, relative POA values. In these instances it is desirable to search such edges prior to extending the search to other edges in the graph. Such cases would require edges of similar priority to be clustered into a single priority class.

When the predicted POS value of the search of an edge is regarded as the benefit of traversing that edge, the MBCPP can be seen to have application to the SAR problem. However, the MBCPP is defined such that successive traversals of an edge result in additional benefits. This is not true for edge searches of the search region graph if the POD level of an edge search is set at 1.0 and the subject is stationary. However, the POD level of a clue will be less than 1.0 so additional searching of an edge may still reveal new information in the form of clues. Benefit in this instance can be framed in terms of an increase in knowledge. The re-searching of edges is also likely to result in decreasing marginal benefits. The SAR problem is similar to the MBCPP in that not every edge need be searched and edges having low POS values can be excluded in favour of edges having higher perceived benefit.

The multiple extension of the problem proposed by Malandraki and Daskin with the secondary objective of minimizing the variability of the resulting tours is more applicable to creating routes for multiple resources under an equity objective. A temporal version is also desirable due to the dynamism inherent in the SAR problem.

While an edge-searching only, reconnaissance phase of the SAR problem exhibits features of the single edge routing problems described, it is more accurately addressed by the multiple edge routing variants of these problems.

The notion of 'servicing' an edge in arc routing can be extended to the SAR problem where POD can be viewed as the level to which a region is 'serviced'. Hence, regions can be searched (or serviced) to differing degrees dependent upon resource and time constraints, and the level of expected benefit. It would also be possible to stipulate a minimum service level requirement where every region was searched to this level and some were searched above this level.

The concept of **deadheading** in arc routing literature is generally defined as the instance where an arc is traversed more than the number of times that it is required to be serviced. Deadheading is, therefore, an indicator of inefficiencies within the network and the objective is to minimize this value. In the SAR problem, the repeated traversal of an edge or region is not necessarily regarded as inefficient. It may be strategically valid to re-search an area, especially one of high priority, in order to increase the probability of detecting the subject or to increase the state of current knowledge. Often advantage can be gained in re-searching, particularly when searching with different resources or searching from a differing angle than the one previously taken. A change in light and travel direction can often allow detection of previously obscured or undetected objects. We have hence re-defined the measure of inefficiency for the SAR problem to be the instance where a single searcher, or team, repeats a task in the same direction in which they previously executed it.

6.4.2 Multiple Route Problems

When multiple routes or paths are desired, k-person routing problems are described. As Frederickson *et.al.* [54, page 179] state: "these problems reflect more of the flavour of real world problems than 1-person problems."

The extension of the differing versions of the CPP to one where multiple routes are required is referred to as the ***k-person Chinese Postman Problem (k-CPP)***, where k = the number of postmen. The problem is to traverse each arc in the network with exactly one postman, in such a way that k tours are created which begin and end at the same vertex, and the total distance over all the tours is minimized. An extension to the k-CPP is made by Frederickson *et.al.*, who consider the problem with the objective of minimizing the length of the longest single tour. Pearn [133] defines this problem as the ***Minimax k-CPP***. The extension to k routes complicates the problem considerably and the literature into this area is relatively sparse in comparison to the single postman problem. Frederickson *et.al.* show that the k-CPP is NP-complete for $k > 1$.

The k-RPP and k-WPP are similarly defined as extensions of their respective single route problems, involving the determination of k routes over the given network. As both the RPP and WPP are NP-complete, their multiple route versions are also NP-complete.

The ***Capacitated Arc Routing Problem (CARP)*** is defined upon an undirected network as the problem of finding routes which pass through a central depot which traverse all arcs but do not exceed the capacity of each vehicle. Each arc has a positive demand associated with it. The total objective is to minimize the combined routing costs. There exist a number of applications of the CARP including the routing of school busses, snow ploughs, rubbish collections, and the reading of electric meters [15, 151]. A ***Capacitated Chinese Postman Problem (CCPP)*** is similarly defined by Christofides [18] but differs from the problem definition of a CARP in that all arc demands in the CCPP must be greater than zero.

Golden and Wong [74] have shown the 0.5-approximate CCPP[5] to be NP-hard. Hence, the 0.5-approximate CARP is also NP-hard as the CCPP is a particular instance of a CARP. Lower bounds for the solution of the CARP are developed by Benavent *et.al.* [11] and Pearn [132], and extend those of Golden and Wong [74], and Assad *et.al.* [6].

Assad *et.al.* [6] note that the capacity restriction of the problem is crucial in defining the difficulty of the problem and that further complexity implications arise when only a subset of arcs, forming disconnected subcomponents, require servicing. The authors also show that the CCPPs exhibiting some special structure are polynomially solvable. In particular, a CCPP conducted over a complete network where arc demands are less than W/n if n is odd, and $W/(n-1)$ if n is even, is solvable, where W represents the capacity of each vehicle and n represents the number of vertices in the network. By restricting arc demand, the computational increase resulting from the packing aspect of higher demand costs is avoided. Special solvable classes include a single path network and a cycle network where the arc demands are all equal.

Golden and Wong [74] show how the CARP relates to other routing problems in the literature. In particular, if one vehicle has capacity to cover the demand in a network, then the CPP is described. If one vehicle has this capacity and positive demand occurs only for arcs in a specified subset, then the CARP describes the RPP. Similarly, the CARP can be shown to have the TSP and the VRP as special cases of it. This can be seen when a vertex requiring service is split into two vertices joined by a zero length arc having demand equal to the original vertex demand.

Roy and Rousseau [145] present the ***Capacitated Canadian Postman Problem***, a version of the CARP where instead of capacities existing on each defined route, a time constraint exists. The routes must begin and end at the same point (which may differ between routes) and cover all arcs requiring service. Roy and Rousseau define the problem over a graph which is undirected, connected and even, and where all arcs require service. Each arc has a differentiated cost defined, based on whether the arc is

[5]The α-approximate version of a problem is that of determining the solution with cost at most $(1+\alpha)$ times that of the optimal solution [48].

to be serviced or traversed without service.

The Delaunay TIN with $(3n - H - 3)$ edges constructs a sparse network of edges even when all edges are required, *i.e.*, all edges have a positive demand in the definition of the CARP. In addition, the search region graph can be viewed as a perverse network structure if the edges are not only sparse but also have large demands, *i.e.* if the ratio of the time to search an edge and a resource's available hours is large. This will occur if the three dimensional length of the edge is great and/or the terrain is difficult.

Of the multiple arc routing problems in the literature, the problem most relevant to the edge routing of resources in the reconnaissance phase of the SAR problem is a mix of the k-RPP and k-WPP *i.e.*, a windy version of the k-RPP. The 'windy' nature of terrain traversal is then incorporated for multiple routes over a selected subset of edges. The Capacitated Canadian Postman Problem in its formulation of a time constraint upon each route is also representative of the SAR problem where a resource cannot be allocated search tasks beyond an upper time limit in order to prevent fatigue. In general the CARP class of problems, though static, are related to the SAR problem when a resource's available search hours are viewed as their capacity, which cannot be exceeded.

In all arc routing problems the objective of minimizing the cost of arc traversals does not incorporate the objective of maximizing the time to detection of a search subject. Hence, an alternative objective needs to be formulated or implicitly incorporated by the identification of a subset of regions which, when searched, will meet this objective.

6.5 Arc and Vertex Routing

Orloff [124] investigates arc and vertex routing, and defines a ***General Routing Problem (GRP)*** as a generic approach to solving them both. The GRP is defined as the problem where each arc in a required subset of arcs, and each vertex in a required subset of vertices, is traversed via a total route of least cost. The CPP, RPP and TSP are therefore special cases of this problem. Orloff distinguishes between vertices which represent the intersections of arcs, and vertices which require a specific service.

Orloff [125] extends the GRP to the instance where k 'vehicles' are required to service a network. This multiple version is termed the k-GRP. The problem determines the minimum cost k cycles which traverse a set of required arcs and visit a required set of vertices. The problem is formulated for a set of origin vertices and a common destination vertex, with each cycle containing an arc (assumed to be of zero cost) from the destination vertex to the origin vertex.

The k-GRP is the most general routing approach of those studied and as such is the most relevant to the routing element of the SAR problem.

6.6 Partitioning

The Arc Partitioning Problem (APP) is introduced by Bodin and Levy [12]. The APP is formulated over a service network (a subset of a complete travel network) and has applications in the reading of household electricity meters, rubbish collection and postal delivery, where only a subset of arcs are serviced in a single day. Every arc of the service network has a service time and the arcs are undirected. The objective of the APP is to partition the network into connected subnetworks such that the workload of each partition is balanced and each arc is assigned to exactly one partition. The partitioning is considered to be balanced if the workload of each partition lies between a given upper and lower bound. A penalty function is employed by Bodin and Levy to cope with violation of these bounds; the penalty for violating either bound can be varied to represent their relative severity.

Workers are transported to a starting point within each partition rather than being assigned a route from a common depot. A problem related to the APP which does determine a route through each partition from starting depots is the ***Arc Oriented Location Routing Problem***, also addressed by Levy and Bodin [108]. The APP is transformed to the Arc Oriented Location Routing Problem by the addition of a supervertex which acts as a dummy depot. Additional arcs with zero cost are added to connect the supervertex to a vertex in each partition. In this problem, objectives of minimizing deadheading time and minimizing the number of depots are also employed.

The ***Exact Partitioning Problem*** is to determine a collection of vertex-disjoint partitions of minimum cost and size k, such that all vertices are covered. The partitions studied by Goemans and Williamson [70] consist of trees, cycles or paths. The minimum-weight perfect matching problem is an exact partitioning problem when k=2; the TSP, hamiltonian path problem and the minimum-cost spanning tree problem, can all be considered generalizations of the exact partitioning problem. The ***Lower Capacitated Partitioning Problem*** is defined as for the exact partitioning problem, except that instead of each component being required to have k components, each component must have at least k components. The problems are NP-complete.

The APP is one of few problems in the literature which directly considers the objective of workload equality, an objective which is central to the problem of tasking search resources. Multiple tours are sought, which are separate and distinct, hence minimizing the redundancy of the service workers. Each arc has an assigned service cost which could be viewed as the time taken to search that edge in the SAR problem. Each tour is minimized in terms of the length of this service time and hence fits with the objectives of the SAR problem of minimizing the search time of a given resource's path, and minimizing the redundancy of resources overlapping one anothers' paths. The APP also transfers the workers to a point in each partition. This has application to the SAR problem where searchers do not begin their assigned tasks from the search base but are transferred to their respective search regions via a helicopter.

The exact partitioning problem ensures that partitions are disjoint which, if applied to the SAR problem, would ensure that the tasks allocated to individual search resources would not overlap, removing redundancy from the equation. The problem has the objective of minimizing cost which can be adapted to finding paths which minimize time. However, the specification of a size k does not represent the problem encountered in SAR, as regions in the TIN are not similar in work effort required so solving the problem with respect to this parameter would not equate with work balance objectives.

6.7 Scheduling

Search regions are sequenced for searching and then allocated to available search resources; the sequencing of these tasks is therefore critical to the outcome of the search. Sequencing of search regions cannot, however, be separated from the SAR environment as it is inherently linked with the problem of assigning and routing the search regions. Sequencing or scheduling the search tasks contains obvious parallels to production (machine) scheduling applications, which are addressed in a large body of literature.

A single machine model is appropriate when we consider that the search of a region can be viewed as analogous to a region being processed by a single search resource (machine). When a region is searched by more than one resource, in consecutive sequence, a multiple machine application can be considered. In the production environment such a sequence is explicitly known and defined by the actions upon the product which occur in a set sequence. In the SAR environment this sequence will be determined dynamically depending on resource availability and changing knowledge.

Within the literature on production scheduling applications a number of objectives are considered. These include minimizing the total time elapsed (make span), minimizing the idle time of resources and attaining a uniform rate of activity. These three objectives are adaptable and applicable to the SAR context. It is desirable that the total time of the search operation be minimized, and that the idle time of search resources be minimized. If it is assumed that the number and type of search resources remains constant throughout the operation then a uniform rate of activity is also desired. Alternatively, search activity can be concentrated in the periods where labour is available in a proportionate ratio.

Pre-emption is allowed within the search environment unlike many production scheduling applications. Cancellation of search tasks is allowed when new information indicates that other search regions are now of higher priority. Such regions are searched before others which have previously been scheduled, hence overriding the initial sequence. This is similar in many ways, and aptly illustrated by, the sector laddering approach utilized in the UK. Scheduling in the SAR problem is hence dynamic rather than static.

The notion of tardy jobs which are rejected if not processed in time in a production context are not applicable to the SAR context, as regions which should be searched do not become redundant beyond a given time frame. Search tasks may become redundant upon the receipt of new information but these are distinguishable from 'tardy jobs'.

6.8 The SAR Problem

In summary, each phase of the search operation can be seen to be both related and different from existing problems in the literature.

6.8.1 Phase One Search

The Phase One search is termed a hasty or reconnaissance search. It is designed to be executed quickly and to cover regions of high probability. Generally the path of least resistance is taken [103] including marked tracks, ridge lines and stream beds. It is primarily for this reason that we restrict the searching in this phase to edges in the search region graph which represent linear features present in the search area. Not all edges in the graph need to be searched during this phase, hence a required set of edges is defined based upon a priority ranking scheme.

During this phase any huts in the vicinity will generally be checked to ascertain whether the subject has moved through there, at what time, and whether in fact trip intention details have been recorded. Huts are represented as vertices on the search region graph, hence those huts requiring to be visited can be labelled as required vertices.

The objective of this search phase is then to find the k paths of minimal cost which 'cover' all the required edges and vertices and maximize POS returned, where $k=$ the number of search resources utilized during this search phase. The paths should be completed within a given time frame to meet the requirement that the search phase be executed quickly. Paths may include edges which are not in the required edge set if so doing expedites the minimal path cost objective. It is also desired that work effort between resources should be approximately equal. In this respect the problem is similar to the Arc Partitioning Problem of Bodin and Levy [12].

The basic Phase One search problem requires all search resources to begin searching from a common vertex — the search base. However, search paths may finish at any vertex of the search region graph. Bilevel variants on this problem enable resources to be transported to the start of their path by helicopter or 4-wheel drive, thus enabling paths to begin at any point on the graph which is accessible by the available transport mode. Such transferral of resources is similar to both the Arc Partitioning Problem addressed by Bodin and Levy [12] and the Canadian Postman Problem of Roy and Rousseau [145]. Termination vertices of search paths may need to be selected in accordance with the desired qualities of starting vertices for the resource's next search assignment.

As a specified edge and vertex set are defined that must be included in the search paths, the problem can be formulated as a windy-version of a k-General Routing Problem [125]. The set of required vertices can be transformed into edges such that edge routing alone can define a solution to the original problem. This can be achieved by splitting a vertex into two vertices joined by an edge of zero length, as proposed by Golden and Wong [74]. Under this transformation, or in instances where the required vertex set is empty, the problem can be described as a windy-version of the k-Rural Postman Problem. If edges must be traversed in a priority ordering then the Chinese Postman Problem with precedence relations on the arcs is described [45].

As search paths can be defined to not only traverse required explicit edges, but also 'cover' required implicit edges, the problem displays similarities to those in the coverage routing literature. Within the model an implicit edge is covered if their respective required vertices are present on a search path. These required vertices can be defined as standard required vertices in the context of covering problems [9]. Alternatively a similar representation to the Shortest Covering Path Problem can be adapted for a multiple resource, windy-version, where the coverage measure is defined by the visibility measure on an edge, and vertex coverage (edge end vertices) is specified.

As the problem can be represented as essentially equivalent to problems of known NP-complete complexity the problem defined by the Phase One search is also NP-complete. It is unique to other problems in the literature in that we are unaware of any k-Rural Postman Problems and solution approaches defined as windy-versions, nor any k-Shortest Covering Path Problem formulations, let alone those with windy edges. Additional restrictions on the time limit of search paths and work balance among resources uniquely identify the SAR problem.

6.8.2 Phase Two Search

The Phase Two search is termed the efficient or general search. A combination of edge and triangular region searching is employed as searching is no longer restricted to the linear features of the terrain. If the problem is formulated as a vertex and edge routing problem then it can be defined as a k-General Routing Problem [125] with windy edges and as such is NP-complete.

6.8.3 Phase Three Search

If in a Phase Three search it is assumed that there do not exist edges that require searching, then the problem reduces to determining k search paths which search a set of required triangular regions so as to maximize a defined objective.

Triangular regions can be searched by a variety of search patterns, as will be outlined in Chapter 8. An individual search pattern achieves a set POD level and is described by the beginning and end points of the pattern (initially assumed to be vertices of the region), the spacing between the searchers and the direction of movement of the resource.[6] Depending on the orientation of the search pattern a different traversal time results. It is assumed that each triangular region is searched by only one search resource (if a triangular region is too large to allow it to be searched by a single resource within the time constraints it is partitioned into regions which can be). The path of a search resource is defined by the regions which it is allocated to search and the least time connections between the beginning and end points of the search pattern determined for each region.

The triangular regions can be represented as vertices by solving the problem over the dual of the search region graph; the dual being formed by placing a vertex at the centre of mass of each triangular region and connecting these through the edges of adjacent regions. When represented in this way the problem can be viewed as a k vertex routing problem with a set of required vertices which must be visited. As there exists a predicted POS return at each vertex the problem can be seen to be related to the Multiple Tour Maximum Collection Problem addressed by Butt and Cavalier [13], where the prize collected is equal to the POS value of the vertex which is to be maximized with respect to the time constraint. Additionally, a set of vertices is required to be searched.

The shortest paths between the exit vertex of one region and the entry vertex of the next region in the search path can be modelled on the primal search region graph by the edges of the graph, as this structure is already in place and is hence computationally inexpensive. This also allows edges which have not yet be searched to be included in the access paths connecting regions — even if this results in some relaxation of the shortest path being utilized. Alternatively, the shortest paths could be found via planar unfolding or the discrete geodesic methods of the weighted region problem as addressed by Mitchell [119]. The costs of the edges in the dual search region graph which connect two vertices representing triangular regions, equal the cost of the shortest path connection between the two regions in the primal search region graph.

[6] A ground search resource comprises a team of searchers of fixed size.

If the required triangular regions are to be searched according to a priority ordering then the problem is a multiple path variant of the Clustered Travelling Salesman Problem [51].

6.9 Problem Formulation

Golden and Wong [74, page 308] state that "an NP-completeness proof is a strong indication that the problem is intractable". As NP-complete problems in the literature can be shown to be special cases of the SAR problem, the general SAR problem is also NP-complete. Hence the mathematical formulations now presented serve to facilitate understanding of the problem and its relationship to other problems in the literature, rather than to offer a model to be solved. Initially differing objectives of the SAR problem are considered and then a formulation is presented for the apportioning aspect of the problem. Two mathematical formulations of the SAR problem under edge searching only are then presented. The first formulation follows a routing approach; the second is presented from a coverage-routing perspective. A further formulation is then presented for the SAR problem when only triangular regions are searched. The formulations are static and are framed for a specific initial allocation of search tasks to available resources.

6.9.1 The Objective Function

All decision-making within the SAR operation is focused on the following key issues which should not be compromised:

- to locate the subject in the quickest possible time;
- to maintain the safety of search resources;
- to operate on a priority basis;
- to develop efficient paths with respect to the time taken to execute them; and
- to accurately calculate the relevance of clues and new information to provide the best platform for informed decision-making.

In particular, a number of alternative objectives for the SAR problem can be considered. These include:

- minimize the time to detection of the subject;
- maximize POS in each planning time period (when the problem is divided into lengths of a defined horizon length);

- maximize cumulative POS up to a certain time period;
- maximize the rate of POS obtained per unit of time;
- minimize the time required to complete a set level of coverage over a given set of regions;
- maximize a weighted level of coverage where coverage is weighted more highly for regions rated as being of higher priority;
- maximize the size of the search area coverable within the planning period;
- maximize the coverage of each search region;
- minimize the cost of not searching some regions;
- minimize the length of each search resource's path;
- minimize the variability of path duration between search resources;
- minimize the unallocated search hours of each search resource;
- minimize the non-necessary overlap of search resources' paths; and
- minimize the time difference between when a region is searched and the time position it occurs in the original, desired event sequence.

Among this list there exists a hierarchy of objectives. For instance, the objective of minimizing the time to subject detection effectively sums up the entire objective of the search, while the remaining objectives seek to expand upon this and quantify it in some way. These include maximizing the chance of detection, maximizing the utility of a given time period, minimizing redundancy in resource allocations or maximizing resource utility, and seeking resource allocations which strive for equity among resources. Some of these objectives are conflicting, such as the objective of maximizing coverage and size of a search area. Some objectives, such as the objective of minimizing path length variability among resources, can be expressed either as a hard constraint within the problem formulation or as a soft constraint in the form of a penalty function within the objective function. A further example of such a soft time constraint which may be violated at a penalty, would be to allow the length of a search resource's path to violate the resource's useful search hours. The penalty incurred can be modelled as a decrease in the POD level of the search as the resource fatigues. The motivation behind the use of such a constraint is to both increase the POS and to reduce the opportunity cost which arises when a search resource distant from a region would have to be allocated to search a region which a resource near, or at, their search hour limit could search more quickly. A penalty function can also be considered in relation to scheduling search regions. In this application, if a region of given priority is not searched by a specific time, a penalty would be incurred.

Problem constraints can be subject to update and revision as part of a learning process of how effectively they address the problems at hand. This competence can be measured in terms of the time needed to recompute search paths or the number of assignment changes *i.e.*, the total change from an initial path.

The objectives which seek to maximize POS can be viewed as objectives which override cost and resource equity considerations. POS can also be viewed as a superior measure of maximizing the chance of detection, superseding objectives of maximizing coverage or area searched, as it directly incorporates knowledge of the subject's location probability distribution.

A multi-phase objective function can be developed which alters temporally with respect to distinct search phases, taking into account short, medium and long term decision-making. Structuring the objective function in this way decomposes the problem into subproblems based on time period planning horizons. It would appear that the most practical planning time frame to apply to the SAR problem is that of a half shift of six hours [121].

A number of differing objectives can also be incorporated by formulating a weighted multi-objective formulation, where each contributing objective is weighted by its relative significance. As the SAR problem is dynamic in the most general case, an objective which incorporates time is desirable. For example, the objective of maximizing the rate of POS obtained over time or the objective of maximizing POS over each planning horizon (when the problem is decomposed into problems of a defined horizon length) can be used.

6.9.2 Allocation Formulation

The apportioning problem in the SAR operation involves that of dividing the search area into search tasks of less than T hours in duration, and then loading these tasks onto the available search resources. Eilon and Christofides [47] address loading problems where physical items are loaded into boxes. The authors define a matrix of loading problems defined via three problem objectives (O_1, O_2, O_3) and two problems situations (S_1 and S_2).

The problem situations are defined as:

S_1 the capacity of the available boxes equals or exceeds the total size of the items, and all items can be allocated to a box;

	O_1	O_2	O_3
S_1	1	–	–
S_2	2	3	4

S_2 not all items can be allocated to a box, either because the available capacity is less than the total size of the items, or because the individual item sizes do not allow a viable packing.

S_1 corresponds in the SAR operation to all search tasks being allocated to the available search resources, for the present search phase. This will occur if unlimited search resources are available, however for a search operation of any realistic size, covering a realistic area, this will not occur and situation S_2 will avail.

The three objectives of the problem matrix are:

O_1 box-based objectives:

- **(a)** minimize the number or value of the boxes required;
- **(b)** minimize the volume or value of the unused space of partially filled boxes.

O_2 item-based objectives:

- minimize the number or value of unallocated items.

O_3 combination objective:

- minimize the weighted value of the boxes used and the unallocated items.

In the SAR problem these objectives can correspond to:

- O_1 **(a)** minimize the number[7] of search resources required;
- O_1 **(b)** minimize the total unallocated search hours;
- O_2 minimize the total POS value of the unallocated search tasks;
- O_3 minimize the number of search resources and the total POS value of unallocated search tasks.

The problem defined by S_2 and O_2 is a multidimensional knapsack problem with n boxes. This problem is especially relevant to the urgency and priority issues of SAR embodied in the POS values of the regions to be searched. Objective O_1(a) is not

[7] Alternatively value can be utilized if there exists a cost to purchasing a resource *eg.*, a helicopter.

particularly relevant to the SAR problem as a fixed, limited number of search resources is usually available to the search manager who, rather than seeking to minimize this number, seeks to maximize their utility. As such a useful approach for the SAR apportioning problem would be to combine objectives $O_1(b)$ and O_3 into a weighted objective. We define this objective as:

- O_4 minimize the unallocated search hours of each search resource and the total POS value of unallocated search tasks.

We then create a problem class defined by O_4 and S_2.

In the SAR problem we assume that all search resources have the same capacity. This is represented by useful search hours which equate to T hours for any one search path allocation. Capacity requirements are additive.

A more general problem is also presented by Eilon and Christofides where each item is characterized by several attributes, each with their own weighting. In turn the boxes are also characterized by a capacity weighting for each attribute. An illustration of this in the SAR problem would be to characterize a search task by such attributes as search hour duration and level of difficulty. A search resource could then only be assigned a search task if they have search hours unallocated and they have not yet reached their level of difficulty capacity (either imposed because of limitations on experience or ability, or to make tasks assigned to resources 'equal' in difficulty in order to preserve morale).

The assumption is made that

$$T \geq Q_i > 0 \quad \forall i \in N$$

i.e., the search area is divided into regions which are each able to be searched within T hours.

A static integer programming formulation of the SAR apportioning problem in its most general form, without specific routing constraints, can then be defined.

$$\text{minimize} \sum_{k=1}^{s}(T - \sum_{i=1}^{n} x_{ik}Q_i) + \sum_{i=1}^{n} POS_i(1 - \sum_{k=1}^{s} x_{ik}) \quad (1)$$

s.t.

$$\sum_{i=1}^{n} Q_i x_{ik} \leq T \quad \forall k \in K \quad (2)$$
$$\sum_{k=1}^{s} x_{ik} \leq 1 \quad \forall i \in N \quad (3)$$
$$x_{ik} = 0 \text{ or } 1 \quad \forall i, k \quad (4)$$

where

Q_i = search duration of region i (hours) including access time

s = number of search resources

n = number of search regions

T = maximum search hours allocated to resource k

K = set of search resources

N = set of search regions

$$x_{ik} = \begin{cases} 1 & \text{if search region } i \text{ is allocated to resource } k \\ 0 & \text{otherwise} \end{cases}$$

The objective function is O_4 and the constraints represent:

(2) the duration of the search path of resource k does not exceed T

(3) each task is allocated to a maximum of one search resource

(4) integrality constraint

6.9.3 Edge Coverage Formulation

The following formulation is (loosely) adapted from the notation proposed by Beasley and Nascimento [9], which was initially developed to formulate the single vehicle routing-allocation problem. It was then generalized to form the basis of a unifying framework for a number of coverage and routing problems described in the literature. The main adaptions required are: the necessary alterations to cope with edge coverage rather than vertex coverage; the extension to multiple paths; the inclusion of edge length and edge visibility measures; a priority set, P, of edges which must be searched; specific edge visibility classes; allowing edges to be traversed more than once, with differing traversal costs if searching or just traversing edges; and the introduction of time constraints on paths.

The formulation is that of an 'open tour' problem during a single search phase. In stating that search resources do not have to return to the search base at the end of their path, we allow resources to then be assigned a new path from their current 'end' vertex in the next search phase, or alternatively to be transported by helicopter or a 4-wheel drive vehicle, to the search base or a new search region. Unlike the formulation of Beasley and Nascimento, no explicit finishing vertex is stipulated. Resources are also permitted to return to the search base during path execution if this is efficient with respect to minimizing the time taken to complete the search path in its entirety. The formulation assumes that all search resources move at an identical speed, and that separate traversal times are incurred depending on whether the edge is being searched or deadheaded.

For the search region graph $G = (V, E)$, where V represents the vertex set and E represents the edge set, the following variable definitions are utilized:

c_{ij} = time required to traverse edge(i,j) if searching the edge (where $c_{ii} = 0$)
$\bar{c}_{ij}$ = time required to traverse edge(i,j) if not searching the edge (where $\bar{c}_{ii} = 0$)
d_{ij} = cost of implicitly searching edge(i,j) modelled as a time delay at the vertex on the path
f_{ij} = cost of not searching edge(i,j) weighted by priority class
l_{ij} = length of edge(i,j)
v_{ij} = visibility measure of edge(i,j)
P = set of edges which must definitely be searched
Q_{path} = set of edges that are included in the paths of the search resources
Q_{imp} = set of edges that are implicitly searched by the search resources
Q_{non} = set of all edges that are not searched by any resource
R_1 = set of all Class 1 edges *i.e.*, edge(i,j) where $l_{ij} \leq v_{ij}$
R_2 = set of all Class 2 edges *i.e.*, edge(i,j) where $l_{ij} \leq 2v_{ij}$
R_3 = set of all Class 3 edges *i.e.*, edge(i,j) where $l_{ij} > 2v_{ij}$
V_r = set of all required vertices which must be on a search path of some resource
x_{ijk} = number of times edge(i,j) is traversed by resource k ($x_{ijk} = 0$ if $i = j$)
T = maximum time limit on a resource's search path
t = minimum time limit on a resource's search path
U = upper penalty weight
L = lower penalty weight
s = number of search resources
K = set of all search resources
n = number of vertices in the graph; 0 = base vertex

$$y_{ijk} = \begin{cases} 1 & \text{if edge}(i,j) \in E \text{ is explicitly searched by resource } k \ (\in Q_{path}) \\ 0 & \text{otherwise} \end{cases}$$

$$a_{ijk} = \begin{cases} 1 & \text{if edge}(i,j) \in R_1 \text{ is implicitly searched by resource } k \ (\in Q_{imp}) \\ 0 & \text{otherwise} \end{cases}$$

$$b_{ijk} = \begin{cases} 2 & \text{if edge}(i,j) \in R_2 \text{ is implicitly searched with resource } k \text{ visiting both edge vertices} \\ 1 & \text{if edge}(i,j) \in R_2 \text{ is implicitly searched with resource } k \text{ visiting one vertex} \\ 0 & \text{otherwise} \end{cases}$$

$$w_{ij} = \begin{cases} 1 & \text{if edge}(i,j) \in E \text{ is not searched; i.e., } (i,j) \in Q_{non} \\ 0 & \text{otherwise} \end{cases}$$

$$u_{ik} = \begin{cases} 1 & \text{if vertex } i \text{ is on the path of resource } k \\ 0 & \text{otherwise} \end{cases}$$

The set of edges is defined by $E = R_1 \cup R_2 \cup R_3$ or $Q_{path} \cup Q_{imp} \cup Q_{non}$.

Additionally Q_{path}, Q_{imp} and Q_{non} are mutually exclusive sets and $Q_{path} \cup Q_{imp} \subset P$, and $R_1 \cap R_2 \cap R_3 = \emptyset$.

Edges not included in P may be searched if efficient with respect to the objective function but edges in P must be searched.

The objective of the problem initially considered, is to minimize:

- the costs associated with explicit searching, with an associated weight in the objective function of λ_{path};
- the costs associated with implicit searching, with an associated weight in the objective function of λ_{imp}; and
- the costs associated with non-searched edges, with an associated weight in the objective function of λ_{non}.

These objective measures are defined, respectively, as:

$$z_{path} = \sum_{k=1}^{s}\sum_{i=0}^{n}\sum_{j=0}^{n} c_{ij}y_{ijk} + \sum_{k=1}^{s}\sum_{i=0}^{n}\sum_{j=0}^{n} \bar{c}_{ij} \max\{x_{ijk}(1-y_{ijk}), (x_{ijk}-1)y_{ijk}\}$$

$$z_{imp} = \sum_{k=1}^{s}\sum_{i=0}^{n}\sum_{j=0}^{n} (d_{ij}a_{ijk} + d_{ij}b_{ijk})$$

$$z_{non} = \sum_{i=0}^{n}\sum_{j=0}^{n} f_{ij}w_{ij}$$

An alternative objective is to utilize a penalty function of search path time violation and workload equity between the search resources. This objective is similar to that utilized by Bodin and Levy [12] for the Arc Partitioning Problem and has an associated weight in the objective function of weight λ_{eq}. Here the time constraints on path lengths are soft constraints, and the degree of violation above and below the given time parameters T and t is penalized by a weight U and L, respectively.

$$z_{eq} = \mathrm{U}\sum_{k=1}^{s} \max\{T(k) - T,0\} + \mathrm{L}\sum_{k=1}^{s} \max\{t - T(k),0\}$$

where

$T(k)$ = the total path costs for a single resource k and is defined as

$$\mathrm{T(k)} = \sum_{i=0}^{n}\sum_{j=0}^{n} c_{ij}y_{ijk} + \sum_{i=0}^{n}\sum_{j=0}^{n} \bar{c}_{ij} \max\{x_{ijk}(1-y_{ijk}), (x_{ijk}-1)y_{ijk}\} + \sum_{i=0}^{n}\sum_{j=0}^{n}(d_{ij}a_{ijk}+d_{ij}b_{ijk})$$

The formulation under the initial objective function is as follows:

$$\min \lambda_{path}z_{path} + \lambda_{imp}z_{imp} + \lambda_{non}z_{non} \quad (1)$$

s.t.

$z_{path} + z_{imp} \leq T$		(2)
$z_{path} + z_{imp} \geq t$		(3)
$\sum_{k=1}^{s}(y_{ijk} + a_{ijk} + b_{ijk}/2) + w_{ij} = 1$	$\forall(i,j) \in E, k \in K$	(4)
$\sum_{k=1}^{s}(y_{ijk} + a_{ijk} + b_{ijk}/2) = 1$	$\forall(i,j) \in P$	(5)
$\sum_{j=1}^{n} x_{0jk} \geq 1$	$\forall k \in K$	(6)
$y_{ijk} \leq x_{ijk}$	$\forall(i,j) \in E, k \in K$	(7)
$u_{ik} + u_{jk} \geq a_{ijk}$	$\forall(i,j) \in R_1, k \in K$	(8)
$\sum_{k=1}^{s} b_{ijk}u_{ik} \geq 1$ and $\sum_{k=1}^{s} b_{ijk}u_{jk} \geq 1$	$\forall(i,j) \in R_2$	(9)
$\sum_{k=1}^{s} u_{ik} \geq 1$	$\forall i \in V_r$	(10)
$y_{ijk}, a_{ijk} \in (0,1)$	$\forall i,j,k$	(11)
$b_{ijk} \in (0,1,2)$	$\forall i,j,k$	(12)
$w_{ij} \in (0,1)$	$\forall i,j$	(13)
$u_{ik} \in (0,1)$	$\forall i,k$	(14)
x_{ijk} integer	$\forall i,j,k$	(15)
path connectivity constraints		
subtour elimination constraints		

where

(1) - weighted objective function

(2) and (3) - hard time constraints on path lengths(optional)

(4) - every edge is either explicitly searched, implicitly searched or not searched

(5) - all edges in the required priority set are searched

(6) - all resource's search paths begin at the base vertex

(7) - explicitly searched edges must be in the path of some resource

(8) - implicitly searched edges in edge Class 1 must have at least one of their end vertices in the path of the resource searching them

(9) - implicitly searched edges in edge Class 2 must have both of their end vertices in the path(s) of the resource(s) searching them — one resource may cover one vertex and another resource the other vertex, or a single resource may cover both of the end vertices

(10) - all required vertices are on a search path

(11), (12), (13), (14) and (15) - integrality constraints

Path connectivity constraints ensure that each resource's search path is connected by entering and exiting each vertex in the path (except for the base and the last vertex) the same number of times. Subtour elimination constraints prevent disconnected subcomponents appearing in paths which are disconnected from the search base vertex. Time and cost are used interchangeably, expressing the same cost measure of edge traversal.

The SAR problem considering only edge traversal is NP-complete as it reduces to the k-Windy Postman Problem when P contains all edges $\in E$, $R_1 \cap R_2 = \emptyset$ (and hence $Q_{imp} \cap Q_{non} = \emptyset$), $\bar{c}_{ij} = c_{ij} \forall (i,j) \in E$, and no time limits on the search paths exist. When P does not contain all edges $\in E$ a windy version of the k-Rural Postman Problem is described. The inclusion of R_1 and R_2 brings in the notion of combined vertex and edge routing as in the k-General Routing Problem which is also NP-complete.

Special cases of the SAR problem exist where data values are appropriately set and where slight changes to the formulation are required. These are now outlined.

6.9.3.1 Special Cases

- There are no limits on path length ($t = 0$ and $T = \infty$), or only an upper or lower time limit exist.

- All edges are to be searched ($P = E$).

- All edges are implicit edges. Here all edges $\in P$ are searched if the required end vertices are searched, *i.e.* $R_3 = \emptyset$. In the instance where all edges $(i,j) \in R_2$ a coverage of required edges can be achieved by a vertex cover of the graph.
- Paths can begin at vertices other than the search base, by specification.
- Paths of all resources end at a specific terminus vertex.
- $\lambda_{non} = 0$ if there are no costs associated with not searching edges.

6.9.3.2 Variants

A closed tour problem can be formulated by the inclusion of an additional constraint and fixing the last vertex in the tour to be equal to the base vertex. Search resources can all begin at differing starting vertices by the inclusion of a variable b_k which defines the base vertex for resource k. The base inclusion constraint (6) would then be altered to

$$\sum_{j=0}^{n} x_{b_k,j,k} \geq 1 \quad \forall K \in k, j \neq b_k$$

6.9.4 Alternative Coverage-Routing Formulation

Alternatively, the edge only search phase of the SAR problem can be formulated based upon the coverage notation of Current *et.al.* [35] for the Maximum Covering Shortest Path Problem.

The same variables are utilized as above with the following variables being introduced to represent the coverage requirements of each edge and vertex:

$$S_{mn1} = \{m \cup n | l_{mn} \leq v_{mn}\}$$

where S_{mn1} represents the set of vertices which cover edge (m,n), and edge (m,n) is a Class 1 edge, *i.e.*, $(m,n) \in R_1$. Here the covering distance S of Current *et.al.* is replaced by the visibility measure of the edge. And

$$S_{mn2} = \{m, n | l_{mn} \leq 2v_{mn}\}$$

where S_{mn2} represents the set of vertices which cover edge (m,n), and edge (m,n) is a Class 2 edge, *i.e.*, $(m,n) \in R_2$.

The main changes in the formulation adapted from Current *et.al.* are that the problem is extended to multiple paths and the single objective is extended to a biobjective formulation where the objectives are to:

- minimize the total path time of all search resources, with an associated weight in the objective function of λ_{tot} where $z_{tot} = z_{path} + z_{imp}$ as previously defined;

- maximize POS, with an associated weight in the objective function of λ_{pos}.

The variables w_{ij} and f_{ij} are no longer required as the total POS of searching is maximized rather than minimizing a cost of non-coverage (min $\lambda_{non}z_{non}$). As in the previous formulation, the path objective and time constraints can be replaced by the workload equity and soft time constraint penalty function as a variant on the problem.

The POS objective is formulated as

$$z_{pos} = \sum_{k=1}^{s}\sum_{i=0}^{n}\sum_{j=0}^{n} pos_{ij}y_{ijk} + \sum_{k=1}^{s}\sum_{i=0}^{n}\sum_{j=0}^{n} pos_{ij}(a_{ijk} + b_{ijk}/2)$$

where pos_{ij} is the POS value predicted from searching edge (i, j).

The objective function and coverage variables are similar to the Maximum Covering Shortest Path Problem presented by Current *et. al.* [35] although no demand exists. Coverage, instead of being measured by demand met, is measured here by the total predicted POS of the edges searched (either explicitly or implicitly). Within the formulation each edge is searched, explicitly or implicitly, at most once.

The formulation is then:

$$\text{maximize } \lambda_{pos}z_{pos} - \lambda_{tot}z_{tot} (1)$$

s.t.

$z_{path} + z_{imp} \leq T$		(2)
$z_{path} + z_{imp} \geq t$		(3)
$\sum_{j=1}^{n} x_{0jk} \geq 1$	$\forall k \in K$	(4)
$\sum_{k=1}^{s}(y_{ijk} + a_{ijk} + b_{ijk}/2) = 1$	$\forall (i,j) \in P$	(5)
$\sum_{i=0}^{n}\sum_{j \in S_{mn1}}(x_{ijk} - a_{mnk}) \geq 0$	$\forall (m,n) \in R_1, k \in K$	(6)
$\sum_{i=0}^{n}\sum_{j \in S_{mn2}}(x_{ijk} - b_{mnk}) \geq 0$	$\forall (m,n) \in R_2, k \in K$	(7)
$y_{ijk} \leq x_{ijk}$	$\forall (i,j) \in E, k \in K$	(8)
$\sum_{k=1}^{s} u_{ik} \geq 1$	$\forall i \in V_r$	(9)
$y_{ijk}, a_{ijk} \in (0,1)$	$\forall i,j,k$	(10)
$b_{ijk} \in (0,1,2)$	$\forall i,j,k$	(11)
$u_{ik} \in (0,1)$	$\forall i,k$	(12)
x_{ijk} integer	$\forall i,j,k$	(13)
path connectivity constraints		
subtour elimination constraints		

where

(1) - objective function consisting of:
- min total cost of edge search paths and max total POS
(2) and (3) - hard time constraints on path lengths (optional)
(4) - all resource's search paths begin at the base vertex
(5) - all edges in the required priority set are searched
(6) - edges in edge Class 1 must have at least one of their end vertices in the path of the resource searching them
(7) - edges in edge Class 2 must have both of their end vertices in the path(s) of the resource(s) searching them - one resource may cover one vertex and another resource the other vertex, or a single resource may cover both of the end vertices
(8) - explicitly searched edges must be on the path of some resource
(9) - required vertices are on some path
(10), (11), (12) and (13) - integrality constraints

6.9.5 Triangular Search Region Formulation

A static routing formulation is developed for the search of triangular search regions only. The formulation draws upon the similarities of the problem to the Multiple Tour Maximum Collection Problem (MTMCP). These similarities are evident when the dual of the search region graph is utilized as the underlying graph, representing the triangular regions by vertices, and the POS value of the region is represented as a prize value. The notation employed by Butt and Cavalier [13] for the MTMCP is adapted, along with the incorporation of the highest priority regions as a set P. Set P contains all triangular regions which are required to be searched, and is analogous to the edge search formulation.

Butt and Cavalier consider the single objective of maximizing prize return given a fixed time constraint. This objective is adapted to the SAR problem by considering a biobjective of maximizing POS (analogous to maximizing prize return) and minimizing a penalty function incorporating a soft constraint, rather than a hard constraint, on the time length of search paths. The penalty function also seeks to attain work equity between all of the search resources and is similar to that proposed by Bodin and Levy [12].

The variables utilized in the formulation consist of the following:

POS_i = POS value associated with triangular region i

b_i = time required to search triangular region i

d_{ij} = time required to travel from region i to region j (access time)

P = set of triangular regions which must be searched — highest priority regions

T = maximum time limit on a resource's search time

t = minimum time limit on a resource's search time

U = upper penalty weight

L = lower penalty weight

s = number of search resources

n = number of triangular regions in the graph; 0 = search base

N = set of triangular regions, including the dummy search base

K = set of search resources

$$x_{ijk} = \begin{cases} 1 & \text{if region } j \text{ is searched immediately after searching region } i\text{, by resource } k \\ 0 & \text{otherwise} \end{cases}$$

As spatial triangles are effectively defined by spatial vertices in the transformation to the SAR problem, b_i is significantly larger than d_{ij}, unlike in the MTMCP. In addition d_{ij} is not explicitly represented. In the SAR problem d_{ij} represents the access time of moving from the location where a triangular region search is completed (initially assumed to be a vertex in the TIN) to the location where the searching is to begin for the next region in the resource's path (also initially assumed to be a vertex). Hence, the value of d_{ij} is represented by the shortest path between the two vertices.

Butt and Cavalier determine k distinct tours which begin and end at a depot vertex. We adapt the problem to determine k distinct paths which all begin at a base vertex (the search base) but can end at any vertex.

The problem itself is more complicated than the MTMCP in that b_i, the time taken to search triangular region i, must be defined by a specific search technique, POD level and orientation, *i.e.*, the entry and exit vertex. To fix b_i apriori the entry and exit vertices must be set. This can be done individually for each region by deciding upon a search technique and POD level, and then determining the least cost orientation for this to be executed in. In such a deterministic approach there is no room for adjusting the search orientation of a region with respect to the access costs d_{ij}, of moving between the regions which constitute its neighbours in the search path.

The graph which the problem is formulated over consists of n triangular regions, the set of all triangular regions being represented by N. To counteract the problems of representing the search base (we assume that the search base is located at a vertex on the TIN) by a spatial triangle, as the triangular regions are represented, we construct a "dummy vertex", with no spatial qualities, to represent the search base. This vertex is linked, by edges, to each triangular region which has the actual search base vertex as one of its three vertices. We represent this set of triangular regions by the set S. The search base is defined as vertex 0. The edge costs $d_{0,j} = 0\ \forall j \in S$ if the vertex m where the search of region j is to begin from is the base vertex 0, otherwise $d_{0,j}$ is equal to the cost associated with the region edge (search base, m).

To ensure that all triangular regions can initially be reached from the base vertex we include additional edges from the 'dummy' base vertex to all regions which do not have the search base vertex as one of their vertices. The costs assigned to these edges are determined from the shortest path distance between the search base vertex and the starting vertex, m, of each region. Additionally we define $POS_0 = 0$ and $b_0 = 0$.

The objective which seeks to maximize POS is assigned the weight λ_{pos} in the objective function and is defined as

$$z_{pos} = \sum_{k=1}^{s} \sum_{i=0}^{n} \sum_{j=1}^{n} POS_i x_{ijk}$$

The total length of a resource's path is formulated as

$$T(k) = \sum_{i=0}^{n} \sum_{j=1}^{n} d_{ij} x_{ijk} + \sum_{i=0}^{n} \sum_{j=1}^{n} b_j x_{ijk}$$

This path length measure is then incorporated into the penalty function on resource work equity and path length. The penalty function is assigned a weight of λ_{eq} in the objective function and is defined as

$$z_{eq} = U \sum_{k=1}^{s} \max\{T(k) - T, 0\} + L \sum_{k=1}^{s} \max\{t - T(k), 0\}$$

Hence the objective of the problem can be formulated as

$$\text{maximize } \lambda_{pos} z_{pos} - \lambda_{eq} z_{eq}$$

The static routing formulation for the problem can then be expressed as follows:

$$\text{maximize } \lambda_{pos} z_{pos} - \lambda_{eq} z_{eq} \quad (1)$$

s.t.

$$\begin{array}{lll} \sum_{k=1}^{s} \sum_{j=1}^{n} x_{ijk} \leq 1 & \forall i \in N & (2) \\ \sum_{j=1}^{n} x_{0jk} = 1 & \forall k \in K & (3) \\ \sum_{k=1}^{s} \sum_{j=1}^{n} x_{ijk} = 1 & \forall i \in P & (4) \\ x_{ijk} \in (0,1) & \forall i,j,k & (5) \end{array}$$

path connectivity constraints

subtour elimination constraints

where

(1) - objective function consisting of:

- max total POS

- min penalty function based upon the cost of each resource's path with respect to limits on search time

(2) - all triangular regions are searched by at most one resource

(3) - all resource's search paths must include the search base

(4) - all regions within the highest priority set are searched

(5) - integrality constraints

A more flexible approach to determining the search orientation of a triangular region and hence determining the b_i values would be to incorporate the entry and exit vertices of a region as variables in the problem formulation. This would allow the b_i value to be selected which resulted in the least time search path once access times had been incorporated. For example, the following variables could be defined to replace b_i and d_{ij} respectively.

b_{imn} = the time required to search triangular region i entering at vertex m and exiting by vertex n

d_{i_n,j_m} = the time of the shortest access path between the exit vertex n of region i and the entry vertex m of region j

Entry and exit vertices m and n would be constrained to those vertices describing the triangular region.

The formulation could also be varied such that resources begin from differing starting points, not just the search base, as for the edge formulation variants. A hard time constraint could also be included, as for the MTMCP, instead of the utilization of a penalty function. If re-searching of regions is permitted then constraint (2) can be relaxed.

6.10 Unique Aspects of the Problem

In summary, the features of the SAR problem which uniquely identify it from any one particular problem in the coverage and routing literature are:

- the inherent dynamism;
- windy traversal costs altering with environmental conditions;
- spatial traversal of the triangular regions;
- multiple paths being desired;

- multiple, conflicting objectives; and
- combination edge and triangular region traversal.

As the problem contains a scheduling, apportioning and routing component in a dynamically changing environment, it is not a problem which can be easily solved by any one particular method, and certainly not exactly, within the time constraints defined by search urgency. In light of this, heuristic methods are the most appropriate solution approaches to utilize over defined sub-problems of the SAR problem. In particular, the scheduling components of the problem can be incorporated by first identifying those regions which should be searched first and which can, approximately, be searched within the time frame available. These can be clustered into a primary search area as one approach. Additionally, the problem is naturally decomposable into planning time periods where operations are currently already divided into work shifts and into specific search phases. Solving the SAR problem over a single period provides for a more manageable solution approach which is still responsive to dynamism.

In the following chapters heuristic solution approaches to the SAR problem will be outlined along these lines. The first methods developed are for the sub-problem of a reconnaissance search phase which concentrates solely on searching the edge regions of the search region graph, and is constrained to the duration of the first planning period. Methods are then presented for the general search phases, Phase Two and Three, followed by specific dynamic search strategies.

CHAPTER 7

The Reconnaissance Search: Edge Routing

7.1 Preamble

The reconnaissance or 'hasty search' activated in the initial phase of the search operation is modelled over the search region graph by edge-only traversal. It is assumed that information is available which will enable POA values to be initially allocated to each element of the TIN.[1] This information may be in the form of information specific to the subject, such as the recorded intentions of a path that they were to follow or a point that they were last seen at, or it may be more general, in the form of historical search data relevant to the search area. POA values may also be derived from generalizations based on the characteristics of the subject or from data resulting from the trail-based POA method. Given that each edge has a POA value initially assigned to it, a POD value of 1.0 is then allocated to each edge search to give a POS indicator for each edge that is equal to its POA value. It is these POS indicators which are utilized in planning the paths of the search resources during this search phase.

It is assumed that the search paths, once allocated, remain unchanged (static) throughout this search phase unless the subject is located. The justification of this assumption is that the primary task of this search phase is to gather as much information on the subject's movements as possible, with the resources and time frame available. Once any additional information is obtained, POA values are updated and are used to plan future search phases. Dynamically responding to the first clues found during this stage of the

[1] An element is the smallest definable entity, which in this instance is either a triangular region or an edge of a triangular region.

search, may result in resources being inefficiently reassigned to areas due to misinformation, or assigned to follow clues which are subsumed by information gathered later in the original assignments. Unless the urgency of the search is high, or information is discovered which leads directly to the subject, it is assumed that a static reconnaissance search will not endanger the life of the subject if promising leads are not followed until the second phase of search. In high urgency situations a dynamic response may be incorporated or additional resources can be kept on standby to be routed to any high priority areas which may arise.

The reconnaissance search is of $PERIOD$ hours in length. This is set as a parameter of the simulation and will generally be equal to half a shift or six hours. Each search resource is allocated a search path not exceeding $path\_limit$ where $path\_limit \leq PERIOD$ and is defined with consideration to weather and light conditions, and whether night searching will be utilized.

The net benefit of the reconnaissance search is measured by the increase in POS_{cum} and ROC_{cum} achieved over the period, and the search strategy is driven by these measures.

7.2 Modelling Current Search Practices

We begin by modelling specific reconnaissance tasks in a manner which follows their current implementation in the field as outlined in Chapter 3. In particular heuristics are designed which activate:

1. the search of a relevant section of the intended path of the subject;
2. the search of edges incident to the PLS;
3. a perimeter cut;
4. the search of hazards *eg.,* rivers, ridge lines and bluffs;
5. the search of huts.

Each of these actions is performed according to the given ordering and in consideration to the availability and type of search resources on hand. However, if search urgency is high, searches of hazards are conducted first.

It is assumed in the initial simulation model that all resources begin the reconnaissance phase of the search at the search base and that transportation to search regions (other than by foot) is unavailable. The primary search area, G_P, upon which these tasks are executed is a subset of the search region graph, G, and is defined by current POA values. $G_P = (V_P, E_P) \subseteq G = (V, E)$, where V and E are the set of vertices and edges comprising the search region graph, and V_P and E_P are the set of vertices and edges describing the primary search area.

These search tasks are now outlined in detail with the variables used within the algorithm functions being referenced in the glossary.

7.2.1 Searching the Intended Path of the Subject

When the subject's intended path is known at the outset of planning the SAR operation, a search resource can be assigned to search that segment of the intended path which is considered to be the most relevant. Here relevancy can be defined as the portion of the path that the subject is most likely to be currently in the vicinity of. The intended path is initially viewed as a connected set of edges of the TIN (which could be extended to a path that enters triangular regions).

If additional information is available as to the location of the PLS[2] then it is assumed that the relevant section of the intended path begins from this point and further, it is assumed that in conducting this type of search the subject has not backtracked over ground initially covered; this will be a valid assumption in most search instances resulting from either accident or hazardous weather conditions. The exception may occur in those instances where error in navigation is the factor triggering the search, or where medical or weather factors impede navigation. While acknowledging such exceptions this initial search phase is planned on the most likely scenarios.

When no information on the PLS is known, the time difference between when the subject's intentions were recorded and when the search operation began, is computed to estimate the length of the intended path which the subject could have covered. (Alternatively the portion of the path which has the highest POA ratings over all of the path segments can be selected for searching.) The search resource is then assigned to search this portion of the intended path from its origin. The search is initiated from the origin vertex due to no additional information on the subject's movements being available. The resource is allocated a path to the origin vertex from the search base via the shortest edge path, and the intended path is traversed in the same direction as that which the subject intended to follow.

When the PLS is known, the section of the intended path which is searched is that portion which it is estimated that the subject could have covered since the time of the sighting at the PLS. The resource assigned to search the intended path is initially routed over those edges which comprise the shortest path from the search base to the PLS. If

[2] We assume that the PLS is a vertex, both of the TIN and of the intended path.

the time to cover this path exceeds *path_limit* (and the base is positioned closer to the PLS vertex than other vertices on the most relevant section of the intended path) then this reconnaissance task is unachievable under the current assumptions and constraints. In such an instance the initialization of a sub-base closer to the subject's intended path would be warranted.

If the length of the intended path considered critical for searching exceeds the available search time of a single search resource, additional resources can be allocated to search the remaining edges. The heuristic method actions this by routing an additional resource to the end vertex of the previous resource's path, via the shortest path from the search base.

The heuristic method generating this form of search assignment is detailed in Algorithm 7.1 for the first resource to be allocated this type of search task.

7.2.2 Searching Edges Incident to the PLS

When information of a PLS is known it is common practice to allocate resources to search regions adjacent to this position. We assume that there are *tnum* search resources available to assign to this specific search activity. Each of these resources is assigned one at a time until either the regions requiring searching are all allocated, or the available resources are depleted. The searching of the region adjacent to the PLS is restricted to those edges incident to the PLS vertex.

The search resources are initially routed via the shortest available edge path from the search base to the PLS vertex. Resources are then assigned to search the edges adjacent to the PLS which carry the maximum POS values of all candidate edges. Candidate edges are those which are as yet unallocated to any search resource and which when searched by the resource do not violate the *path_limit* time constraint.

The remainder of each resource's search path is constructed by selecting the adjacent, unallocated edge incident to the last vertex in the resource's path, which maximizes POS and which can be searched within *path_limit*. Hence, search paths are constructed which emanate from the PLS, myopically searching edges which return the greatest increase in POS. If no such edge exists and there remain as yet unassigned edges incident to the PLS which can be searched from the resource's current position, any such edge is allocated to the resource. If no such edge exists then the resource is assigned to perform a hazard search if this search does not exceed *path_limit* (detailed in Section 7.2.3.1).

Algorithm 7.2 describes this allocation heuristic.

Algorithm 7.1 function *follow_path*(PLS)

```
    // find relevant section of intended path to follow
    if PLS is known then
        pstn ← position of PLS on intended path
        time ← D_{base,PLS}
        if (time < path_limit) then
            path ← shortest path from base to PLS
            maxtime ← search commencement - time subject sighted at PLS
        end
        else
        // base is too distant from PLS to effect search
    end
    else
        pstn ← 0
        time ← D_{base,path_{I,0}}
        if (time < path_limit) then
            path ← shortest path from base to path_{I,0}
            maxtime ← search commencement - time intentions recorded
        end
        else
        // start vertex of intended path is too distant from PLS to effect search
    end
    t ← 0.0
    stop ← −1
    limit ← −1
    for j = pstn to num_I and stop ≠ 1 do
        if (t < maxtime and limit ≠ 1) then
            t ← t + cost_{path_{I,j},path_{I,j+1}}
            if (time + t < path_limit) then
                add vertex path_{I,j+1} to path
            end
            else
            limit ← 1
        end
        else
        stop ← 1
    end
    return(path)

end
```

Algorithm 7.2 function *incident_PLS*($PLS, tnum$)

```
for k = 0 to tnum do
    time_k ← 0.0
    num_k ← 0
    limit_k ← 1
end
k ← 0
end ← −1
if (D_{base,PLS} > path_limit) then
    // not possible to route resources to PLS from current base in time available
    end ← 1
end
while (k < tnum and end ≠ 1) do
    increment path of resource k by shortest path from base to PLS
    time_k ← D_{base,PLS}
    while (limit_k ≠ 1) do
        select unsearched edge (path_{k,num_k}, m) which maximizes POS
         and where time_k + c_{path_{k,num_k},m} ≤ path_limit
        if candidate edge found then
            num_k ← num_k + 1
            path_{k,num_k} ← m
            update time_k
            if (time_k = path_limit) then  limit_k ← 1
        end
        else if (there exists an unsearched edge (PLS, m) and
        path_{k,num_k} ≠ PLS) then
            if (time_k + D_{path_{k,num_k},PLS} + c_{PLS,m} ≤ path_limit) then
                update time_k
                increment path of resource k by shortest path from
                path_{k,num_k} to PLS
                num_k ← num_k + 1
                path_{k,num_k} ← PLS
            end
            else if (time_k + D_{path_{k,num_k},i} + c_{m,PLS} ≤ path_limit) then
                update time_k
                increment path of resource k by shortest path from
                path_{k,num_k} to m
                num_k ← num_k + 1
                path_{k,num_k} ← PLS
            end
            if (time_k = path_limit) then  limit_k ← 1
        end
        else
            // attempt to allocate resource k unassigned hazards to search
            hazard_route(k)
        end
    end
    if still unassigned edges adjacent to PLS then
    k ← k + 1
    else
    end ← 1
end

end
```

7.2.3 Alternative PLS Search

The Marin County SAR Team [114] notes that "working into the search area from outside towards the PLS" acts as a useful confinement technique in the reconnaissance search phase. In the TIN representation this translates to a search path which traverses edges that lead towards the PLS vertex and which searches the edges adjacent to the PLS. We define the set L to contain the vertices which are connected to the PLS vertex by an edge in the search region graph. Search resources are initially allocated to search the shortest path from the base to a vertex in L, and to then search the edge from this vertex to the PLS. If there exists remaining search time for that resource they are then allocated to search another, as yet unsearched, edge incident to the PLS vertex. This heuristic method is described in Algorithm 7.3.

7.2.3.1 Hazard Searching En Route

If a resource has been assigned a partial search path which cannot be completed with the edges belonging to that specified task (*eg.*, unassigned edges incident to the PLS), without violating *path_limit*, the search path may be completed by allocating the resource to search identified hazards in the primary search area. Such hazards may include rivers, bluffs and ridges in deteriorating weather conditions. Hazards comprising edge segments are represented by two hazard markers indicating their beginning and end. Hazards which can be represented on the TIN by a single vertex *eg.*, a waterfall, have two identical hazard markers equal to that vertex. Hazards are allocated an individual search time $hsearch_{ij}$, where $i =$ the starting hazard marker and j the hazard label.

When a resource is allocated an unassigned hazard to search in order to complete their search path, those hazards which can be searched within the remaining search hours are ranked in increasing order of their shortest path distance from the resource's current position. The shortest path is taken as that to the closest of the two hazard markers. The resource is then allocated a path to the first hazard in this ranked list with this procedure being repeated until no unallocated hazards remain which can be searched by the resource within the hours available. This approach assumes that all hazards are equally hazardous. Alternatively, hazards can be weighted by criticality as well as by access time.

An allocation approach which considers the biobjective of maximizing POS and minimizing the time required to reach a hazard could also be developed by temporarily reducing the time needed to search edges with greater POS, so as to increase the likelihood of such edges being components of the shortest path to the hazard. The resulting actual time of the shortest path would still be restricted by the *path_limit* constraint.

Algorithm 7.3 function *towards_PLS*($PLS, tnum$)

```
for k = 0 to tnum do
    time_k ← 0.0
    num_k ← 0
    limit_k ← 1
end
k ← 0
end ← −1
while (k < tnum and end ≠ 1) do
    if (D_{base,v} > path_limit ∀v ∈ L) then
        // not possible to route resources from current base in time available
        end ← 1
    end
    else
        while (resources remaining and edges incident to PLS not yet searched) do
            v ← vertex ∈ L which is closest to base
            increment path of resource k by shortest path from base to v
            add edge (v, PLS) to path of resource k
            time_k ← D_{base,v} + c_{v,PLS}
                if resource k not at path limit then
                    select the unsearched edge (PLS, v) which maximizes POS
                     and can be searched within path_limit
                    add edge(PLS, v) to path of resource k
                    time_k ← time_k + c_{PLS,v}
                end
                else
                limit_k ← 1
            end
            k ← k + 1
        end
    end
end

end
```

Algorithm 7.4 function *hazard_route*(*resource*)

```
limit ← −1
start ← path_{resource,num_resource}
while (limit ≠ 1 and unassigned hazards remain) do
    for i = 0 to hazards do
        if hazard not yet assigned then
            pstn_i ← closest hazard marker to start
            r_i ← D_{start,pstn_i}
        end
        order hazards in increasing order of r_i
        i ← hazard with lowest r_i rating
        h1 ← pstn_i
        h2 ← second marker of hazard i
        if (D_{start,h1} + time_resource + hsearch_{h1 i} ≤ path_limit) then
            spath ← shortest path from start to h1
            increment path of resource by spath
            increment num_resource
            time ← time + D_{start,h1} + hsearch_{h1 i}
            if (h1 ≠ h2) then
                path_{resource,num_resource} ← h2
                num_resource ← num_resource + 1
                start ← h2
            end
            else
            start ← h1
        end
        else
        limit ← −1
    end
end

end
```

The single objective method is detailed in Algorithm 7.4.

7.2.4 Searching of Hazards

The combined co-ordination of sending resources from the search base to search specific hazards identified in the primary search area, is expressed in Algorithm 7.5. The algorithm is essentially a parallel version of *hazard_route()* where resources are assigned to search the unassigned hazard closest to the last vertex on their path, which does not violate *path_limit*. All resources are initially allocated a path from the search base to the closest of the two hazard markers of the hazard assigned. A resource's search path is considered to be complete when no unallocated hazards can be searched by that resource within the available search time.

Algorithm 7.5 function *hazard_parallel*($tnum$)

```
hct ← 0
for k = 1 to s do
    num_k ← 0
    time_k ← 0
    limit_k ← −1
end
for i = 0 to hazards do
    pstn_i ← closest hazard marker to base
    r_i ← D_{base,pstn_i}
end
order hazards in increasing order of r_i
for k = 0 to tnum do
    // initially allocate a hazard to be searched by each resource
    i ← unallocated hazard with lowest r_i rating
    h1 ← pstn_i
    h2 ← second hazard marker of hazard i
    if (D_{base,h1} + hsearch_{h1,i} ≤ path_limit) then
        time_k ← time_k + D_{base,h1} + hsearch_{h1,i}
        spath ← shortest path from base to h_1
        increment path of resource k by spath
        increment num_k
        if (h1 ≠ h2) then
            path_{k,num_k} ← h2
            num_k ← num_k + 1
        end
    end
end
for k = 0 to tnum do
    if (time_k = path_limit) then
    limit_k ← 1
end
while (limit_k ≠ 1 for some k and hct < hazards) do
    i ← resource with greatest number of allocated search hours and not at limit
    t ← path_limit − time_i
    find unallocated hazards at a time distance less than t from path_{i,num_i−1}
    if (such a hazard exists) then
    select the closest of these and add this hazard to resource i's path via the
    shortest path connection
    else
    limit_i ← 1
end

end
```

Due to the fact that all resources begin their search path from the search base it is possible that some identified hazards are simply out-of-range, in terms of available search time. This is a potential problem for all the reconnaissance phase heuristics when using foot access and can be overcome only by starting resources from differing and dispersed locations *e.g.*, *sub-bases*, or by utilizing other means of task transfer such as a helicopter or 4-wheel drive vehicle.

7.2.5 Perimeter Cut

The aim of a perimeter cut of the primary search area is to establish whether or not the subject has moved beyond the perimeter of that area, *i.e.*, to detect any tracks (or other clues) leaving the region which may pertain to the subject. Current practice indicates the use of a method which follows the perimeter and a method which cuts at right angles to the intended route of travel, if this known. Essentially, these construct an intelligent form of binary search that seeks to establish which portion of the greater search region the subject is in, with minimum search effort.

The cut around the search perimeter is a continuous cut, if such a cut can be achieved in the search hours available. The searchers concentrate on searching the ground immediately at their feet.

Several forms of perimeter cut searching on the TIN are developed depending on whether information on the subject's intended path of travel is known or not.

7.2.5.1 Cutting the Intended Path

When the subject has left intentions of their path of travel (assumed initially to be a contiguous subset of edges of the TIN) we develop a heuristic to model the path of a search resource cutting at right angles to the intended route. As the intended route is represented by a series of edges the search is modelled by generating a search path which traverses edges that intersect the intended path through each of its vertices.

We define the intended path of the subject, of length n, say, by the vertices it moves through:

$$(path_{I,1}, path_{I,2}, path_{I,3}, \ldots, path_{I,n})$$

There will be two triangular regions adjacent to any vertex $path_{I,j}$ which have vertex $path_{I,j+1}$ in common. This is due to the construction of the TIN and the connectivity of the intended path. The only exception is where edge $(path_{I,j}, path_{I,j+1})$ lies on the convex hull of the TIN. Of the two edges incident upon vertex $path_{I,j}$ which also comprise an edge of these two triangular regions, one will be directed towards a vertex positioned closer to the perimeter of the search area than the other. We label this edge $(path_{I,j}, q)$ as illustrated in Figure 7.1.

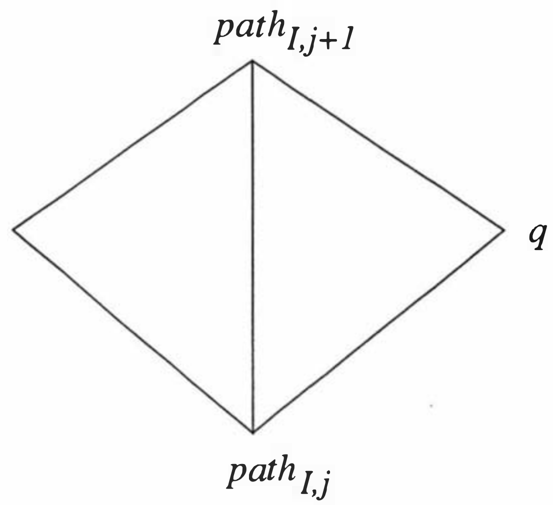

Figure 7.1: Edges adjacent to the intended path of the subject.

If edge $(path_{I,j}, path_{I,j+1})$ occurs on the convex hull of the TIN then edge $(path_{I,j}, q)$ is selected from the only triangular region which has both $path_{I,j}$ and $path_{I,j+1}$ as its vertices. Of the two edges which could be taken to move a search resource from $path_{I,j}$ to $path_{I,j+1}$ so as to cross the intended path at vertex $path_{I,j+1}$, the edge $(path_{I,j}, q)$, followed by edge $(q, path_{I,j+1})$, are selected. By selecting the edge $(path_{I,j}, q)$ the resource moves towards the perimeter to encompass some of the objective of the overall perimeter search, and may in fact hit the perimeter, depending on its position. Alternatively an edge could be selected based on cost or POS considerations.

The resulting search path is then composed of alternating 'moves'; a move towards the perimeter followed by a move towards the intended path. This path is illustrated in Figure 7.2.

The initial vertex of the intended path, $path_{I,1}$, is reached from the search base via the shortest edge path connection and the first resource is allocated to search the initial portion of the intended path until *path_limit* is reached. If the terminal vertex of the intended path, $path_{I,n}$, has not been reached, additional search resources are allocated to intersect the remaining portion of the intended path. Additional resources are initially allocated the shortest path from the search base to the next vertex on the path from that covered by the previous search resource. A search path of this type could be constructed from the PLS or some other critical point along the intended path, rather than from the initial vertex, and could be terminated prior to vertex $path_{I,n}$ if the likelihood of the

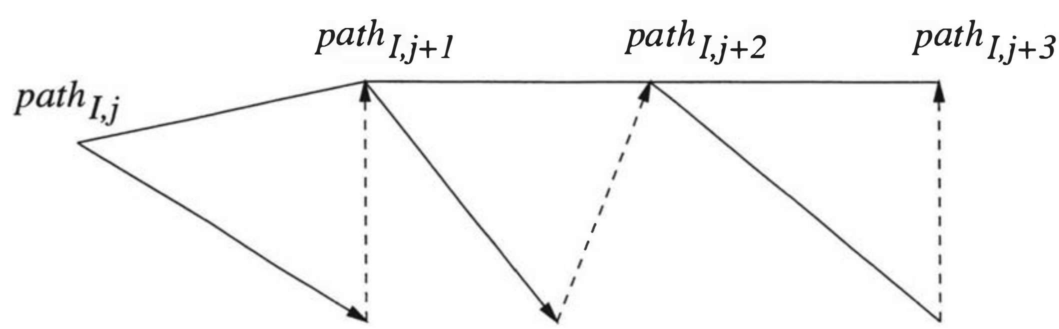

Figure 7.2: Search path of alternating moves.

subject reaching this vertex was low.

If the resource completing the search at $path_{I,n}$ has remaining search time available it is then allocated to the closest vertex on the perimeter and continues with a perimeter edge search from this vertex, selecting the initial edge as the one with maximum POS. In instances where two edges have the same POS value, the edge of minimum cost is selected.

Algorithm 7.6 describes the generation of an intersecting search of the intended path, beginning at vertex $path_{I,1}$ and terminating at vertex $path_{I,n}$.

7.2.5.2 Perimeter Cut

A full perimeter cut of the convex hull of the primary search area can also be actioned when the intended path of the subject is known. The initial starting point of the perimeter cut begins at the perimeter vertex closest to a vertex on the intended path. Hence, if a limited number of resources are available and all the edges comprising the perimeter cannot be searched in the reconnaissance phase, those edges of closest proximity to the intended route are searched. (Closest, in this context, is defined in terms of time, not distance.) If only the PLS is known, the perimeter vertex closest to this vertex is selected as the starting point of the perimeter search. If both the intended path and the PLS are unknown, then the starting point of the perimeter search is the vertex on the convex hull of the primary search area which is closest to the search base.

The first resource assigned to searching the perimeter is allocated a path from the search base to this starting vertex, which we label h_0, and then follows a connected path of perimeter edges, beginning with the edge having greatest POS. The search path terminates when the perimeter edges have all been searched or *path_limit* is reached. If perimeter edges remain unsearched and *tnum* search resources are available to assign to this search task, search paths are sequentially assigned to resources until their available

Algorithm 7.6 function *intersecting_path*(*tnum*)

```
for k = 0 to tnum do
    time_k ← 0
    limit_k ← − 1
end
k ← 0
j ← 0
while (k < tnum and j < num_I) do
    while (D_{base,path_{I,j}} > path_limit) do
        // vertex out of range
        j ← j + 1
    end
    if (j < num_I) then
    increment path of resource k by shortest path between base and path_{I,j}
    while (limit_k ≠ 1 and j < num_I) do
        find two triangular regions having (path_{I,j}, path_{I,j+1}) as an edge
        label these two regions tri1 and tri2
        // if the edge exists on the hull, only one such triangular region exists
        x ← vertex of tri1 ≠ path_{I,j} or path_{I,j+1}
        y ← vertex of tri2 ≠ path_{I,j} or path_{I,j+1}
        next ← closest vertex of x and y to the perimeter
        if (c_{path_{I,j},next} + time_k ≤ path_limit) then
        add edge (path_{I,j}, next) to path of resource k and update time_k
        else
            next ← other vertex of x,y pair
            if (c_{path_{I,j},next} + time_k ≤ path_limit) then
            add edge (path_{I,j}, next) to path of resource k and update time_k
            else
            limit_k ← 1
        end
        if (limit_k ≠ 1) then
            if (c_{next,path_{I,j+1}} + time_k ≤ path_limit) then
            add edge (next, path_{I,j+1}) to path of resource k and update time_k
            j ← j + 1
        end
    end
    k ← k + 1
end
while (limit_k ≠ 1 and k < tnum) do
// execute a perimeter search with the remaining resources

end
```

Algorithm 7.7 function *perimeter(tnum)*

```
for k = 0 to tnum do
initialize resource variables
k ← 0
end ← −1
if PLS is known then
h_0 ← perimeter vertex closest to PLS
else if (intended path known) then
h_0 ← perimeter vertex closest to a vertex on the intended path
else
h_0 ← perimeter vertex closest to base
if (D_{base,h_0} ≤ path_limit) then
    time_k ← D_{base,h_0}
    increment path of resource k by shortest path between base and h_0
    j ← 0
    h_{j+1} ← such that edge (h_j, h_{j+1}) has the greatest POS and h_{j+1}
    is on the perimeter
end
else
    // no perimeter search possible from base
    end ← 1
end
while (end ≠ 1) do
    while (limit_k ≠ 1) do
        if (time_k + c_{h_j,j_{j+1}} ≤ path_limit) then
            add edge (h_j, h_{j+1}) to path of resource k and update time_k
            j ← j + 1
            h_{j+1} ← next vertex on perimeter such that h_{j+1} ≠ h_{j-1}
        end
        else
        limit_k ← 1
        if no perimeter edges remain to be searched or available resources are
        depleted then end ← 1
    end
    if (end ≠ 1) then
    // perimeter edges remain to be searched
        k ← k + 1
        if PLS or intended path known and k = 2 then
            h_j ← h_0
            h_{j+1} ← vertex of perimeter edge (h_j, h_{j+1}) not previously searched
        end
        else if (D_{base,h_j} + c_{h_j,h_{j+1}} > path_limit) then
            if edge adjacent to h_0 is unsearched then
                h_j ← h_0
                h_{j+1} ← vertex of perimeter edge(h_j, h_{j+1}) not previously searched
            end
            else
            end ← 1
        end
        increment path of resource k by shortest path between base and h_j
    end
end
for resources allocated search paths do
alter path direction if better POS over time is achieved

end
```

search hours are exhausted, at which point the next available resource is assigned a search path from the base to a vertex on the perimeter.

Where the intended path or PLS is known, the second search resource also begins a perimeter search from vertex h_0 but traverses the unsearched perimeter edges in the opposing direction, *i.e.*, one resource traverses the perimeter edges clockwise, the other anti-clockwise, as depicted in Figure 7.3.

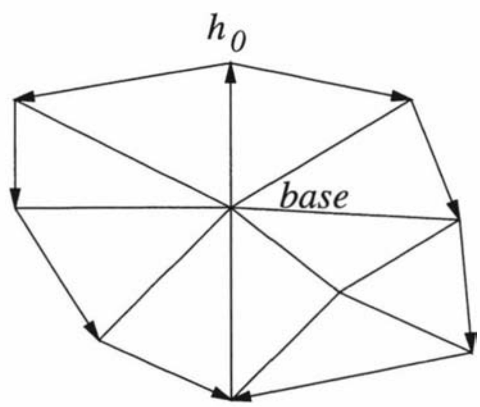

Figure 7.3: A perimeter search by two search resources when subject intentions or PLS are known.

When no directional or locational knowledge is available then a continuous perimeter search is executed with additional resources searching the perimeter from the vertex at which the previous resource terminated their search. This is illustrated in Figure 7.4 for a search base located interior to the primary search area. If it is not possible for the next search resource to move from the search base to this vertex within *path_limit*, the resource is instead routed to the initial starting vertex of the perimeter cut and searching is commenced in the opposite direction to that traversed by the first resource. Search allocation continues until no further perimeter edges can be searched in the available search time or all perimeter edges have been assigned.

Algorithm 7.7 describes the heuristic method of allocating *tnum* resources to search the edges comprising the perimeter of the primary search area.

The assumption is made that a search resource completes the search of an entire edge and that searching does not terminate at the point along an edge at which the resource's search hours are exhausted. This is due to the inherent difficulties faced by a second resource in navigating to this exact point in order to commence searching without omitting or re-searching any portion of that edge.

It may be possible to achieve greater POS return over time or greater time efficiencies by reversing the direction of traversal of a given resource's path. For example, if the path

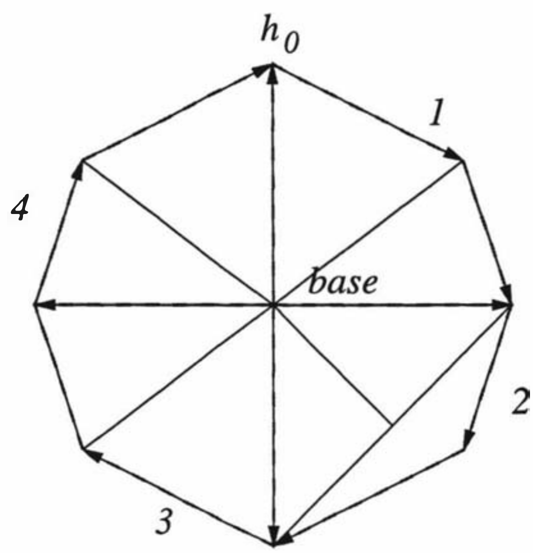

Figure 7.4: A perimeter search by four search resources, from an interior search base, when no intended path or PLS information is known.

of a resource is represented as:

$$base, \ldots, h_0, h_1, \ldots, h_{j-1}, h_j$$

where $base, ..., h_0$ is the shortest path from the base to the starting perimeter vertex h_0, then the reversed path would be $base, \ldots, h_j, h_{j-1}, \ldots, h_1, h_0$, (where the resource is initially routed via the shortest path from the search base to perimeter vertex h_j.).

Improvements to the search paths may also be obtained by allowing a resource k to slightly exceed *path_limit* in order to search the next edge of the hull, (h_j, h_{j+1}) which would otherwise be out of range. This could be monitored by a hard penalty *eg.*, $time_k + c_{h_j,h_{j+1}} < U$, where U is an upper limit, or by a soft penalty approach where a penalty value could be applied for each unit of search time exceeding *path_limit*. Additionally, if the last resource searches only a small portion of the perimeter it may prove more favourable, in terms of resource utilization, for the previous resource to exceed *path_limit* and search that portion instead.

7.2.6 Search of Huts

Huts in the primary search area are searched by the ground resources allocated to follow the intended path of the subject, or alternatively, if helicopter transport is available, a TSP tour of the huts can be conducted by a helicopter. The tour can be constructed by one of the many TSP heuristics available. The route taken by the helicopter can be extended to cover less dense areas that may or may not have been already allocated to some other search resource for searching.

7.3 Additional Search Heuristics

In addition to the reconnaissance search methods currently utilized we now consider a new set of search planning heuristics to extend the tool kit available to search managers. These tasks incorporate some of the wider concepts of edge routing approaches detailed in the literature.

7.3.1 Median Search Path

The motivation behind a median search path heuristic is to allocate a search resource to search a series of edges which are 'central' to the primary search area. Locating central paths through graphs is addressed by Slater [149] who works upon an undirected graph, and focuses primarily on tree networks. He extends results utilized in location theory to determine central points in a graph. We extend Slater's notation and concept of the eccentricity measure,[3] $e(P;G)$, of a path, P, through a connected graph, G, to the search region graph. The notation is adapted to cope with positively weighted (windy) edges, a fixed initial point and a maximum length on the path. This adapted measure of eccentricity is defined as:

$$e(K;G_P) = \max_v\{d(v,K) : v \in V_P)\}$$

where G_P is the primary search area graph, V_P the set of vertices of G_P, and $d(v,K)$ represents the shortest path distance in G_P from vertex v to a vertex on the path K. A path K is a *path center* of G_P if $e(K;G_P) \leq e(K';G_P)$ for any path K' in G_P.

A *path median* of G_P is defined by Slater as that path having the minimal distance measure $d(K;G_P)$ over all paths, where $d(K;G_P)$ is a minisum measure defined as:

$$d(K;G_P) = \sum_v\{d(v,K) : v \in V_P\}$$

7.3.2 Hedging Search Path

We define a *hedging search path* in G_P as the path which is the 'most distant' from other search paths in that search phase but whose edges still fall within the primary search area. A hedging path may be implemented during the reconnaissance phase as a path which covers regions 'most different' in nature (in terms of location or search scenario) from

[3]Eccentricity is a minimax measure of centrality.

those already allocated, in an attempt to cover scenarios considered less probable under the current circumstances. An additional purpose of such a path, in all search phases, is to minimize the response time to areas in the primary search area not covered by other resources. Such a path effectively 'hedges' the search scenario and related strategy being followed by search management in their task allocations.

Both static and time-dependent variants of hedging search paths are considered. The static variant seeks to determine a path with the greatest value of the following hedging measure based upon the measure of eccentricity:

$$h_1(K; G_P) = \max_v \{T(K, v) : v \in V_S\}$$

where V_S is defined as the vertices of currently assigned search paths. We use $T(K, v)$ to measure the shortest time required to reach a vertex v on an existing path from the hedge path, K, as G_P is not necessarily a symmetric graph.

A second hedging measure can be defined based on the measure of distance as:

$$h_2(K; G_P) = \sum_v \{T(K, v) : v \in V_S\}$$

A time-dependent hedging search path finds the path which is farthest from any other resource's position at a given time t. We simplify this somewhat to define a resource's position at t to be the vertex on their search path that the resource is closest to. The hedging measure is then redefined as:

$$h_1(K(t); G_P) = \max_v \{T(K(t), v) : v \in V_S(t)\}$$

where $K(t)$ represents the vertex on hedging path K at time t and $V_S(t)$ the vertices at which other active search resources are positioned closest to at time t.

Similarly the second hedging measure h_2 can also be redefined to give the following time-dependent measure:

$$h_2(K(t); G_P) = \sum_v \{T(K(t), v) : v \in V_S(t)\}$$

A myopic heuristic is now defined which constructs a hedging search path that maximizes a given hedging measure at the inclusion of the next edge in the path. The heuristic ensures that the resulting path is less than *path_limit* in length and initially seeks to select unallocated edges as candidates for inclusion, so as to increase the POS returned. This method of constructing a hedging search path is detailed in Algorithm 7.8 for a resource k.

The four different selection criteria (hedging measure) for *obj* are:

Algorithm 7.8 function *hedging_search_path*(k, obj)

```
t ← search commencement
time_k ← 0.0
limit_k ← −1
path_{k,num_k} ← base
num_k ← 1
while (limit_k ≠ 1) do
    for all vertices v adjacent to path_{k,num_k} do
        ct ← 0
        if (c_{path_{k,num_k} v} + time_k ≤ path_limit and v ≠ path_{k,num_k −1}) then
            if edge(path_{k,num_k}, v) is not allocated to another resource then
                candidate_ct ← v
                ct ← ct + 1
            end
        end
        if (ct = 0) then
            // relax restriction on selecting an unallocated edge
            for all vertices v adjacent to path_{k,num_k} do
                if (c_{path_{k,num_k},v} + time_k ≤ path_limit and
                v ≠ path_{k,num_k −1}) then
                    candidate_ct ← v
                    ct ← ct + 1
                end
            end
        end
    end
    if (ct ≠ 0) then
        t ← search commencement + time_k
        // select appropriate obj algorithm to select best candidate edge
        if (obj = 1) then
        bestu ← minisum_time(candidate, ct, t, path_{k,num_k})
        else if (obj = 2) then
        bestu ← minimax_time(candidate, ct, t, path_{k,num_k})
        else if (obj = 3) then
        bestu ← minisum_static(candidate, ct, path_{k,num_k})
        else
        bestu ← minimax_static(candidate, ct, path_{k,num_k})
        path_{k,num_k} ← bestu
        time_k ← time_k + c_{path_{k,num_k −1}, path_{k,num_k}}
        num_k ← num_k + 1
    end
    else
        // no feasible edges adjacent to path_{k,num_k}
        limit ← −1
    end
end

end
```

Algorithm 7.9 function *minisum_time*(*candidate*, *ct*, *t*, *start*)

```
    best ← −1
    bestu ← −1
    for i = 0 to ct do
        sum ← 0.0
        u ← candidate_i
        for k = 0 to resources do
            v ← vertex closest to resource k at t
            sum ← sum + D_{u,v}
        end
        if (sum = best) then
            // break ties based on POS value of edge
            if (POS_{start,u} > POS_{start,bestu}) then
                bestu ← candidate_i
                best ← sum
            end
        end
        if (sum > best) then
            bestu ← candidate_i
            best ← sum
        end
    end
    return(bestu)
end
```

1. $h_2(K(t); G_P)$: minisum — time dependent;

2. $h_1(K(t); G_P)$: minimax — time dependent;

3. $h_2(K; G_P)$: minisum — static;

4. $h_1(K; G_P)$: minimax — static.

Essentially the minimax selection criteria obtains a 'center' path, while the minisum selection criteria obtains a 'median' path, through the primary search area in consideration to the paths of the other search resources. Specifically each criterion can be defined as in Algorithm 7.9, Algorithm 7.10, Algorithm 7.11 and Algorithm 7.12.

A biobjective could also be formed to construct a hedging search path which also maximizes the POS measure over all edges in the primary search area which remain unallocated after other resource assignments.

7.3.2.1 Search Phase Two and Three

During search Phases Two and Three, where the main search concentration is on region coverage, the hedging search path definition can be altered to represent the distance

Algorithm 7.10 function *minimax_time*(*candidate*, *ct*, *t*, *start*)

```
    best ← −1
    bestu ← −1
    for i = 0 to ct do
        u ← candidate_i
        min ← INF
        for k = 0 to resources do
            v ← vertex closest to resource k at t
            if (D_{u,v} < min) then
            min ← D_{u,v}
        end
        if (min = best) then
            // break ties based on POS value of edge
            if (POS_{start,u} > POS_{start,bestu}) then
                bestu ← candidate_i
                best ← min
            end
        end
        if (min > best) then
            bestu ← candidate_i
            best ← min
        end
    end
    return(bestu)
end
```

between a vertex in the hedging search path and the centre point c of each of the regions currently being searched by other search resources. Here the time-related eccentricity measure becomes:

$$h_1(K(t); G_P) = \max_c \{T(K(t), c) : c \in C(t)\}$$

and the time-related distance measure becomes:

$$h_2(K(t); G_P) = \sum_c \{T(K(t), c) : c \in C(t)\}$$

where $C(t)$ is the set of region centres at time t.

7.3.3 Selection of Shortest Path

When determining the shortest path between a search resource at position i and a search resource at position j on the TIN, a number of possible measures exist. We require a measure that is both realistic and able to account for the actual response time over the given terrain.

Algorithm 7.11 function *minisum_static*(*candidate*, *ct*, *start*)

```
    best ← −1
    bestu ← −1
    for i = 0 to ct do
        u ← candidate_i
        sum ← 0.0
        for k = 0 to resources do
            min ← INF
            for m = 0 to num_k do
                v ← path_{k,m}
                if (D_{u,v} < min) then
                min ← D_{u,v}
            end
            sum ← sum + min
        end
        if (sum = best) then
            // break ties based on POS value of edge
            if (POS_{start,u} > POS_{start,bestu}) then
                bestu ← candidate_i
                best ← sum
            end
        end
        if (sum > best) then
            bestu ← candidate_i
            best ← sum
        end
    end
    return(bestu)

end
```

Algorithm 7.12 function *minimax_static*(*candidate*, *ct*, *start*)

```
best ← −1
bestu ← −1
for i = 0 to ct do
    u ← candidate_i
    min ← INF
    for k = 0 to resources do
        for m = 0 to num_k do
            v ← path_{k,m}
            if (D_{u,v} < min) then
            min ← D_{u,v}
        end
    end
    if (min = best) then
        // break ties based on POS value of edge
        if (POS_{start,u} > POS_{start,bestu}) then
            bestu ← candidate_i
            best ← min
        end
    end
    if (min > best) then
        bestu ← candidate_i
        best ← min
    end
end
return(bestu)
end
```

Two methods for obtaining a (shortest) geodesic path between i and j are via planar unfolding or a region-traversal graph approximation [119, 162]. However, a more computationally efficient approach is to use the shortest edge paths between the vertices closest to i and j as an approximation to the real traversal of the resources. As a number of the edges model real features such as tracks and ridge lines, this approximation is not considered unrealistic, as these features will often generate the quickest means of moving from one point to another. We assume, through the use of GPS that the locations i and j can be determined (within error bounds) at any time, t.

As the TIN forms a sparse graph, the degree of each vertex limits the choices of search paths at any point. In particular, search paths will be influenced by the choice of the initial portion of a path. Shortest edge paths may also overlap due to the restriction on the number of edge connections between any pair of vertices. As the path of least resistance is required, an additional constraint that no edges of terrain four classification can be utilized may be employed.

7.3.4 Nearest Neighbour Approach

We adapt the nearest neighbour insertion method used to approximately solve the TSP, to create edge paths for each available search resource. The search paths are initialized from the starting vertex of each resource, $start_k$, which may differ over resources. The heuristic procedure simultaneously creates the search paths for all the resources in an attempt to equalize the number of search hours allocated to each for equity purposes.

The 'nearest' edge can be defined traditionally in cost-terms or can be redefined such that the 'cost' is equal to the increase in the objective function gained by adding the edge to the search path. The heuristic is generalized to accept a given partition of the search region graph within which to determine a path for each resource. This partition may be equal to the primary search area graph, G_P, or it may be a strategically-defined subgraph of it. Hence, as all the search regions within this subgraph have high priority the method, by selecting the least cost insertion, attempts to search these regions in the quickest time.

Those edges that emanate from a resource's starting vertex are first partitioned by edge class and then ranked within each partition with respect to cost. Starting with the Class 3 edges, each search resource is then initially allocated an edge to search. Where more than one resource shares a starting vertex, edges incident upon the vertex are assigned sequentially in order of class partition (Class 3 first) and in increasing cost within each class partition. This procedure is detailed in Algorithm 7.13. Unallocated edges which fall within the specified search partition are then allocated to search resources such that the edge selected is that which increases the current path length by the least amount but does not exceed *path_limit*. The increase in the path length is the shortest path length connecting the edge to the end of the current path, including the length of the edge. This method of path generation is detailed in Algorithm 7.14.

7.3.5 Assigning Resources to Implicit Edges

Class 1 and Class 2 edges whose lengths fall within the visibility cover of a resource positioned at their end vertices are termed *implicit edges* as they do not need to have their length explicitly searched in order to be visibly covered. Implicit edges not comprising an edge in any search path, but having their required end vertices (either vertex for a Class 1 edge and both vertices for a Class 2 edge) in some search path, are searched by that resource. An implicit edge is considered to be searched if these required vertices are 'delayed at' by the resource waiting at each vertex for $DELAY$ minutes, searching the edge length visible from this vertex for the subject during this time. If more than one search resource visits a required vertex of an edge, the resource which currently has the

Algorithm 7.13 function *base_arc_allocation*()

```
done ← 0
while (done ≠ resources) do
    for k = 0 to resources do
        find the first resource k not yet allocated an initial edge
        base ← start_k
    end
    ct ← 0
    for k = 0 to resources do
        if (start_k = base) then
        ct ← ct + 1
    end
    base_list ← all edges which emanate from base
    base_ct ← number of edges incident to base
    sort base_list by edge class and in increasing cost
    within each class partition
    j ← 0
    for k = 0 to ct do
        path_{k,0} ← base
        path_{k,1} ← second vertex of edge in base_list ranked at position j
        time_k ← c_{path_{k,0},path_{k,1}}
        num_k ← 2
        done ← done + 1
        if (j = base_ct − 1) then  j ← 0
        else
        j ← j + 1
    end
end

end
```

Algorithm 7.14 function *nearest_neighbour(partition)*

```
    initialize edges and vertices in partition as requiring searching
    base_arc_allocation()
    for k = 0 to resources do
        limit_k ← 0
    end
    while (there exists a search resource not at its limit and required edges to assign) do
        // sequentially assign edges, one at a time to each resource
        k ← resource with least number of allocated search hours and under path limit
        if there exist unallocated edges which can be searched by k within path_limit then
            add edge which results in least increase in path length to end of path of
            resource k via shortest edge path
        end
        else
        limit_k ← 1
    end
end
```

path of smaller length is allocated to delay at the vertex.

It should be noted that the search of an implicit edge in this way does not guarantee absolute detection of a subject located on the edge as an explicit search of the edge would under the detection assumptions outlined in Chapter 5.

7.4 Adapting Approaches In the Literature

Win [172] developed an algorithm to solve the (single) Windy Postman Problem (WPP) on Eulerian graphs, based upon a minimum cost flow approach. Win's approach finds the least cost orientation for a single tour. Such an approach can be adapted to the k-WPP in one of two ways.

1. Route then divide — find a single route via Win's method and then divide this into k connected partitions, one for each search resource [54, 151].

2. Cluster then route — partition the edges requiring searching into clusters, one for each resource, and then find the least cost oriented tour through each cluster [12, 108].

Before Win's algorithm is applied it is necessary to ensure that the subgraph formed by the edges requiring searching is connected. This can be achieved by connecting sub-components via a Shortest Spanning Tree (SST). The resulting graph is then augmented via Edmonds and Johnson's [46] minimum weight matching approach to obtain an Eulerian graph. Having transformed the problem to that of an Eulerian graph, Win's algorithm can then be applied to find the least cost tour through the graph. If the search base is not incident upon any edges requiring searching and, hence, is not a vertex on the tour, the base is inserted into the tour via a least cost edge connection. A path solution, rather than a tour solution, can be found by deleting the greatest cost edge incident to the search base.

Win's algorithm for the WPP is now outlined.

7.4.1 Win's Algorithm

1. Orient each edge in the graph in the least cost direction (ties are resolved arbitrarily) to obtain a digraph, $D_G = (V, A)$.

2. Create a new digraph, $D' = (V, A')$, having 3 edges for every edge in D_G with the following properties:

   (a) **edges and edge capacities:**

   - one edge identical to that in D_G (*i.e.*, edge in least cost direction), with infinite capacity;
   - one edge in the reverse direction to that in D_G (*i.e.*, edge in greatest cost direction), with infinite capacity; and
   - one auxiliary edge in the reverse direction to that in D_G, with a capacity of 2 and cost $= 0.5(\text{cost}_{\textit{greatest cost direction}} - \text{cost}_{\textit{least cost direction}})$.

   (b) **vertex demands:**

   - for each vertex i in D_G the demand is calculated as:

   $$d_i = outdegree_i \text{ in } D_G - indegree_i \text{ in } D_G$$

   where a negative demand represents a source of supply.

3. Find the minimum cost flow over D'.

4. Using the optimal flow values obtained in 3, create a digraph $D'' = (V, A'')$. This is done by examining the flow of each auxiliary edge such that:

   (a) if the flow = 0: place $(y^*_{ij} + 1)$ copies of the edge, in the least cost direction (*i.e.*, the original orientation is maintained) in A'';

(b) if the flow = 2: place $(y_{ji}^* + 1)$ copies of the edge, in the greatest cost direction (*i.e.*, the original orientation is reversed) in A'',

where y_{ij}^* = the flow through the edges oriented in the least cost direction and y_{ji}^* = the flow through the edges oriented in the greatest cost direction.

5. D'' is now Eulerian and any Eulerian diwalk through it will be a Windy Postman Tour for the original graph.

7.4.2 Route Then Divide

A single tour can be found over those edges in the graph requiring searching via the above method. This route can then be divided into k-paths by the following steps.

1. Orient the route in the direction which results in the least total traversal time.

2. Allocate the first search resource a path beginning from the search base and traversing the tour in this direction until the search hours accumulated by the addition of the next edge in the tour would exceed *path_limit*. Halt the path at this point and delete the last segment of the constructed path if it contains solely non-required edges.

3. Allocate the next search resource the shortest path from the search base to the beginning vertex of the next edge in the WPP tour which requires searching. Construct a path from this edge until *path_limit* is reached, by following the direction of the WPP tour determined in 1. Delete the last segment of the path if it contains solely non-required edges.

4. Repeat step 3 until all required edges are included in the path of a resource or all available resources are depleted.

5. Seek time improvements by reversing the direction of searching of each resource's path. This is achieved by first directing the resource to the last vertex in the path of an edge requiring searching (via the shortest path), and deleting the initial path between the search base and the first vertex of the tour.

The heuristic method is illustrated in Figure 7.5.

7.4.3 Cluster Then Route

We now describe a method to first partition the primary search area graph, before applying the adaptation of Win's algorithm to determine a search path through each partition.

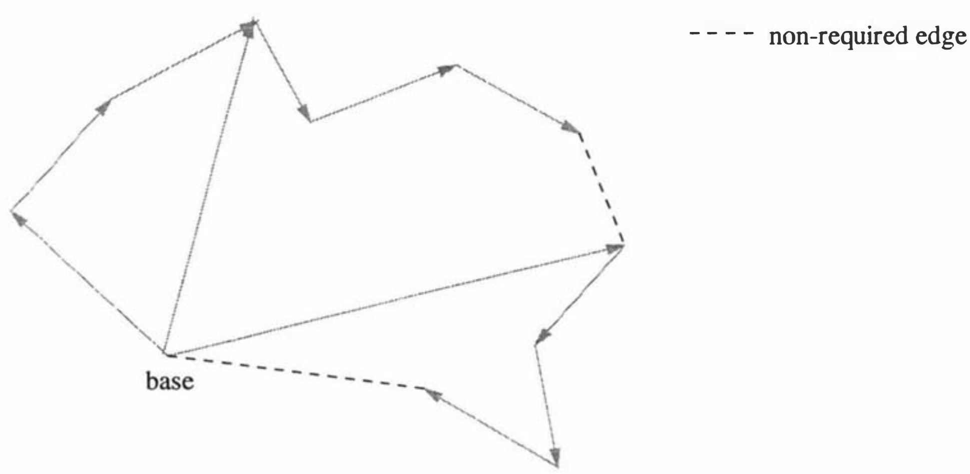

Figure 7.5: Division of a WPP tour into k components

7.4.3.1 Partitioning the Search Area

A heuristic method is developed to partition the primary search area into k partitions, where k denotes the number of resources available for searching. The method assumes that the number of triangular regions present exceeds k. The method can be described by the following steps.

1. Calculate the area covered by the primary search area, weighted by terrain density[4]

$$\text{total weighted area} = \sum_{i}^{ntri} area_i \times density_i$$

   where $ntri$ = the number of triangular regions.

2. Calculate the target weighted area for each resulting search partition as

$$\text{target weighted area} = \frac{\text{total weighted area}}{k}$$

3. An initial partitioning is found via the triangular regions incident upon the search base vertex as illustrated in Figure 7.6.

4. If the initial number of partitions does not equal k:

   - if the number of partitions exceeds k, recursively amalgamate the two partitions of least weighted area into one partition until the number of partitions equals k;

[4]The primary search area is defined so as to include entire triangular regions, not fractions of these.

- if the number of partitions is less than k, recursively partition the partition with greatest weighted area into two, until the number of resulting partitions equals k.

5. Move the partition lines in proportion to the target weighted area, in an iterative approach, such that the weighted partition area of each partition is no more than 10% different from the target weighted area.

6. Allocate each edge requiring searching to exactly one search partition.

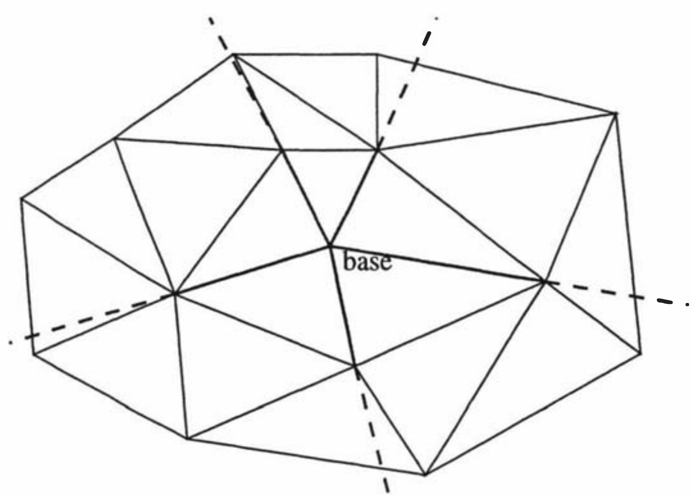

Figure 7.6: Partitioning the primary search area via the regions incident to an interior base vertex.

A weighted measure of area is utilized so as to place greater emphasis on those regions more likely to contain Class 3 required search edges with the intention that resulting partitions will contain similar proportions of these edges. The area covered by a partition is approximated by first determining the number of vertices of each triangular region which fall within the polygon described by the search partition. A ray test is conducted to determine whether a vertex lies within the partition. The ray test method is described by O'Rourke [128] and determines the number of times that a ray, through the vertex, crosses the polygon. If this number is odd, then the vertex lies within the polygon describing the partition. The area contributed by a triangular region i falling, in some part, within the partition is approximated by :

$$approx\_area = \frac{area_i \times density_i}{\text{number of vertices of } i \text{ within search partition}}$$

The allocation of edges requiring searching to exactly one partition, and hence to exactly one search resource, is determined as the final partitioning step. If both end vertices of an edge fall within a partition, the edge is allocated to that partition. If, however, the two vertices lie within two separate partitions, the edge is allocated first to the partition with the smallest total length of required edges, and then to the partition where its inclusion will make the degree of the vertex within that partition even. If this is not possible, a random allocation is made. The design of this allocation scheme is to reduce the later deadheading required to connect vertices of odd degree when constructing paths.

A path is then constructed through each search partition via the adaptation of Win's algorithm, such that each edge requiring searching is on a path.

7.5 A Special Case

7.5.1 A Minimum Weighted Matching Approach to Path Creation

A special case of edge searching exists where all edges of the primary search area are included in the visibility cover if and only if both of their end vertices are traversed. Thus a visibility cover of the entire region will result if a vertex cover is found. To find a vertex cover, represented as k disjoint search paths emanating from a common search base, a minimum weighted matching approach is developed.

1. If an odd number of vertices exist in the primary search area graph, G_P, amalgamate the two vertices connected by the shortest path into one vertex. The resulting number of vertices is represented by n.

2. Determine a minimum weight perfect matching over G_P utilizing any approach, such as Derig's FORTRAN code [38]. The matching will select $n/2$ disjoint edges which cover all vertices in the least cost way.

3. If $k = n/2$, each search resource is allocated to search an edge in the matching as illustrated in Figure 7.7. The path constructed is the shortest length path from the base, through the edge. Otherwise steps 4–10 are executed. These steps are depicted in Figure 7.8 for a small problem.

4. All edges connected by the matching procedure are duplicated and then 'compressed' to form pseudo-vertices.

5. Perform a minimum weight perfect matching over the resulting pseudo-vertices using, as weights, the shortest path lengths between pseudo-vertices.

6. Duplicate each edge in the matching so that an Eulerian diwalk can be found through the edges comprising the pseudo-vertex.

7. Recursively perform steps 4–6 until the number of pseudo-vertices required is greater than, or equal to k, for the first time.

8. If the number of pseudo-vertex=k then the edges comprising each pseudo-vertex provide a search tour for each resource, otherwise pseudo-vertices are amalgamated until k pseudo-vertices remain. This is achieved by a least cost rule which successively amalgamates the two pseudo-vertices which when connected by the shortest path have a resulting Euler diwalk, through their edge components, of minimal length.

9. Each resulting pseudo-vertex is 'uncompressed' to its individual edge components and an Eulerian diwalk is found through the connected edges.

10. The vertex, within each pseudo-vertex, having the shortest path to the base is identified. A path is then found through the pseudo-vertex by deleting the edge with the largest cost adjacent to this vertex as illustrated in Figure 7.9. Further duplicated edges may be able to be deleted to shorten the length of the final search path if the starting vertex of the path has degree 2. This is shown in Figure 7.10.

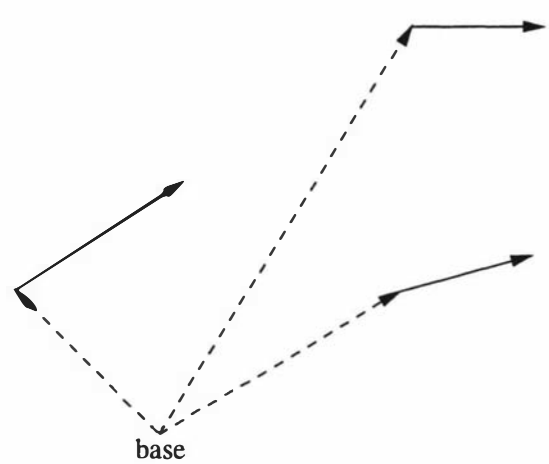

Figure 7.7: k search paths over k matched edges.

The method is designed for a primary search area which has been defined such that it can be searched from the search base with the available k resources. If this does not hold, then steps 4–6 are halted when the Eulerian diwalk through the edges comprising a pseudo-vertex exceeds *path_limit*. The pseudo-vertex is the decomposed into its original two halves such that the resource's time limit is not exceeded. The clusters of edges comprising each pseudo-vertex are then evaluated as to their POS contribution, with the clusters of higher POS return being allocated to the limited number of resources available.

The method utilizes a minimum cost weighting approach where cost is measured in terms of traversal time, c_{ij}, and POS is implicitly incorporated in the definition of the primary search area being that area containing those regions with highest POS return. An alternative approach would be to instead minimize the $c_{i,j}$/POS return in order to link together components returning maximum POS for time.

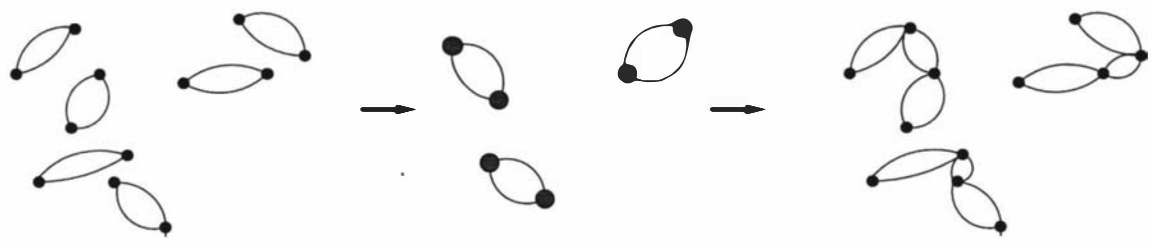

Figure 7.8: An original matching compressed to pseudo-vertices and then expanded to give three search paths.

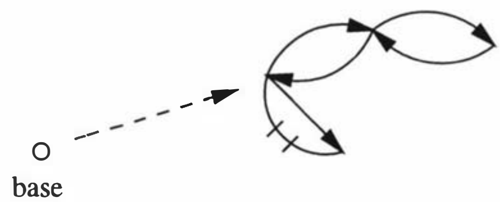

Figure 7.9: A search path from the search base through a pseudo-vertex.

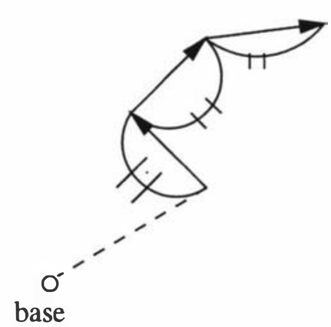

Figure 7.10: A search path beginning from a vertex of degree two.

Where search resources can be transferred to particular regions of the search area, *eg.* by helicopter or 4-wheel-drive vehicles, each resource is transferred to a particular pseudo-vertex. The Eulerian diwalk is directed in the least cost direction or via another objective such as a biobjective on cost and POS return over time. The starting vertex can be selected on the basis of the vertex incident to the edge with greatest cost, that is most easily accessible, or the vertex which forms the smallest length path for helicopter or vehicle, when linked with other drop-off points. This variant of the problem is a bilevel routing problem. The last edge in the resulting tour is deleted to form a path.

7.5.2 Adapting the Special Case to Region Searches

If a subset of triangular regions requires searching then we can transform the search region graph to its dual, with the 'centre' of each triangular region being a vertex in the dual. Regions are then searched if the vertex representing their centre is included within a resource's search path.

A vertex cover of those triangular regions requiring searching is desired and can be found by following the minimum weighted matching procedure outlined above. Obviously the searching of each region in particular, or each vertex in the dual, is not as straightforward as in the primal graph and becomes a subproblem in itself. Traversal and search times are associated with both the edges and the vertices in this graph. (The graph is outlined more specifically in Section 6.9.5.)

7.5.2.1 Adapting the Vertex Cover Approach to Non-special Cases

Where the primary search area graph consists of a mix of edge classes we can apply the matching approach of the special case to detected subgraphs which exhibit the special case attributes. Alternatively, additional edges could be added to the graph to transform the graph such that it meets the special case criterion.

7.6 Improvement Strategies

Once the search paths have been determined via initial heuristic approaches, improvement strategies can be applied to seek more efficient paths that obtain an increase in the objective function value. The initial search paths can be improved by transferring and exchanging edges from one search path to another, and by interchanging the order of searching of tasks, or altering the traversal direction of a portion of the path, within a single search path. A purely "greedy" approach which accepts improvements to the objective function as soon as they are encountered is computationally less expensive than approaches evaluating all possible path improvements. Such a method is dependent upon the order of comparisons, and any subsequent insertions and deletions. Specific improvement strategies include the following:

k-interchange: where a connected segment[5] of k regions within a resources' path is interchanged with another segment of length k also within the resources' path;

exchange: where single search regions or connected segments of search regions are exchanged between the paths of resources;

switch: where the order of searching a segment of a resource's path is reversed;

insert: where unallocated search regions are inserted into the path of a search resource not yet at its search hour limit;

transfer: where search regions allocated to one resource are transferred to another resource.

A single improvement strategy can be followed or a combination of several strategies, with the ordering of these resulting in differing solutions.

As the SAR problem exhibits dynamic properties, there exists a trade-off between the computational time involved in seeking improvements to initially generated search paths and the value such improvements add to a problem that is altering over time. In particular, if the information on which the path planning was generated changes significantly, such that different search paths are required to accurately address the problem at hand, the time spent seeking improvements can be made largely redundant if only a small portion of those paths have been searched prior to the change in information. Additionally, as the information on which planning is based is uncertain, paths cannot be guaranteed to produce the type of results predicted by a high POS return, such as detection of the subject or clues. It is for these reasons that we do not implement improvement strategies within the path generation phase of the SAR simulation.

7.7 Bilevel Routing

When resources are transported to the beginning of their route via a helicopter or 4-wheel drive vehicle, location-routing is invoked, as is a bilevel transportation problem with side constraints. A two level decision-making process is required to determine where to begin

[5]Segments of connected edges may be defined in number by a randomly generated number or set as a given parameter.

a search resource's path and how to construct that path. Additionally, the route taken by the transport vehicle must also be determined.

The initial phase of the problem exhibits a TSP structure where the route of the transportation vehicle visits the starting vertices of each search resource's path with an objective of minimizing time. Additional capacity constraints exist which may require more than one tour to be constructed, and a priority ordering may exist as to which resources to transport first, as in the TSP with precedence constraints [51].

This problem is viewed as an extension of the current framework and, as such, we do not address specific solution methods for it within this thesis.

Reconnaissance heuristics can also be applied in a Phase Two search in the form of a trail search. Further search heuristics with application to the general Phase Two search are examined in the next chapter.

CHAPTER 8

General Search Phase

Physical searching of the triangular regions of the search region graph can be achieved in a number of ways. A selection of these possible approaches is now investigated with the objective of determining those which are the most efficient. Efficiency is defined from the perspective of the search objective and also in realistic consideration of the constraints faced in the field.

8.1 Prior Edge Searching

In the instance where the full visibility coverage model is utilized (as described in Section 5.1.8) and edges are traversed at a search speed, an edge search extends visually into the interiors of the triangular regions adjacent to that edge, up to the visibility horizon of the searcher. The visibility horizon of the searcher in this instance is given by the visibility measure of the triangular region. The effect of such searching can be seen by considering the analogy of a paint brush; where the width of the brush is equal to the sum of the visibility measures of the adjacent triangular regions. As a search resource searches along the edge, the terrain which falls within the visibility cover of the resource is that region painted by the brush, as illustrated in Figure 8.1. Under changing search conditions the visibility measures of the triangular regions will also alter, hence the area of the triangular region falling within the visibility cover may vary along its edges.

Future search strategies that require the physical search of a triangular region must then evaluate whether any of the terrain represented by that triangular region has been 'painted' and hence included in the visibility cover. This terrain may be excluded from future coverage or re-searched, increasing the POD_{cum} level of that already covered portion of the region.

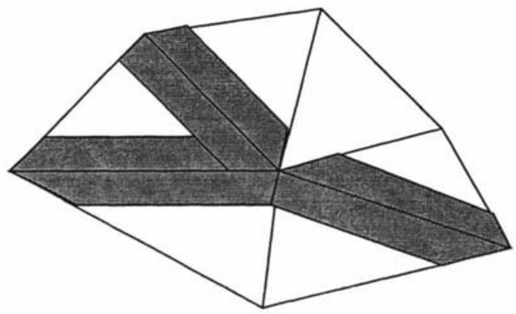

Figure 8.1: Paint brush analogy of the visibility cover arising from edge searching.

If prior edge traversal of the search region graph has occurred then the three edges of any triangular region will have been included within the visibility cover of a resource, in the following scenarios:

- the three edges defining the region are Class 1 edges and two of the three vertices of the triangular region have been visited;
- two of the three edges comprising the triangular region are Class 1 edges and the remaining edge is in Class 2, and the end vertices of the Class 2 edge are visited;
- one edge is in Class 1 and two edges are in Class 2, or all three edges of the region are in Class 2, and all three vertices of the triangular region are visited;
- one edge is in Class 3, and the remaining two edges are both in Class 2 or comprise one Class 1 edge and one Class 2 edge, and the Class 3 edge and the remaining vertex of the triangular region are traversed;
- one edge is in Class 3 and the remaining two edges are in Class 1, and the Class 3 edge is searched;
- two edges are in Class 3 and the remaining edge is either a Class 1 or Class 2 edge, and the two Class 3 edges are searched; and
- all three edges of the triangular region are Class 3 edges and all edges are searched in their entirety.

These instances describe the minimal requirements for the three edges of the triangular region to fall within the visibility cover of the problem. They also represent the minimal requirements for the entire triangular region to be included within the visibility cover of a resource, if the density of the edge vegetation along implicitly searched edges and that of the vegetation of the triangular region is the same, or the vegetation within the region interior is less dense than along the edges. This result is also dependent upon the relative lengths of the implicit edges in comparison to their visibility measures. If these conditions do not hold then the area represented by the triangular region may not completely fall within the visibility cover.

If the triangular region is not visibly covered via edge searching but all of the edges of the region are in the visibility cover (having the minimal edge and vertex visitations as described above) then the shape of the portion of the triangular region falling outside the visibility cover will be described by one of the following:

- A triangle with the same proportions as the initial region — if all three edges are Class 3 edges and are each searched, but the centre of the region[1] is located at a distance greater than the visibility measure from all of the edges.
- A triangle if two explicit edges are present and searched, and the third edge is an implicit edge. In this instance the triangle depth is reduced on the edges corresponding to the explicitly searched edges as represented by Figure 8.2.

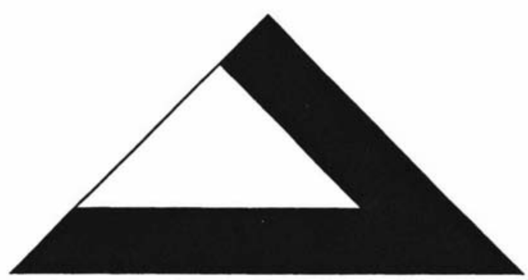

Figure 8.2: A triangle reduced in depth along the two searched edges.

- A triangle (being reduced in depth on the side of the explicit edge) if one edge is an explicit edge and is searched and the remaining two edges of the region are Class 1 edges.
- A "Y-shape" if all three vertices are visited but no edges are traversed, and the visibility horizon of the searcher at each vertex does not extend as far as the centre of the triangular region as depicted in Figure 8.3.

  If the visible areas overlap, a "curved" triangle will result as illustrated in Figure 8.4.
- An "arrow" if only two vertices of the triangular region are visited, as depicted in Figure 8.5.

[1]The centre of the triangular region being defined as the interior point which is equidistant from all

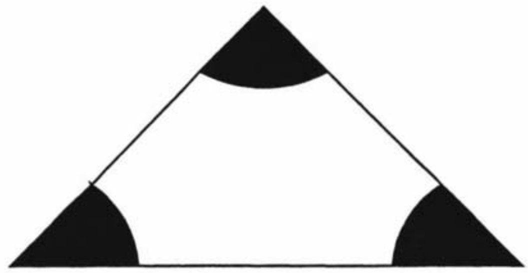

Figure 8.3: "Y-shape"

Figure 8.4: "Curved" triangle.

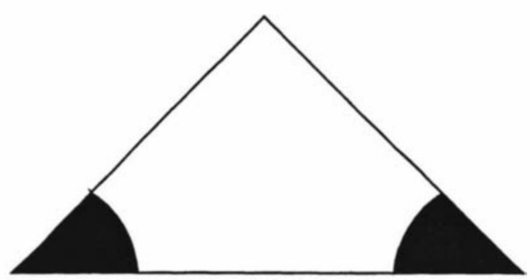

Figure 8.5: "Arrow shaped region".

- A triangle with a "bite" taken out of it, in the case where an explicit edge is searched and the remaining vertex of the region is also visited. This situation is illustrated in Figure 8.6.

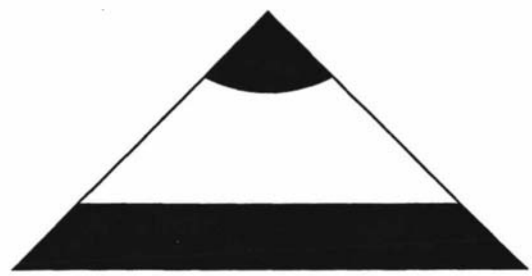

Figure 8.6: Triangle with a "bite" removed.

If edge coverage is incomplete then the following cases can be observed:

- if only one vertex of the triangular region is visited a triangle with a "bite" taken out of it remains;
- if only two vertices of the region are visited, an "arrow" shape represents the uncovered area of the triangular region; and
- if all three vertices of the triangular region are visited a "Y-shape" represents the uncovered area of region.

It would be possible to preprocess the triangular regions comprising the search region graph with regard to the type of coverage which would be required to visibly cover each region under the existing weather and light conditions. Each region could then be categorized by these coverage requirements, with re-processing occurring with changing conditions.

We do not consider it practical to search the exact portion of a region falling outside of the visibility cover when the shape of this portion is irregular, with irregularity being defined as anything more complicated in shape than a triangle and including the 'Y-shape', 'arrow shape' and the 'triangle with bite removed'. The 'curved' triangle could be approximated by a triangle in most instances without incurring a great increase in redundancy as the percentage area error is likely to be small on average. The triangular region as a whole, or the approximate triangle shape remaining outside the visibility cover after the edge searching phase is complete, can be searched by a number of methods. A selection of these is now outlined.

edges.

8.2 Methods of Individual Region Coverage

A common search technique employed in the field is a sweep search where searchers move abreast of each other, in one line, following a set bearing. Applying this method to searching a triangular region over varying terrain would have the advantage of being consistent with current practices. Currently such sweeps are executed by following the contour of the terrain, wherever possible, to conserve the energy of the searchers (New Zealand Search and Rescue course, April 1996). For this reason it is also preferable to begin a sweep at an uphill position, moving downwards on consecutive sweeps.

8.2.1 Region Sweep Search

The ***Region Sweep Search*** technique models such a search when the width of the search corridor[2] is large enough to cover a region in one sweep. The sweep begins from a chosen edge of the region — the base line — with a team of resources searching the region from this edge to the opposing vertex of the region. It is assumed that adjacent regions are not visibly covered, *i.e.*, these regions are not 'seen into' by the end searchers. This will be an accurate assumption when regions are steeply angled to one another at boundary intersections and is consistent with the visibility assumptions of the model.

The number of searchers required to conduct a sweep search over a region will depend upon the number of searchers available and the desired POD level of the search (fixed by the spacing of the resources). Assuming that the region can be searched within the given time constraints, the number of searchers required to search the region in one sweep is calculated as:

$$\text{number searchers} = \frac{\textit{searchable base width}}{\textit{searcher spacing}}$$

where the *searchable base width* = (*length of starting edge - covered edge length*) and the *covered edge length* is the depth of the region already falling within the visibility cover from any prior edge traversal. If the portion of the region already in the visibility cover is re-covered, then the searchable base width is equal to the length of the region edge from which the sweep commences.

Classical search theory formulae can be utilized to calculate the time needed to execute such a search. We assume that a single sweep of the region is executed only when the length of the region's diagonal can be traversed by a searcher within their available search hours. If this were not the case then either searcher-replacement would be necessary or the region would need to be further partitioned. In searching a region from

[2]The combined width of the spacings between the searchers in the team plus twice the visibility measure of the region.

the starting edge across to the opposing vertex the assumption is also made that the team narrows their search spacing as this vertex is approached, so that only this region is searched. This effectively increases the POD level by a uniform rate from the starting edge to the finishing vertex. This is illustrated in Fig 8.7.

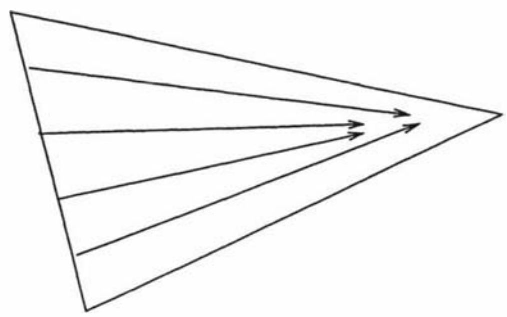

Figure 8.7: Pattern of searcher spacing when conducting a sweep of a triangular region.

8.2.1.1 Base Line

The region edge that the sweep will begin at is ideally chosen as the one which allows the ensuing sweep to move with the contour of the land. If the contour of the land is not changing significantly then a number of choices may be available. In this instance the shortest edge can be chosen to reduce the number of searchers required, or the edge resulting in the quickest search time can be selected. Alternatively, if the search resource is to proceed to search another region after the current sweep is completed, the edge which best facilitates the quickest search of both regions at the desired POD level is selected as the starting edge.

The efficiency of the region sweep method depends on how the sweeps are conducted, not only at the micro-level of each triangular region, but also at the macro-level, where sweeps are scheduled and routed over a selection of triangular regions which form the resource's search assignment. It may not be time-efficient, over the entire search assignment, to sweep each region via its optimal pattern.

8.2.1.2 Search Shape

Sweeping a rectangular shape is more efficient in terms of searcher utilization than a triangular shape as the same spacing between searchers can be maintained throughout the movement, without a "narrowing" occurring as the sweep moves from one side of the region to its opposing vertex (assuming that searchers are confined to searching only the current region rather than a union of regions). It would be trivial to construct an elongated triangular region that would be quite inefficiently searched if done so individually. In order to avoid the 'narrowing' of searcher spacing as the sweep nears the terminal vertex of the sweep, regions can be "pattern matched" into groups of regions which together form a surface that is more rectangular in nature.

How such an amalgamation of regions is to be swept then becomes a modelling decision. Regions could be swept together or some component of the grouping could be searched as one. This may result in the direction of movement being angled across a region into a neighbouring region. A disadvantage of this approach is that the natural or man-made boundary guides represented by the edges in the TIN model are lost as navigation guides as searchers "overflow" into neighbouring regions. Additionally, later confusion may result as to which areas have actually been searched and which have not.

An ideal grouping of regions would be one which coalesces regions of similar priority rating and terrain density (to avoid altering the searcher spacing when moving between regions so as to maintain the same POD level), and which contains no internal boundaries such as streams. The number of regions that could be amalgamated into one region would be constrained by resource availability.

8.2.2 Width Strip Search

The ***Width Strip Search*** is similar to the Region Sweep Search, however, it is assumed that there are not enough searchers to cover the region in one sweep and so it is searched in multiple strips. Searchers sweep across, and back over, the region in strips, remaining parallel to one edge (labelled the base edge) as illustrated in Figure 8.8. Each sweep is a search corridor in width and searchers ideally move with the contour of the terrain. Searchers pivot on the inside searcher when positioning to search the next strip, following markers laid on the previous movement. This method is both manageable and navigationally practical for a team of searchers in the field.

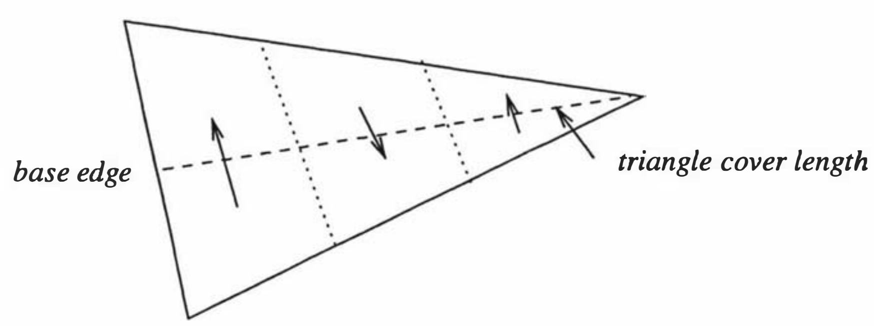

Figure 8.8: Width Strip Search.

If the inside searcher pivots to travel back along his previously marked path an overlap will occur as the resource searches the same terrain as on the previous sweep. As the markings along the path are likely to be visible from only a close distance it is not feasible to offset the return path any significant distance by traversing a path that maintains such markings within view. Alternatively, compass bearings can be utilized to offset the return path by a searcher spacing, in order to maintain a similar level of coverage over the entire region. The number of strips required to cover a triangular region by this method, when the pivoting searcher returns along their previous path, is calculated as:

$$n = \lceil \frac{\textit{triangle cover length}}{2 \times \text{searcher spacing}} \rceil$$

where the *triangle cover length* is the length of the edge bisecting the region vertex opposite the base edge with the base edge. This length is depicted in Figure 8.8. This search technique assumes that no prior edge search of the region has occurred or such covered portions of the region are re-covered, and the outside searcher begins his path on the perimeter edge. If re-coverage of areas covered by prior edge searching is not desirable then the search can begin at the perimeter of the uncovered region, with searchers pivoting when the perimeter of this uncovered region is encountered.

8.2.3 Recursive Perimeter Search

The ***Recursive Perimeter*** method of searching is executed by a team of searchers forming a line. The team first searches the perimeter of the region and then moves in towards the centre of the region, repeating the same perimeter pattern until the entire region has been covered. This is visually represented in Figure 8.9.

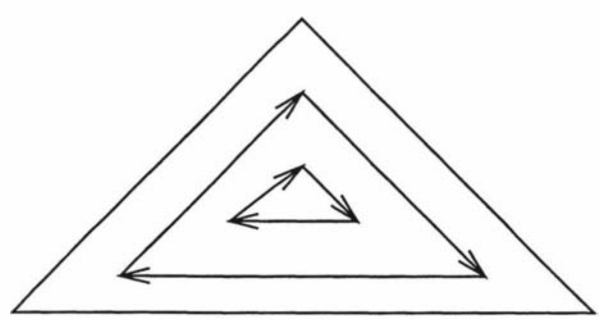

Figure 8.9: Recursive Perimeter Search.

The technique has similarities to the marine Box Search method where the box shape is replaced here by the triangular shape of the region and the search length is decreasing with each recursive pivot, rather than increasing.

The method has advantages of practicality on the ground when the inside searcher marks his path as he searchers. Then, when the team has completed the current perimeter search, they can pivot on these markings to ascertain that no ground is left unsearched or unnecessarily re-searched. Pivoting on the inside searcher when transferring between perimeters will hold the current position; it should also reduce difficulty in following any unclear signs as the searcher who laid the path markers will now be following them. It would also be possible to alternate between clockwise and anti-clockwise orientation when searching alternate perimeters to ensure that the searcher following the markings (and hence the same path as in the previous search) was travelling in the opposite direction to the previous movement. Ideally this would increase the likelihood of detecting anything previously missed. As the initial perimeter can be traversed in two possible orientations the orientation must be determined which results in the quickest overall search time.

Utilizing a complete sweep along each side of the perimeter will double search the overlapping corners unless the searchers prepare for the search of the next side by staggering a diagonal into the corner. However, this approach may prove difficult navigationally in difficult terrain and may impede communication between the searchers.

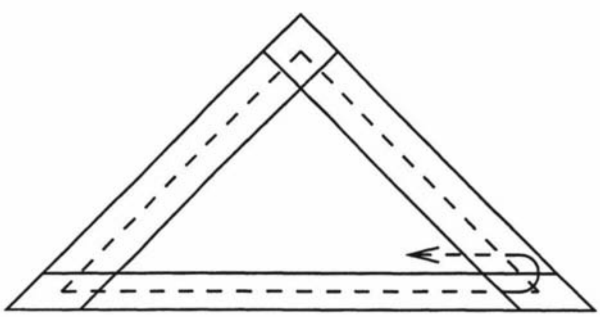

Figure 8.10: Pivoting between successive perimeters.

Disadvantages of the Recursive Perimeter search method include: the level of re-searching in the region 'corners' when changing the search trajectory; the direction of travel being defined by the region edges, rather than following the contour of the land; and searching inefficiencies when the remaining central area of the region to be searched is small relative to the size of the team's search corridor.

8.2.4 Recursive Triangulation

The technique of ***Recursive Triangulation*** extends the initial concept of triangulating the search region to obtain a TIN, to further triangulating each triangular region which cannot currently be included in the visibility cover by edge searching alone. The triangular region is assumed to have a planar face without internal barriers and the three edges of the region are regarded as its convex hull. The objective is then to recursively triangulate the region to the stage where the new edges of the triangulation can be traversed to obtain a complete visibility cover of the original region. If a single search resource is assigned to search this region then their route is found by solving the Windy Postman Problem (WPP) over the edges comprising the triangulation of that region. The solution can be found in polynomial time if the resulting subgraph is Eulerian. Hence a triangulation which generates even-degree vertices is desirable, as this reduces inefficiency, by eliminating the need for multiple edge traversals when augmenting the subgraph to meet Eulerian properties.

The z-coordinate of each point created by the triangular procedure is determined via interpolation. These new points are stored only temporarily in order for the triangular region at hand to be searched, and are then discarded to free computer memory. These points can effectively be viewed as Steiner vertices — the creation and use of such vertices could also be implemented if their inclusion reduces the cost of traversal by edge traversal alone.

The initial step in the Recursive Triangulation procedure is to find the centre of the triangular region, so as to partition the interior into three triangular regions. This is achieved by creating an edge between the centre point and each vertex of the triangular region. These edges effectively form the medial axis of the triangular region[3] [128] as illustrated in Figure 8.11. The medial axis is the set of points which are equidistantly closest to two or more points on the triangular region perimeter, the centre being equidistant from the midpoint of all three edges defining the perimeter. If the new set of interior edges do not form a cover for the terrain represented by the original region, a further triangulation is performed over the newly created triangular regions. Several ways in which this may be achieved are now presented.

8.2.4.1 Simple Triangulation

A simple triangulation can be performed by forming edges from the centre of the region to the midpoint of each perimeter edge, as indicated in Figure 8.12. In this triangulation the original region is visibly covered by a resource searching the resulting edges if the centre of each triangle in the triangulation is visibly covered by the resource, *i.e.*, the centre of each triangle is situated at a distance from each edge which is no greater than the visibility measure of the region. Alternatively, and more efficiently, the edges can be

[3] Also defined as the "skeleton" or "symmetric axis" of the triangular region [128].

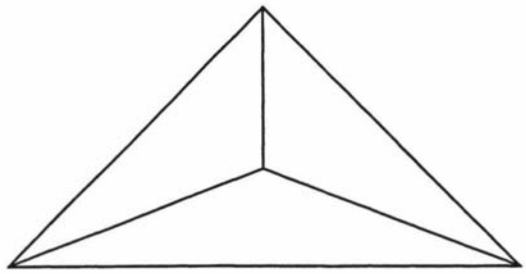

Figure 8.11: Medial axis of a triangular region.

traversed by a team of searchers in a line, such that the central searcher traverses the actual edge. In this instance a visibility cover is achieved if the centre of each triangle is positioned at a distance from each edge which is no greater than half the width of the team's search corridor.

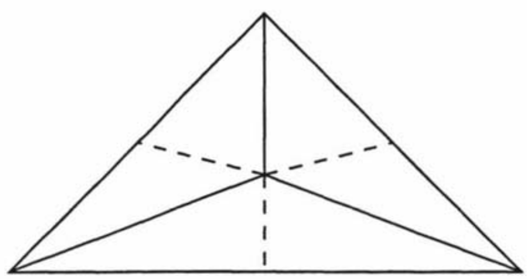

Figure 8.12: Simple triangulation of a triangular search region.

8.2.4.2 Delaunay Triangulation

Alternatively, a Delaunay triangulation which maximizes the minimum angle of newly created triangle faces can be utilized. Minimizing the presence of 'small' angles reduces the degree to which search regions overlap when the edges are traversed. This method can be applied by utilizing the centres of the triangles created in the previous phase as the newly generated formation points. This results in the triangulation depicted in Figure 8.13.

It can be seen that the newly created points from this phase form a central triangle, with a 'petal' formed from each side of the central triangle to each vertex of the original region (shaded in Figure 8.13). If the length of the edges of this central triangle are each less than or equal to the width of the team's search corridor, then traversal of the edges forming each 'petal' will be sufficient to cover the region. If these sufficiency conditions

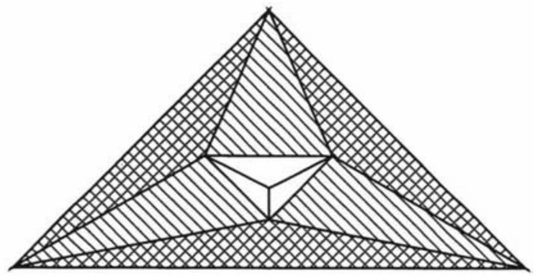

Figure 8.13: Delaunay triangulation of the initial medial axis triangulation.

hold, the three edges of the central triangle are redundant as traversal of the exterior edges will cover the area enclosed by the central triangle. If these interior edges are then removed the resulting triangulation has each vertex having even degree. This is unlike the simple triangulation method at the second phase of triangulation (illustrated in Figure 8.12) where all vertices, apart from the central vertex, are of odd-degree. A path over the region when these central edges are removed is shown in Figure 8.14.

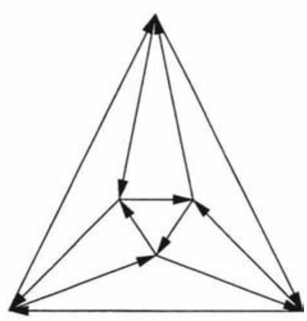

Figure 8.14: Petal traversal to cover a triangular region.

While this result is geometrically interesting, the practical application of such a search pattern is too complex. This is especially true when further triangulation phases are required to achieve a visibility cover of the triangular region via edge traversal.

8.2.4.3 Bisection Triangulation

Bisection triangulation recursively bisects the triangular region until traversal of each edge results in a complete visibility cover. This is achieved by first inserting an edge between the region vertex with the greatest interior angle, and the midpoint of the opposing region edge; thus the original region is bisected and two triangles are created. These two triangles are then further bisected if coverage has not yet been obtained.

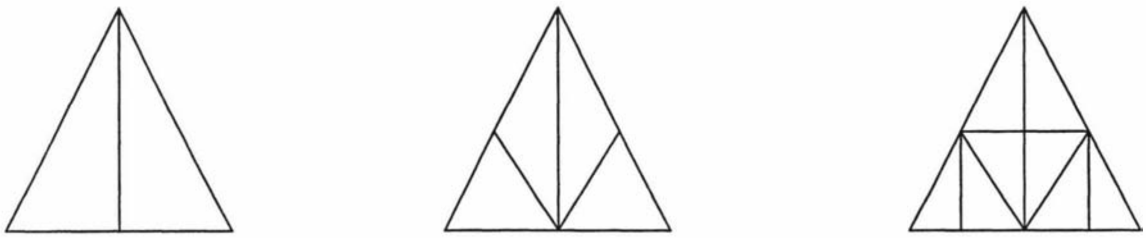

Figure 8.15: Bisection Triangulation — first, second and third bisection.

This process is continued until the region is visibly covered. Each triangle subsequently created is bisected at the next iteration by forming an edge between one vertex of the triangle and the mid-point of its opposing edge. There exist three candidate bisectors for each triangle face. A bisector can be selected via a local approach which maximizes the minimum of all the angles of the resulting two triangles; this is achieved by choosing to bisect the largest angle in the triangle. This procedure is illustrated in Figure 8.15. By bisecting, at each stage, the uncovered portion of the current triangle, the method aims to quickly arrive at a complete visibility cover.

Practically speaking, this method results in a triangulation which has edges that are more easily navigated than the other triangulation methods proposed as searchers can be directed via compass bearings from one point on the perimeter of the triangular region to another point on the perimeter.

8.2.5 General Terrain Traversal Considerations

If restrictions are lifted on searching only one region at a time, a greater realism and often efficiency, can be included in the region search. Such an example exists when a small hill is modelled by a number of triangular regions. Instead of searching each region one at a time, searcher effort can be reduced by searching the hill as one component, circling the feature with the contour of the land from its highest point downwards, to reduce fatigue and the traversal time required. Such a structure can be recognized by identifying all triangular regions which have a vertex of high elevation in common, where this vertex has an elevation value greater than all other vertices comprising the region. Such a search is illustrated in Figure 8.16.

It is not desirable to unnecessarily impose a triangular restriction on the search if a more 'natural' method is deemed to be practically more efficient. Searching along more 'natural' lines may involve searching polygonal regions defined by ridge lines, streams, rivers, tracks and vegetation breaks. These may be triangular regions as modelled by

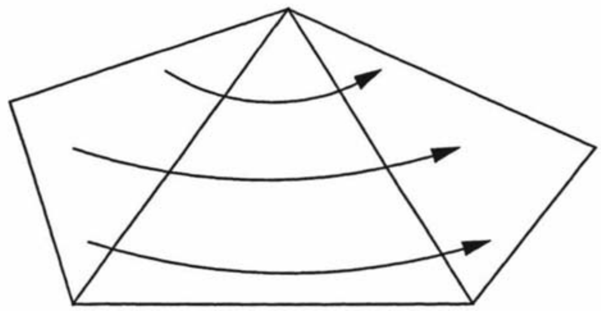

Figure 8.16: Hill search as one component.

the TIN or may cover a more extensive area. As the TIN model attempts to explicitly represent such natural features search regions should be able to be represented in terms of triangular components. It is also desirable that search regions do cover the same, or not greatly dissimilar, terrain types so as to require little variation in searcher spacing and hence maintain the same POD level throughout the search of the region. When selecting triangular regions which may be searched together in a continuous motion, one criterion which can be considered, in addition to terrain classification, is the angle formed between their adjacent planes — the closer that this angle is to 180^o the more desirable this is likely to be. Improvement upon initial combinations can be considered as a post-generation phase.

8.3 Modelling Traversal of a Triangular Region

As has been demonstrated, there exist many different ways to model the actual search method of a resource searching a triangular region. Additionally, the method selected can only ever be an approximation of the actual ground search undertaken in a real environment as the TIN is only a model of that terrain. In order to model the traversal of a triangular region by a resource at an equivalent level of modelling detail to an edge traversal, we approximate the *Width Strip* search method and search a single region at a time. The Width Strip search method is advantageous over the other proposed methods in that it maintains the integrity of parallel sweeps, can be achieved with a small number of resources and is navigationally practical. If a different search method or a more detailed model of movement is desired at some future stage, the search traversal module in the simulation programme is independent and easily replaced.

The approximation of movement of the point representing the search resource is from entry vertex a to exit vertex b, in sweeps directed parallel to the region edge (a, c). This is depicted in Figure 8.17.

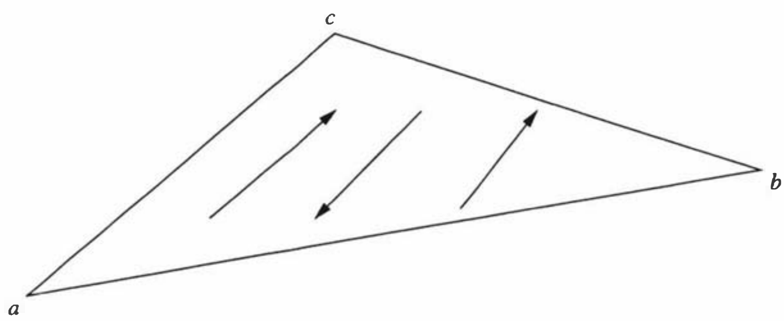

Figure 8.17: Traversal approximation of a triangular region.

The search speed of the traversal is approximated by the average of the speed from vertex a to vertex c, and the speed from vertex c to vertex a. The actual time required to search the region is then calculated by rearranging the classical search theory formula for search effort [150].

$$\text{search hours} = \frac{Z}{W \times V \times k} = \frac{C \times A}{W \times V \times k}$$

As we are modelling the search method by straight, equally spaced, parallel sweeps we can further substitute $C = W/S$.

$$\text{search hours} = \frac{A}{S \times V \times k}$$

This calculation can be evaluated at any time in the simulation to determine the fraction of the task completed so far.

8.4 Search Region Definition

Subdividing the entire search area into regions which particular search resources can be allocated to search is an important preprocessing stage of the search operation. This activity is presently done in New Zealand by someone familiar with the terrain of the area who can approximate, from experience, how long it will take a search resource to search a particular region [165].

A number of factors need to be considered in defining the search regions. LaValla and Stoffel [103] specify these to include:

- the density and difficulty of the terrain;
- the size and expertise of the resource that will be searching the region, and other available resources;

- the current and predicted weather;
- access to the region and available transportation;
- areas which have already been searched;
- clues found and their reliability;
- the avoidance of internal barriers; and
- that the regions are easily mapped and located (preferably visible in the field), with clear boundaries.[4]

Colwell [26] emphasizes the necessity of ensuring that features such as tracks and stream beds are also considered as search regions, even if they are narrow and small in size. Possible boundaries for search regions include: natural phenomena such as ridge lines, vegetation breaks or water networks; and man-made features such as roads, tracks or fence lines. Boundaries can also be defined by map coordinates, and followed by compass bearings or line of sight [103]. Boundaries not naturally defined can be marked as they are searched to guide searchers in adjacent regions or those who may be re-searching the area later [121, 159]. LaValla and Stoffel [103] note that boundaries may be added or deleted, by combining or subdividing regions as the search progresses, but boundaries should not be realigned, to avoid complications to search computations and management directives. However, McConaughy [116] disagrees with LaValla and Stoffel that search boundaries should remain unchanged over the various search phases. He instead advocates that altering the boundaries of search regions in between the reassignment of a region "creates a different look and feel to the search area for the searcher and often leads to different approaches or field tactics being used." This change in emphasis may allow different results to be obtained than those reached in previous searches of that area. McConaughy considers that altering search boundaries will also reduce the risk of boundaries being missed in the search process.

McConaughy recommends that graphical computer software be developed with the capacity for search managers to draw the actual area covered by each search resource upon the screen. This would enable direct computation of search statistics and hence avoid the complication of search computations that arises from altering search boundaries. McConaughy also advocates that the size of search regions should be reduced as the search progresses. Such a reduction in size results in an increase in POD over the smaller areas, as it allows for more thorough searching of an area in the same amount of search

[4]It is crucial that all boundaries are covered by at least one search resource.

time that larger areas were previously searched in. This has the psychological advantage of searchers being asked to search for a "needle" in a smaller "haystack".

McConaughy proposes four practical ways of reducing the size of search regions as an operation progresses. These are:

1. reassignment from search management of different search region boundaries with each re-coverage or subsequent shift;

2. subdivision of original search regions, including the reassignment of portions of a region previously assigned but not covered;

3. division of search regions in the field by search resources;

4. simultaneously assigning multiple resources to a high probability region which has previously been searched.

Perkins and Roberts [138] propose the "APE" method of subsectoring search regions in the field by the search group leader. APE consists of three phases; assessment, planning and evaluation. The method parallels the initial segmenting of the search area by search management. A search region to be subsectored is first assessed with the identification of boundaries, routes, open areas, and other features offering concealment. The leader then plans search tactics for these subareas, possibly eliminating some subareas. After the search of the subareas is complete, a detailed evaluation of the search is reported back to management along with any subareas that require re-searching. Subareas offering little concealment can be viewed as "transit areas between search problems" and may be searched less thoroughly.

8.4.1 Current Approaches

Colwell [20] recommends the following five-step approach to defining search regions:

1. Define a search region as the smallest area which can support a POA value. Each region should, where possible, have clearly defined boundaries and be able to be searched within a day, by one search resource.

2. Assign a POA value to each region.

3. Subdivide any search region which cannot be searched within one day into portions which can.

4. Each search region further subdivided is assigned a POA value equal to the POA value of the initial region, divided by the number of subdivisions.

Perkins and Roberts [137] divide the total search area into regions of similar terrain and of a size large enough to be searched within two hours. This method is strongly dependent upon the terrain gradient and the type of ground cover in any one area. LaValla and Stoffel [103] recommend that a search region be able to be searched within 4–6 hours (a half shift), while Carnes and Cooke [14] recommend dividing a search region into more than one assignment if more than 6 to 8 searchers are needed to search it. In New Zealand there is currently no limit on search time, with a search team being expected to search for an entire day and often sleeping in the bush for several nights. Recently, however, there has been suggestion of moving towards a soft regulation in line with overseas reports [165].

McConaughy [116] examines the trade-off between the overhead time required for task reassignments and the diminishing effectiveness of resources which spend a long time in the field between assignments. He observes that generally an hour, plus transportation time, elapses between when a resource returns to the base for reassignment and when they head back into the field, although he further notes that smaller teams and efficiently run bases may require less time. Further efficiencies can be gained by radio communication of reassignments to the resources in the field. As with other commentators, McConaughy notes from field research that after 5 hours of searching the observation powers of searchers fall off and that tasks of length 4 to 6 hours are "the most productive and require the least overhead cost."

8.4.2 Search Region Definition on the TIN

The TIN model has a number of advantages when considering the problem of search region definition. In particular, many of the TIN edges represent natural and man-made features. The edges will also often model vegetation breaks in terrain cover, enabling the definition of an average representative terrain cover for each triangular region. Hence the TIN edges provide a very good source for possible search region boundaries that are both realistic and manageable. By defining search regions based solely on the triangular partitioning of the TIN a natural and manageable division is achieved, for both search management and searchers in the field.

The model also allows analytical precomputation of approximate traversal times for any given region, based upon the structure of the TIN. This has the advantage that decisions can, if need be, be made independently of an experienced person local to the area. To determine search regions of a size which can be covered in a given time period by a detailed search method, adjacent triangular regions can be combined or larger triangular regions requiring greater search hours can be subdivided. The TIN model enables a straightforward and autonomous approach to the update of search regions in changing conditions as traversal costs are updated. Internal barriers can be avoided and search regions can be determined based upon the available search resources.

Search resources allocated to search two regions of different terrain types will need to alter their spacing between the two regions to ensure that both are covered to the same POD level, or alternatively, more searchers can be deployed to search the region of greater density. To avoid the necessity of this it is desirable that search regions comprise triangular regions of the same terrain density classification.[5] To avoid the additional searcher effort required to traverse obstacles internal to the search region, such as streams or fences, it is also desirable that such features form the boundary of search regions only. Additionally, it is preferable for resources to begin searching from higher elevations, as downwards traversal requires less effort and can be achieved more quickly.

If triangular regions are coalesced or partitioned into regions of a size that can be searched by one resource in the time allowed then the problem becomes an allocation problem.

Heuristic approaches to defining search regions on the search region graph are now detailed.

8.5 Search Region Definition Heuristic

One heuristic approach to search region definition is segmentation via the linear features of the TIN represented by edges. As previously highlighted, man-made and natural boundaries, or barriers, should be avoided as internal components of search regions to reduce search difficulty. It is assumed in the model that all such boundaries are represented by edges in the TIN. In particular, the only barriers to traversal explicitly considered in the model are streams and rivers (which are further classified as 'crossable' or 'non-crossable' depending upon past and present past weather conditions). To prevent these edges occurring interior to a search region, neighbouring triangular regions which share such an edge are precluded from being elements of the same search region.

A preprocessing procedure is defined to determine the search regions.

8.5.1 Preprocessing an Area

Each triangular region is considered in turn, to determine if the area of that region can be searched to a target POD level within the search period defined. A slight tolerance above this limit can be permitted, to acknowledge the likely efficiency of a resource searching an entire region over another resource being allocated to search the remaining portion from

[5]The four classifications modelled are detailed in Chapter 4.

another position on the graph. The time needed to search the region will be influenced by the search pattern (POD level) utilized, the gradient and terrain cover of the region, the physical area size, the current light and weather conditions, and the resource's speed. Any triangular region unable to be searched within this time constraint is partitioned into regions which are able to be searched within this time.

8.5.1.1 Region Partitioning

A triangular region can be divided into triangles of an area that can be searched in the available time by triangulating the region by one of the methods proposed earlier. Alternatively, the region can be subdivided into 'close to equiangular triangles' by creating internal triangles with at least two interior angles equal. This procedure is executed by selecting the vertex of the shortest edge of the region, which has the greater interior angle. A new edge is created by forming an edge between this vertex and its opposing original region edge; the edge is placed at the angle θ indicated in Figure 8.18. The procedure is repeated for each resulting triangle partition until all regions comprising the division can be searched within the given time constraint.

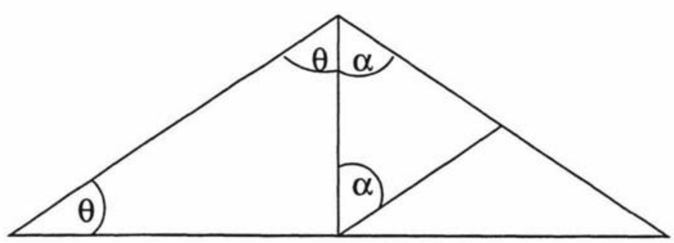

Figure 8.18: 'Close to equiangular triangles' partition of a region.

If search regions are desired which have a search duration very close to the duration of the search period (such that a single region comprises the sole search task allocation of a single resource) adjacent triangular regions can be amalgamated to form a single search region.

8.5.2 Amalgamating Triangular Regions

All triangular regions are explored to determine candidates for amalgamation. Such regions fall into one of the following four cases and are processed in this order.

8.5.2.1 Case One

A triangular region that has all three edges classified as streams or rivers is labelled as a search region.

8.5.2.2 Case Two

All triangular regions which have two edges classified as streams or rivers are considered. The time required to search such a region is calculated, together with the time needed to search the region that shares the remaining third edge. If this combined traversal time falls within the time constraint, both regions are combined into a single search region. This region can then be further augmented by the inclusion of adjacent regions that can be jointly searched within the time limit. The process continues by considering adjacent regions in an outwards movement, where the addition does not result in the crossing of any streams or rivers. This process is illustrated in Figure 8.19.

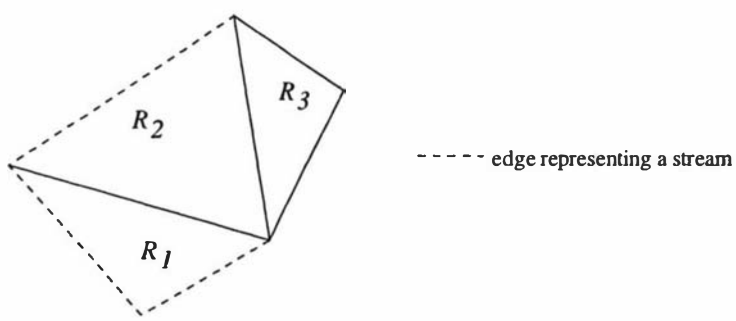

Figure 8.19: Amalgamation of adjacent triangular regions into a single search region.

When there exist a choice of adjacent regions at each further stage of augmentation, the region selected to be included next in the search region is that region which is most similar in vegetation cover to those regions comprising the current search region. When more than one candidate region still exists, a second criterion of selection can be utilized such as selecting the region which results in the most compact overall shape, or the one which results in the greatest increase in total search time whilst still satisfying the time constraint. The rationale behind the first criterion is to facilitate search coverage, while the second criterion is motivated by the fact that regions with greater search time requirements may be more difficult to include in search regions being constructed at a later stage.

8.5.2.3 Case Three

All triangular regions which have a stream or river as one of their edges are selected for consideration and incremented in size along their 'non-stream/river' edge, by examining adjacent regions as described under Case Two.

8.5.2.4 Case Four

Any remaining triangular regions that are not yet close to the search time constraint are considered for amalgamation with adjacent regions, where possible, by the process described above.

The situation where a stream or river edge occurs interior to a search region, and hence hinders searcher movement, is avoided by examining the regions in this order. Some triangular regions may not be able to be amalgamated by this process as, at the time they were considered, all other possible viable adjacencies had been included within another search region. Such triangular regions are then regarded as search regions in their own right.

Forming search regions prior to a search operation reduces the time required for associated tasks during the actual operation. However, they must be defined logically with respect to the terrain and past search statistics so that areas of low and high probability are not coalesced together, thereby producing search inefficiencies.

8.5.3 Trail-based POA Segmentation

An alternative segmentation method is one based on Colwell's research into Trail-based POA values, where search regions are defined around the main trails of the search area [27]. The POA values of the search regions are defined by the values assigned by reconnaissance search resources to possible decision points along the track(s) of the search area, in the initial phase of the search. The search regions may also be pre-defined, based upon available search statistics for the search. The method constructs search regions outwards from the tracks, utilizing natural boundaries to define the search region boundaries.

The approach of Colwell can be modelled on the search region graph by fixing the edges in the graph that represent tracks, as region boundaries. Constructing search regions so that each region has a boundary that is a track increases the accessibility of search regions via foot. However, in reality it may not be possible for all search regions to have a track as a boundary edge, depending on the frequency and location of tracks.

Segmentation is executed via a procedure that moves along a track, defining search regions either side of the track as it moves through the graph. The first track searched is the 'main' track of the search area. The procedure then repeats the search region construction for the unlabelled triangular regions adjacent to the remaining tracks of the region in the order that they branch off from the main track, followed by any separate and disjoint tracks that the region may contain.

The first triangular region to the 'left' of the track (the track forming an edge of the region) is labelled a search region and subdivided if it cannot be searched within the search time constraint; if it can be searched without subdivision, the regions adjacent to that triangular region, moving away from the track, are considered for amalgamation into a larger region which can still be searched within the given time restrictions. Preference for amalgamation is given to those candidate regions having most similar vegetation (a prepartitioning of the region with respect to vegetation classifications could be conducted). Streams and rivers are forbidden as internal components of search regions.

The triangular region to the 'right' of the track is then labelled as a search region and processed in the same way. The method is illustrated in Figure 8.20 for the first triangular region to the 'left' and 'right' of the track. The segmentation procedure then moves to label the next triangular region on the left of the track that is so far unlabelled. The procedure sequentially defines search regions in this fashion until the end of the current track is reached.

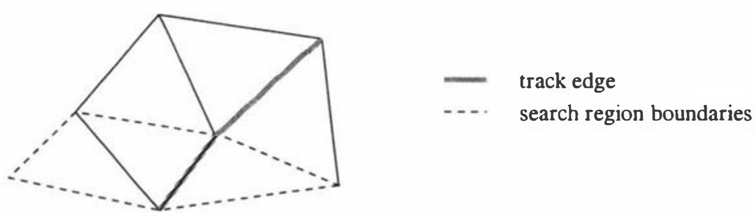

Figure 8.20: Trail based POA segmentation method.

At the end of this process it may be possible that some triangular regions not adjacent to any track remain unlabelled and, hence, not in any search region. These regions are then individually analysed to be either partitioned or coalesced with adjacent regions, as described in the previous heuristics.

8.5.4 Real-time Segmentation

An alternative approach to defining search regions is to make a pass, in real time, through all the triangular regions composing the TIN, dividing any which are unable to be traversed as a whole when searching under the current conditions. As the search paths are determined, combinations of adjacent regions are tested for inclusion in the same search path by a heuristic procedure that ensures that time and inner boundary constraints are not violated. This procedure is more flexible and adaptable to the dynamism of the operation, but it does require more processing time during the operation than other methods, as it is updated for changing search conditions.

8.5.5 Selected Search Region Definition Method

Within the actual SAR simulation developed we consider each edge and triangular region of the search region graph as a single search region. Restricting a search region to contain only one entity of the TIN ensures that the same vegetation is found over the entire search region and that the same search method can be applied uniformly over the region. Instead of seeking to create search regions by amalgamating adjacent regions, we mimic this process by allocating a sequence of search regions to a particular resource via the path generation heuristic methods detailed further into this Chapter. By constructing search assignments of a given time length in real-time, the method adapts to altering conditions and specific types of path assignments. An avenue for future research would be to program such amalgamation techniques and analyse their effective contribution to the search outcome.

An initial segmentation procedure is, however, executed when, after generating the TIN and allocating terrain types to each edge and triangular region, a triangular region requires more than six hours of searching by a team of four ground searchers (in the quickest search direction for a POD level of 50%, assuming perfect weather and light conditions). In these instances the region is partitioned further by approximating the centre of the region and inserting an edge from this point to each vertex of the region, to create three new triangular regions. The x and y coordinates of the centre are approximated by the average of the x and y coordinates of the three region vertices, with the z coordinate of the centre being interpolated from the plane described by the face of the triangular region. This segmentation process is illustrated in Figure 8.21.

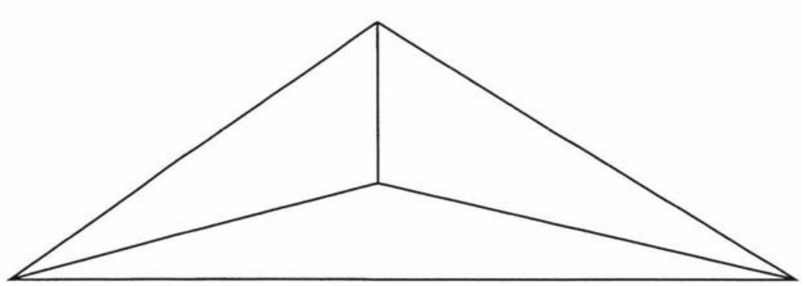

Figure 8.21: Segmentation of a triangular region.

The procedure increases the total number of vertices in the TIN by one, the number of triangular regions by two and the number of edges by three. Each newly created triangular region is assigned the same vegetation type as its 'parent' and each newly created edge is categorized as an untracked edge. If any of the new regions is still unable to be searched to a 50% POD level by a team of four searchers, within six hours, then that region is further subdivided utilizing the same procedure.

This method of segmentation was selected over other methods as it preserves the vertex as an entry point into other search regions.

The model does not consider partitioning a region during the search operation if a desired POD level can not be achieved by a team of searchers within the given time constraints. Instead the region is searched at the maximum POD level which can be attained and then re-searched if required. This can be viewed as a limitation of the current model.

8.6 Resource Allocation

The first question to address in determining task allocations for search resources is which regions to select for searching by a given resource? The second question is how best to sequence these? Ideally, regions will be sequenced by priority, and with respect to access and search times. When one region precedes another in this sequence, the choice of search pattern, and entry and exit points (which dictate the direction of travel) must be selected so as to efficiently link and search the regions. When sequencing regions it is necessary to achieve viable sequences, *i.e.*, those sequences which can be searched within the given time constraints and achieve approximate work equity among the search resources. Quantitative measures which result in such a viable sequence include Colwell's search priority measure [26] and ranking regions based on PSR or expected POS return. An alternative approach to ranking regions for searching is via an exponential smoothing method.

8.6.1 Exponential Smoothing Method

Golden *et.al.* [73] present a heuristic method for finding a single route to solve the time-constrained TSP. The method is iterative and utilizes exponential smoothing to estimate the profit worth of a unit of time at each iteration. The initial profit-time ratio is defined by an educated guess or from preliminary analysis, with the intention being that it estimates the value of the profit-time ratio of the optimal tour. The ratio is adjusted at each iteration by applying exponential smoothing, where the weight of the smoothing, α, can be varied between the values of 0 and 1 to generate a number of solutions. At each iteration the vertex selected to be included in the tour is that which provides an increase in profit-time ratio that is closest to, or exceeding, the current profit-time ratio estimate.

A criterion for selecting which search region should be the next to be included in the search path of a resource can be developed which is conceptually similar to the approach of Golden *et. al.*. Here the profit returned from the visitation of a vertex is the POS value returned from searching a region. Each region i which is able to be searched within the available time, at iteration l of the path generation, can be ranked via the following measure:

$$v_i = POS_i - R_l \Delta time$$

where POS_i equals the expected POS value to be obtained from the search of region i, $\Delta time$ represents the time needed to search region i plus the access time involved in connecting region i into the path of the resource, and R_l is the profit time ratio estimate for iteration l. At each iteration R_l is calculated as:

$$R_l = \alpha(\sum_{k=1}^{resources} \frac{P_k}{time_k}) + (1 - \alpha) R_{l-1}$$

where P_k is the estimated POS obtained from resource k's path and $time_k$ is the time length of resource k's path.

The approach has similarities to Colwell's search priority measure and the PSR criterion in ranking candidates based on a harvest rate, but differs in the initialization and updating of a target harvest rate in each successive period.

Allocation of sequenced regions (or search tasks) can be achieved by a simple allocation method which sequentially allocates tasks to resources. The method allocates the next task in the list to the next search resource, repeating this process until search duration limits are met. Alternative approaches include the adaptation of methods used in vehicle loading applications. Possible approaches could include the following heuristic methods.

8.6.2 List Method

List search resources in decreasing order of their unallocated search hours and allocate the next task in the sequence to the first resource in the list. If this resource is unable to search the task allocate it to the first resource in the list which is able to accomplish the task, if one exists, otherwise allocate the next task in the list which can be searched. By allocating the next search region in the priority sequence to the resource with the greatest unallocated search hours the region is searched sooner. Such an allocation scheme ensures that high priority regions are allocated across the available resources and searched before those of lower priority.

8.6.3 Reshuffle Method

This method follows the list approach but utilizes a reshuffle move amongst the list of search resources to determine which resource will be allocated the next search task. The reshuffle of the resource list may result in the task being allocated to the next resource in the list, or the next but one, *etc.* Such a move can occur at different stages of the allocation process.

8.6.4 Look-ahead Method

The 'look-ahead' heuristic seeks to find a set of tasks from the sequenced list that exactly, or as closely as possible, completes the available search hours of a given resource. The remaining resources are then allocated the remaining search tasks in a similar manner.

In the SAR environment this heuristic would need to ensure that priority tasks at the beginning of the sequence are not sacrificed in preference for those occurring near the end of the sequence that meet the look-ahead requirements. To ensure this a limit can be defined as a position in the sequence list which the look ahead procedure cannot look beyond. Such a limit can be defined as the furthest position, x, in the sequence where the following condition holds:

$$\sum_{i=0}^{x} q_i \leq \text{ total resource hours available}$$

where q_i represents the search duration of task i. Specifically, an allocation method can be described which allocates search tasks to resources, one resource at a time, by looking through the task sequence, up until the duration limit, for the task of greatest duration that can be searched by the resource. This process is continued until the duration of all remaining tasks in the sequence exceeds the unallocated search hours of that resource. Reorder the tasks of each individual resource such that the tasks appear in the same order as they did in the initial sequence. This preserves the original priority ordering.

Improvement routines which enable single or aggregate work units of equal size to be exchanged between resources, could then be employed to seek improvements in the value of the objective function.

8.6.5 Access Component of Search Duration

The search duration measure q_i needs to be fixed in advance for these allocation approaches even though the time needed to access each region cannot be determined until the preceding task in the resource's path is fixed. One way in which an approximate access time component can be incorporated within the duration measure of each search task is to define a measure of relative 'closeness' of a region to all other regions. Such a closeness measure can be calculated as:

$$\sum_{j=1, j\neq i}^{n} \frac{c_{ij} + c_{ji}}{2(n-1)}$$

where n= the total number of search regions and c_{ij} = the shortest time path from region i to region j. The measure assumes that a path exists between each pair of regions in the search region graph, either via a single edge or a sequence of edges. The set of regions that the measure is calculated over is constrained to the set of regions initially sequenced, *i.e.,* if only those regions in the primary search area are sequenced for searching then the closeness measure of a region i is only calculated for all paths from i to the regions comprising the primary search area.

Another approach to incorporating the access time of search regions in this allocation phase is to express q_i solely in terms of search hours and allow for access of these regions by further limiting the available search hours of each resource. For example, only 80% of a resource's available search hours could be allocated with respect to the search duration of the tasks, with the remaining 20% of time being set aside for accessing these regions. A routing phase can be conducted successively to the allocation phase to incorporate the actual access paths between the regions allocated to a search resource.

An alternative approach is to execute the allocation and routing of search resources simultaneously. In this approach, a search task is allocated to a resource and then the shortest access times to all regions remaining to be allocated is calculated before the next region is allocated to the resource. This may result in a different sequence of search tasks if the task ordering criterion includes an access time component. While this approach is computationally more demanding, it is the approach which appears to most accurately incorporate both access and search time to ensure that the available search hours of each resource are not violated and are filled to greatest capacity. It is also the approach that will be utilized within the search path construction methods developed.

8.6.6 Search Component of Search Duration

The choice of entry and exit vertices for the search of a triangular region can be selected independently of the search tasks either side of it (chronologically) in the search path, in order to select the easiest gradient for the search resources. Or, alternatively, the selection can be made in consideration to the path accessing that region from the previous search task, and the path accessing the following search task (where these tasks exist), to minimize the time needed to complete the entire search path. It is this form of local optimization which we employ to address the problem of how best to orient a triangular region when inserting it into a search resource's path, with the model selecting the 'best' entry and exit vertices from all possible vertex combinations.

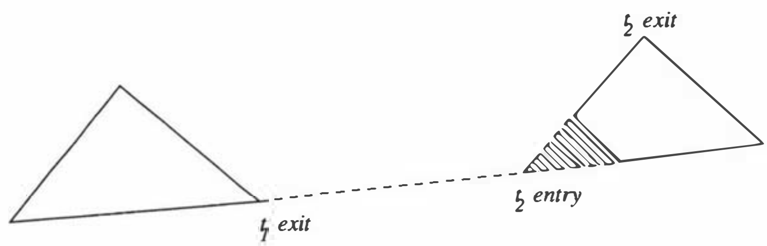

Figure 8.22: Local optimization procedure to select the orientation of a region search when the region is added to the end of an existing path.

8.6.7 Tree Sequencing of Triangular Regions

One way of sequencing, or spatially ordering, triangular regions is an approach which draws upon the relationship between the search region graph and its dual — the Voronoi diagram — and the fact that a binary tree lies embedded within it. The dual of the search region graph has a vertex positioned at the centre of each triangular region. When these vertices are connected to their adjacent vertices, through the edges of the triangular regions, a binary tree is created as illustrated in Figure 8.23. No unique tree exists as regions may 'share' two possible 'parents' as the tree is traversed.

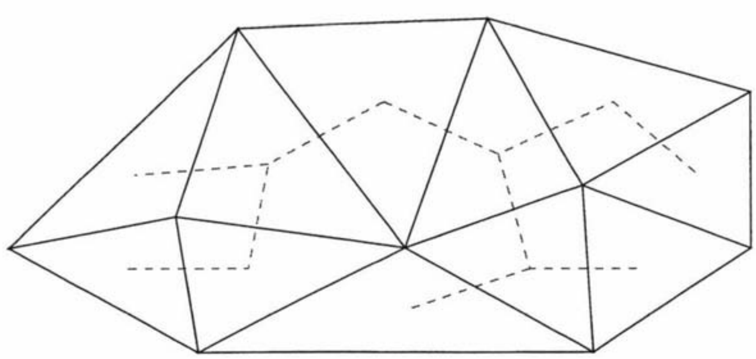

Figure 8.23: Binary tree through a TIN.

It would seem plausible that in many instances areas of similar priority ranking will be spatially close to each other. Hence routing search resources by allocating paths to each resource, which have been found by traversing the underlying tree, can take advantage of these factors. One method that presents itself, is one that establishes the root of the tree in the region with the highest priority for searching. A tree can then be constructed which branches in the direction of regions of high probability from this point. Time restrictions could be incorporated by constraining the length of tree branches, where the length is defined by the total search time required. Boundary restrictions could also be included by preventing branches forming across impassable edges.

More than one tree could be created by selecting multiple roots in consideration to the dispersity of regions of high priority and the number and location of search resources available (as one path per resource is desired).

8.7 Heuristics

Derived from the Greek word "heuriskein", meaning "to discover", the word heuristic is defined nowadays as both an adjective and a noun. Within this thesis we will use the term heuristic in both these capacities following the Oxford dictionary [2, page 553] definitions of the adjective as "allowing or assisting to discover" and "computing proceeding to a solution by trial and error". The noun definition which we will utilize is that of "a heuristic process or method." In particular we follow Foulds' [53, page 929] definition of a process or method "which, on the basis of experiment or judgement, seems likely to yield a good solution to a problem but which cannot be guaranteed to produce an optimum."

Heuristic approaches to problem solution have in the past been criticized for their lack of mathematical rigour, utilizing common sense rules and approximations, with no guarantee that the resulting solution will be even close to optimal, and without the ability to defend claims as to its respective performance. However, since the evolution of computational complexity results, and in particular, the discovery of the classes of NP-complete and NP-hard problems, heuristic methods have become widely accepted as effective approaches to gaining good quality solutions in instances where optimal algorithms are intractable or impractical. Exact procedures may be impractical to implement when they: are computationally prohibitive, require data that is unavailable or inaccurate, are too complicated for users to implement or understand, and fall outside the capacity of available resources. In particular, many real-world applications which require solutions in real-time which can cope with changing scenarios in uncertain environments, require efficient and timely solution approaches. Heuristics in their intuitive approach, exploiting the structure and special properties of the problem at hand, are well suited to these needs.

Reeves [142] states that:

> "heuristics are usually rather more flexible and are capable of coping with more complicated (and more realistic) objective functions and/or constraints than exact algorithms ... thus it may be possible to model the real world problem rather more accurately than is possible if an exact algorithm is used."

Lee [105] considers that the faster response time of heuristic methods "is often significant enough to compensate for the lost guarantee (of optimality)".

Heuristics are generally classified as falling within one or more of the following categories [53, 148]:

Constructive method The constructive heuristic method builds up a feasible solution component by component, applying rules at each decision step to determine which elements to include in the solution. Usually the rules involve a look-ahead or intuitive decision process, or may look to improving the objective by the greatest amount, as in "greedy" heuristics.

Decomposition method The decomposition method first decomposes the problem into subproblems which are solved independently, with their results compiled to form the solution set for the entire problem. This is a useful approach for large and complex problems, especially those with naturally occurring divisions such as time periods.

Learning method In the learning approach latter decisions in the method are made based upon the outcome of previous decisions.

Feature extraction method The feature extraction method, also known as a reduction method, extracts those features of known solutions which appear in a number of problem solutions. The heuristic uses these key properties to form future solutions, thus simplifying the solution of these problems.

Improvement method Improvement methods take a known feasible solution and attempt to improve it by modifying the solution's components by techniques such as element exchange. Not all immediate moves are required to be improving, as can be seen in such approaches as the Tabu search metaheuristic.

Model manipulation method Model manipulation methods attempt to find a solution to the problem at hand by modifying an aspect of the problem model, to ease solution. The solution to this modified model is used as the solution for the original problem. Common approaches include considering a non-linear objective function as linear, and relaxing specific problem constraints.

Inductive methods Inductive methods use properties apparent in the solutions of a simplified version of the complex, or larger problem, to guide heuristic development for the solution of the larger problem, by generalization.

Foulds [53, page 934] considers heuristics to perform well if they "are fast, simple and are capable of producing high quality solutions most of the time." In addition to speed, simplicity and good performance, a "good" heuristic will also meet many of the following criteria [8, 173]:

- require reasonable memory requirements;
- contain good stopping criteria;
- robust;
- accurate;
- accept multiple starting points;
- provide multiple solutions;
- provide statistical performance guides and estimation;
- flexible;
- generalizable to a range of problems;
- provide an interactive ability with the analyst and decision-maker;
- innovative; and
- insightful to particular problems or heuristic design.

Heuristics are often created to be problem dependent and the best heuristic to choose for a particular problem is often not clear cut [142] — the decision depending on a number of considerations, for example, the problem structure and size.

The type of heuristic methods which we develop below are construction methods which are performed upon the sub-problem of resource allocation over a given search period. In this respect the solution approach is also a decomposition method. Learning is also incorporated by basing allocation decisions on the subject probability distribution that has been updated with information gathered in previous searching.

As the SAR problem is dynamic in nature, heuristic approaches to determining a solution need to be more short-term to medium-term in focus, with respect to time. This is due to the changing nature of the subject's location probability distribution as new information is received and processed. It is imperative to send available resources out immediately, rather than sending out some resources and retaining others for future deployment, as time is critical in finding the lost subject. Heuristic methods, therefore, must quickly obtain a solution for the current planning horizon. In planning for the current time horizon only, the time required to generate assignments beyond this period is not made redundant when new allocations supersede those planned. The time horizon initially selected is that of 6 hours or half a shift. This time length is considered to be workable in terms of the nature of shifts and recommended work rates, but is not too short a period as to result in task allocations being continually altered before any significant search effort has been achieved.

Heuristic methods which construct search paths for the available search resources are now developed.

8.8 Path Generation

The POD level at which search regions are to be searched at directly influences the duration time of that search. Two ways in which POD levels can be determined are now detailed.

8.8.0.1 Fixed POD Levels

The first approach is to set a target POD level to which each region will be searched. This target can remain static throughout the search operation or it can vary between search periods. If the target POD level is set at a high level, the time required to search regions increases. To ensure that regions which require search times exceeding the available search resource hours (when searched at this POD level) are not automatically excluded from consideration, the maximum POD level at which such a region can be searched within the hours available is calculated. The search task is then evaluated for inclusion in search assignments at this maximum POD level under the current search criterion. Hence, for any fixed POD allocation procedure the actual POD level at which a region is assigned to be searched at is given by:

$$min\{\text{fixed POD level, maximum POD level}\}$$

The maximum POD levels are rounded down to the nearest 5 or 10% increment.

The maximum POD levels at which regions can be searched at are recalculated when the number of available search hours is reduced. This will occur when some tasks have already been allocated to a search resource and regions are sought which will complete the resource's allocation.

The need to determine the maximum POD level achievable within a given time is only applicable to triangular search regions as the time taken to complete the search of an edge region is not a function of search thoroughness; the search of an edge region always returns a POD level of 1.0.

8.8.1 Variable POD Levels

An iterative heuristic method to determine the ideal POD level to assign to a search task is developed based on the Probable Success Rate (PSR). The allocation of POD levels to search tasks occurs in conjunction with task allocation and routing. For a given search region the search speed is calculated from the search direction resulting in the quickest execution of the search task. The sweep width, W, is set for a particular region in a given set of weather and light conditions. The area of the region is fixed and the POA value decreases with search without detection.

The heuristic orders search regions in decreasing order of PSR and allocates regions for searching based on this ordering. The motivation of the heuristic is similar to the adaptation of Charnes and Cook's algorithm (presented by Stone [152]) for the exponential detection function. The method incorporates priority thresholding, where searching is conducted in one region until it becomes a higher priority to extend search effort into another search region. The first region in the ordered PSR list is searched to a POD level which results in an updated PSR value for that region equal to the PSR level of the next region in the list, at which time the POA value of that region is updated for non-detection. If a resource can be allocated further tasks, then the next region in the list is added to the path and the POD levels applied to searching both allocated regions is that which results in an updated PSR level equal to that of the next region on the list. This process continues until the total hours required to search the regions on the path and to access these regions first exceeds the available resource hours. The actual search hours spent in each region are then decreased in proportion to their contribution to total search hours, such that available effort is not violated. Essentially the heuristic allocates effort to regions in proportion to their contribution to the growth of POS.

Unlike the algorithm featured in Stone [152], access times between regions are also considered in order to realistically address the total number of hours required to search the sequence of regions. For simplicity, POD levels are rounded down to the nearest 5%. If a POD level falls below a stated minimum the procedure is halted and the POD levels of the previous regions are increased to meet the available search hours.

A heuristic which assigns search tasks to k resources simultaneously can also be developed where the k top ranked regions (ordered with respect to PSR) are initially allocated to each of the k resources. These regions are allocated POD levels which are equal to either the maximum POD level attainable in the available hours, or a POD level which results in an updated PSR value equal to the PSR value of region $k+1$ in the ordered list. Region $k+1$ is then added to the path of the resource which has the greatest remaining resource hours still unallocated. The POD level selected for this region is that which leads to an updated PSR value equal to the region next on the list. All other allocated regions have their POD levels similarly updated. This process continues until all resources have met or exceeded their available search hours and POD levels of all regions are adjusted to ensure that no resource violates their search hour allocation.

Instead of allocating the next region on the list to the resource having the smallest number of allocated resource hours, other strategies could be followed, including:

- allocating the next region in the list to the resource which is able to access it the most quickly; and
- allocating the next region in the list to the resource with the greatest number of allocated search hours which is able to search the next region within their total available hours.

Algorithm 8.1 describes the allocation of search regions via this heuristic to a single search resource.

By means of an example a small problem is now considered which consists of four search regions, 6 hours of available resource hours, and a single resource. The data for this problem is contained in Table 8.1. It is assumed in this example that all regions are adjacent to one another so that no time need be set aside for accessing the next region in the path. The regions are numbered in decreasing order of their PSR measure so the first region selected for allocation is region 1. It is assumed that searching is via the critical separation method with parallel sweep searches.

Table 8.1: An example problem for the PSR allocation heuristic.

Region	POA	Area	W(km)	Speed(km/hr)	PSR
1	0.2	0.10	0.010	2	0.040
2	0.1	0.25	0.015	3	0.018
3	0.6	0.25	0.005	1	0.012
4	0.1	0.25	0.010	2	0.008

The POD level at which region 1 will be searched to is that which results in the PSR value equal to the PSR value of region 2 (the second highest ranking region), when

Algorithm 8.1 function *PSR_heuristic*(k)

$hours \leftarrow 0.0$
for $i = 0$ **to** $nregions$ **do**
$PSR_i \leftarrow \frac{W_i \times speed_i \times POA_i}{area_i}$
order regions in decreasing order of PSR
// position 0 in ordered list is region with greatest PSR
$pstn \leftarrow 0$
$rgn \leftarrow$ region at $pstn$ in list
$next \leftarrow$ region at $pstn + 1$ in list
$POA^*_{rgn} \leftarrow \frac{PSR_{next} \times area_{rgn}}{W_{rgn} \times speed_{rgn}}$
$POD_{rgn} \leftarrow \frac{POA_{rgn} - POA^*_{rgn}}{POA_{rgn}(1 - POA^*_{rgn})}$
$hours \leftarrow hours +$ time to search rgn at POD_{rgn}
if ($hours > limit_k$) **then**
allocate rgn to resource k at the maximum POD level with
search hours $< limit_k$
else
    allocate region rgn to resource k at POD_{rgn}
    $stop \leftarrow 0$
    while ($stop \neq 1$) **do**
        $pstn \leftarrow pstn + 1$
        $next \leftarrow$ region at $pstn + 1$ in list
        $hours \leftarrow 0.0$
        for $j = 0$ **to** $pstn$ **do**
            $rgn \leftarrow$ region at position j in list
            if ($POD_{rgn} < 1.0$) **then**
                $POA^*_{rgn} \leftarrow \frac{PSR_{next} \times area_{rgn}}{W_{rgn} \times speed_j}$
                $POD_{rgn} \leftarrow \frac{POA_{rgn} - POA^*_{rgn}}{POA_{rgn}(1 - POA^*_{rgn})}$
                if ($j = pstn$ **and** $POD_{rgn} < MINPOD$) **then**
                // cannot search rgn
                else
                $hours \leftarrow hours +$ time to search rgn at POD_{rgn} + access time
                between adjacent regions
            end
        end
        if ($hours > limit_k$) **then**
            $stop \leftarrow 1$
            for $j = 0$ **to** $pstn$ **do**
                reduce current search hours in proportion to total contribution
                until $hours = limit_k$
                adjust POD_j to allow for decreased search hours
            end
        end
    end
end

end

the POA of region 1 is updated for non-detection. Utilizing the equation for PSR, the updated POA value of region 1 can be calculated as:

$$PSR = \frac{W \times speed \times POA}{area}$$

i.e.,

$$0.018 = \frac{0.01 \times 2 \times POA_1^*}{0.1}$$

i.e.,

$$POA_1^* = 0.09$$

The POD level required to achieve an updated POA of 0.09 can then be calculated. Bayes' formula for updating POA values can be re-arranged to solve for POD, giving:

$$POD_1 = \frac{POA_1 - POA_1^*}{POA_1(1 - POA_1^*)}$$

i.e.,

$$POD_1 = \frac{0.2 - 0.09}{0.2(1 - 0.09)}$$

i.e.,

$$POD_1 = 0.60$$

The number of search hours required to search region 1 to 60% POD (coverage of 0.625) is 3.125 hours, which is less than the 6 available search hours. Hence region 1 is allocated to the resource for searching at a POD level of 60%.

Next region 2 is allocated for searching. In order to do this we seek to find the POD level to search region 2 at which will result in an updated PSR value equal to that of region 3. We find that an updated POA value of 0.067 is required, which correlates to a POD level of 35%. The time needed to search region 2 to a POD level of 35% is 2.137 hours.

The next step in the heuristic is to further increase the POD level allocated to region 1 so that its updated PSR value also equals that of region 3. Increasing the POD level to 74% achieves this. The additional level of POD results in a total search hour requirement for region 1 of 4.807 hours.

The total hours required to search regions 1 and 2 is now equal to 6.944 hours which exceeds the available search hours. This means that only regions 1 and 2 are searched this period. The POD levels are then reduced, in proportion, to meet the six hours of searching available. Region 1 is searched for 4.15 hours at a POD level of 70%, while region 2 is searched for 1.85 hours at a POD level of 25%.

If access times between regions are non-zero then this time must also be included and subtracted from the available resource hours. For example, if it takes 20 minutes to move between region 1 and region 2, then the total hours available to search regions 1 and 2 reduces from six hours to 5 hours and 40 minutes.

Two generic path construction heuristics are now developed; one to generate a search path for a single resource, and one to generate search paths for a set of resources. The heuristics are governed by a number of parameters which generate different path structures and which utilize different resource allocation heuristics. Many of the components of the two heuristics are identical, including; the initialization of a set of candidate search regions to allocate to resources; the ranking of these regions with respect to a given criterion; the insertion of a selected region into the existing path; and the update of planning variables to allow for non-detection of the subject.

8.8.2 Heuristic Search Method Parameters

The following parameters are defined to generate different search path structures.

PRIMARY_AREA Whether a primary search area is initially established and whether this is constructed only at the commencement of the first search period or at the commencement of all search periods.

CANDIDATE_REGION1 The subset of search regions considered as candidates for allocation to search resources in the first search period. This subset will consist of one of the following:

- edge search regions only;
- triangular search regions only;
- edge and triangular search regions; and
- those search regions not yet searched.

CANDIDATE_REGION2 The subset of search regions considered as candidates for search allocation in subsequent search periods.

MULTIPLE A Boolean parameter indicating whether a search region can be searched more than once in the same search period.

TIME_LIMIT The maximum time duration of a search path that can be allocated to a resource with the possibility of an additional **TMORE** hours being allocated if the Boolean parameter **STRICT** equals FALSE.

PATH_FRACTION The fraction of the path **TIME_LIMIT** that is allocated as the actual limit on time duration.

MINPOD The minimum POD level of any search assignment.

PODLIMIT The maximum POD level of any search assignment.

STARTPOD The target POD level of search task allocations for the first search period. This is then incremented by **POD_INCREMENT** to set the target POD level for each subsequent search period.

RESOURCE_CRITERION The criterion on which resources are ranked for selection for task allocation:

- *greatest_hours*: greatest number of remaining available search hours;
- *least_hours*: least number of remaining available search hours (this criterion allocates paths to one resource at a time and will give the same result as $ALLOCATION = one\_resource$);
- *criterion*: closest to the region with the highest value of the **COMPARISON** criterion;[6] and
- *index*: index order.

ALLOCATION The method of search path allocation to resources, one of:

- *one_resource*: A resource is allocated its entire search path before the next resource is assigned its path.
- *parallel*: Search paths are allocated to all resources at the same time, with **RESOURCE_CRITERION** determining which resource will be allocated the next task in their search path.

ALLOCATE_TASK The method of allocating search tasks to search resources, one of:

- *single_task*: Only one search task is allocated to each resource at a time.
- *path_limit*: Search tasks are allocated to each resource until the time length of the resource's path would exceed the defined path limit.

CONNECTION The way that a selected region is added to the path of the search resource, one of:

- *path_end*: The search region is added to the end of a resource's path via the shortest edge path.

[6] Only used when the **COMPARISON** criterion contains no access component.

- *cheapest_insertion*: The search region is inserted into the resource's existing path such that the total increase in path length is the smallest over all possible insertions.

COMPARISON The criterion for region selection when constructing search paths. The following criteria are considered:

- search priority value;
- POS;
- PSR;
- search hours;
- distance from resource;
- combined search and access hours;
- elevation; and
- degree of exit vertex.

The elevation of a region is used as an allocation criterion with the purpose of generating 'down hill' paths that should be both quicker and more energy-conserving for the resources. The degree of the exit vertex dictates the number of possible access paths available when continuing on to other regions, as well as defining the number of adjacent regions, thus providing wider choice in assignment options.

The search priority measure differs from that proposed by Colwell [26] in that access times are not calculated from a common base but from the resource's position at each point of the path being constructed, and exit times are only incorporated if a region is to be inserted into a resource's path, rather than being added to the end of the path, *i.e.*, **CONNECTION** = *cheapest_insertion*. We refer to the POS, PSR and search priority criteria as 'priority based criteria' as they contain a predictive measure of successful subject detection.

COMPARISON2 The criterion for the selection of regions comprising the second portion of a resource's path, when a hybrid path generation method is utilized. The criteria set is identical to that of **COMPARISON**.

HYBRID A hybrid search path is created by using the **COMPARISON** criterion to select the regions comprising the beginning of the path and the **COMPARISON2** criterion to create the second portion of the path. The first portion of the path is fixed as one of:

- $first\_task$: The first task.
- $first\_half$: The search tasks filling the first half of the path time limit.

This approach is similar to the hybrid criterion developed by Golden *et. al.* [72], to solve the CCPP.

8.8.3 Selection of Candidate Search Regions

Initially a set of candidate search regions is defined from which search regions are then selected for allocation to search resources. The first step in this process is to define a primary search area if the parameter $PRIMARY\_AREA$ is initialized. The second step is to select the subset of search regions which are viable allocation options from those specified by $CANDIDATE\_REGION1$ or $CANDIDATE\_REGION2$. If a primary search area is initialized then only regions falling within this region are candidates for allocation. In addition, any regions that will flood within this period are not considered as candidate regions to be assigned to resources for either searching or access. Any regions which have a $\mathrm{POD_{cum}}$ value equal to 1.0 are also not considered as search candidates, because the stationary subject would have been detected if located within that region.

The procedure for defining a primary search area is now described.

8.8.3.1 Primary Search Area

A primary search area can be established at the commencement of the search operation and then re-established at the beginning of each successive search period to account for changing information. The simulation parameter $PRIMARY\_AREA$ controls if and when this establishment occurs.

The primary search area is created from search regions which have the highest probability of containing the subject and which can be searched to the target POD level, within the time frame of one search period, given the available search resources. In particular we develop a heuristic 'packing' procedure that attempts to 'pack' as many of the highest ranking POA regions into this primary search area.

The time needed to search a region is calculated as the average search time, over all entry and exit vertex combinations, required to achieve the set POD level of the current search period (or the maximum POD level that can be achieved within $PERIOD$ hours of searching if this target cannot be met).

Rather than adding search regions to the primary search area until the total resource hours available are met, based solely on this average search time, an access time component is also incorporated. In particular we assume that the fraction, $FACTOR$, of the available hours are exhausted in accessing search regions. Hence we add regions to the primary search area until the addition of the next region will cause the sum of their search times to exceed

$$FACTOR \times TIME\_LIMIT \times \text{number of search resources}$$

If the region with the next highest POA value is currently flooded, it is included in the primary search area, but, as the region will not be assigned as a search task in the current period, its search time is not incorporated into the sum of region search times. Additionally, if the region with the next highest POA value cannot be searched at a POD level exceeding *MINPOD*, the next ranking region is still considered for inclusion in the primary search area.

Once the packing of regions is complete, the convex hull of all regions in the primary search area is determined. Any search regions which do not fall in the subset of regions comprising the primary search area, but whose co-ordinates fall completely inside this convex hull, are also included within the primary search area. This does not, however, include untracked edge regions.

The formation of the primary search area is described in Algorithm 8.2 and depicted in Figure 8.24.

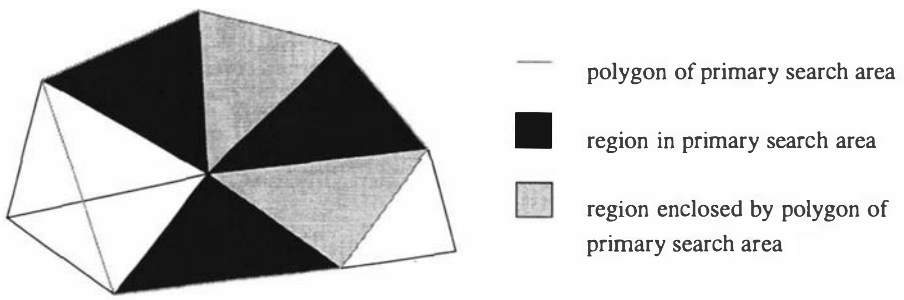

Figure 8.24: Creation of a primary search area.

8.8.4 Allocation of Search Regions

The initial time limit of a search path is governed by the parameters *TIME_LIMIT*, *TMORE*, *STRICT* and *PATH_FRACTION*. Together these parameters allow for either a hard or soft path constraint. They also allow for only a fraction of a resource's available search hours to be utilized in an initial assignment, if this is desirable for flexibility in regards to later task allocations. This may be a particularly useful approach for those problem instances displaying high levels of dynamism. The initial time limit calculated is the maximum time limit possible. In instances where a search resource has

Algorithm 8.2 function *primary_search_area*()

```
rgnct ← 0
order search regions in decreasing order of POA
maxtime ← FACTOR × TIME_LIMIT × number of search resources
totalhr ← 0.0
while (totalhr < maxtime and  unallocated regions remaining with POA > 0.0) do
     rgn  ← next region in ordered POA list
     if rgn is flooded then
          hr ← 0.0
          flood ← 1
     end
     if (POD_rgn > MINPOD) then
          hr ← average search time for unflooded rgn
          totalhr ← totalhr + hr
          if (totalhr < maxtime or flood = 1) then
               // add region to primary search area
               primary_area_rgnct ← rgn
               rgnct ← rgnct + 1
          end
     end
end
create a convex hull from vertices of regions in primary_area
for i = 0 to nregions do
     if rgn not in primary_area but lying completely inside convex hull
      and not an untracked region then
          primary_area_rgnct ← i
          rgnct ← rgnct + 1
     end
end

end
```

already spent some time searching, or when the number of search hours remaining in the period are less than this limit, the path limit is reduced accordingly.

The actual search method to be utilized over each region in the path is stored as the path is generated. A set of planning variables is also initialized to store POD and predicted POS statistics for each search region as the search paths are generated. These predicted POS values are later input into the objective function. Once a region is fixed in a resource's path, its POA and POD_{cum} values are temporarily updated for planning purposes to account for non-detection, which in turn impacts upon the predicted POS values. Actual POA and POD_{cum} updates are, however, only updated after the search of a region has been successfully completed. If, at the commencement of a search period, some resources are still completing the search of regions assigned to them in the previous period, then the POA and POD_{cum} values of these regions are temporarily updated for non-detection before any further path generation occurs.

The POD level at which a region i is to be searched is calculated as:

$$min\{phasepod, maxpod_i\}$$

where $phasepod$ = the target POD level for the search period and $maxpod_i$ = the maximum POD level to which region i can be searched to. The only exception to this are edge regions which are allocated a POD level equal to 1.0 regardless of the value of *phasepod*. The variable *phasepod* is set to $STARTPOD$ in period one and then incremented by $POD\_INCREMENT$ in each successive period. The maximum POD level is rounded down to the nearest 5%.

At the outset of path generation, regions are ranked with respect to the criterion defined by *COMPARISON*. Regions are then re-ordered before each task allocation. The ranking procedure redetermines the maximum POD level at which each region can now be searched, given the number of allocation hours remaining and the time needed to access the region from the position of insertion in the resource's path. If this maximum POD level falls below the specified parameter $MINPOD$ then the region is discounted as a candidate for allocation. The time required to access and search the region is calculated as the least total time over all entry and exit vertex combinations yielding a POD level greater than $MINPOD$.

The highest ranked region is inserted into the resource's path at the position dictated by the parameter $CONNECTION$. If $MULTIPLE$ is set to TRUE, such that a region may be searched more than once in a single search period, we assume that it is re-searched after the initial search has been completed, *i.e.*, we assume that there is no simultaneous searching of any region by more than one search resource. This assumption dictates the first possible position in a resource's path at which a region being re-searched can be inserted, irrespective of $CONNECTION$. If this position is beyond the maximum path position which can currently be assigned, the next ranked region is allocated instead.

When $CONNECTION$ = *cheapest_insertion* the position at which the region is inserted is selected from all viable positions as the one resulting in the least over-all time increase. When a region is inserted at a position in the path adjacent to access-only edges, these edges are deleted if no longer required, as depicted in Figure 8.25. In this Figure, edge region i is to be inserted as the second edge region in the path after vertex a. As the edge regions adjacent to vertex a are currently only being utilized as access from vertex a to vertex b, these edges can be deleted and replaced by the shortest edge path connection between vertices a and x, and y and b.

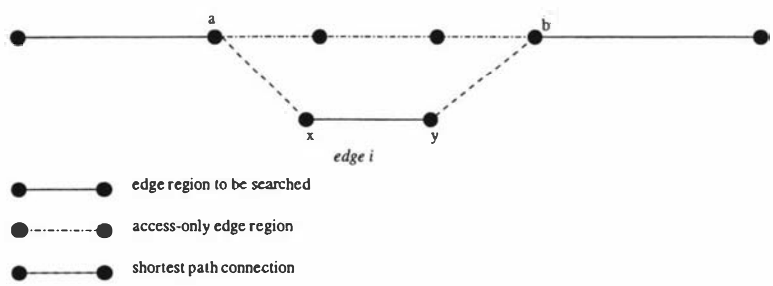

Figure 8.25: The cheapest insertion of a region into a search path.

If an edge region is to be allocated to a resource for searching but this region is already temporarily assigned to be used as access to a region by a resource or resources, then the first resource to encounter this edge is designated to search it. (As the time to search an edge and to use it only for access purposes is deemed to be the same in the simulation model this will not cause a resource to exceed their search hour limits.)

These components of path generation combine to facilitate generic search path construction. In particular Algorithm 8.3 generates a generic search path for a single search resource and Algorithm 8.4 generates generic search paths for a set of resources.

It may arise that a viable path cannot be constructed from the set of parameters specified, for the number of allocation hours and the set of candidate regions available. In these instances Algorithm 8.5 is utilized to assign the resource a region to search adjacent to their current position. The region selected is ideally the region returning the highest valued $COMPARISON$ criterion value. If all adjacent regions have been searched or allocated for searching, and multiple searching of a region is not considered, the parameter $MULTIPLE$ is over-ridden allowing re-searching of the region with the highest ranking to occur. When no adjacent regions can be searched above the $MINPOD$ level, this criterion is also over-ruled to extend the candidate region set to incorporate

Algorithm 8.3 function *generate_path*($resource, cost$)

```
limit ← min{remaining useful search hours, remaining search time this period}
path_resource,0 ← start_resource
order candidate tasks with respect to COMPARISON criterion
while ((ALLOCATE_TASK = one_task and no tasks allocated)
 or (ALLOCATE_TASK = path_limit and resource not at limit)
 and tasks remaining < limit) do
    rgn ← select next candidate task in ordered list with total path
    increase < limit which will not result in simultaneous searching
    if rgn is an edge and already allocated to resource for access only then
    change status of edge from access to search
    if (CONNECTION = path_end) then
    rgn is added to the end of the resource's path via the shortest edge path
    connection
    else
    rgn is inserted into the resource's path to increase the total path length the
    least and after any previously allocated searches of rgn

    update planning POA values for non-detection
    update PODcum variables
    update limit
    reorder candidate tasks with respect to COMPARISON criterion
end
if no candidate tasks are able to be searched within limit then
alternative_path(resource, limit, cost)
end
```

regions which can only be searched below the $MINPOD$ level. Of these, the region allocated to the resource for searching is the region returning the highest POS value.

8.8.4.1 Special Task Types

In addition to the general search path constructions described it is also possible to allocate special types of search tasks to resources based on those described in Chapter 7. These include the following:

A. follow the intended path of a subject past the PLS;

B. search regions adjacent to the PLS;

C. follow a TSP tour through hazards and huts located in the search area;

D. search a connected sequence of homogeneous edge regions *i.e.* stream beds, ridge lines or tracks;

E. follow a binary cut path through the intended path of the subject;

Algorithm 8.4 function *generate_kpaths*(*resources*, *ct*, *cost*)

```
rct ← 0
for k = 0 to ct do
limit_k ← min{remaining useful search hours, remaining search time this period}
if (ALLOCATION = one_resource) then
    // allocate path to one resource at a time
    while (rct < ct) do
        resource ← resource selected from RESOURCE_CRITERION
        generate_path(resource, cost)
        rct ← rct + 1
    end
end
else
    // allocate paths in parallel
    while (rct < ct) do
        resource ← resource selected from RESOURCE_CRITERION
        order candidate tasks with respect to COMPARISON criterion
        rgn ← select next candidate task in ordered list with total path
        increase< limit_resource which will not result in simultaneous searching
        if such a region exists then
            if rgn is an edge and already allocated to a resource for access then
            change status of edge from access to search
            if (CONNECTION = path_end) then
            rgn is added to the end of the resource's path via the shortest edge
            path connection
            else
            rgn is inserted into the resource's path to increase the total path
            length the least and after any previously allocated searches of rgn

            update planning POA values for non-detection
            update PODcum variables
            update limit_resource
            if (ALLOCATE_TASK = one_task) then
                limit_resource ← −1
                rct ← rct + 1
            end
        end
        else
            // no candidate tasks are able to be searched within limit_resource
            alternative_path(resource, limit_resource, cost)
            limit_resource ← −1
            rct ← rct + 1
        end
    end
end

end
```

Algorithm 8.5 function *alternative_path*(*resource*, *limit*, *cost*)

```
    ct ← 0
    oct ← 0
    for all regions i adjacent to start_resource do
        if region i is not an untracked edge and is not being searched at
        this time and POD_cum,i < 1.0 then
            hr ← best time to search region i, including access
            if hr ≤ limit and search POD > MINPOD then
                neighbour_ct ← i
                ct ← ct + 1
            end
            else
                other_oct ← i
                oct ← oct + 1
            end
        end
    end
    if (ct = 0 and oct = 0) then
        // no neighbouring regions are candidates for allocation
        return(−1)
    end
    if (ct = 0) then
        // no neighbouring regions can be searched within limit above MINPOD
        if all regions in other have been allocated for searching
         and MULTIPLE = FALSE then
        override MULTIPLE condition
        region ← region in other with greatest POS return when MINPOD
        condition is over-ruled
    end
    else if (ct ≠ 0) then
        // neighbouring regions can be searched within limit and above MINPOD level
        sort neighbour in order of COMPARISON criterion
        if MULTIPLE=FALSE then
            if all regions in neighbour have been allocated for searching then
            region ← region in neighbour with highest rank - override MULTIPLE
            criterion
            else
            region ← highest ranked region in neighbour not yet allocated for searching
        end
        else
        region ← region in neighbour with highest rank
    end
    add region to the path of resource
    limit ← limit − search and access time for region
    return(1)

end
```

F. conduct a perimeter search of the primary search area;

G. follow a hedging path or median path

In addition to these, specific redeployment paths are detailed in Chapter 11. These paths are utilized when a search resource is redeployed from their current position to another position on the search region graph and utilize shortest edge paths which cover unsearched edge regions.

8.8.5 Specific Path Constructions

Within the SAR simulation environment we consider four specific path generation methods.

8.8.5.1 Benchmark Method

The first method is used as a benchmark against which to compare simulation computational results. As there are differences in the manner in which search allocations are made between search managers and regions of the country, and indeed the world, we do not propose that this allocation method is representative of current search practice but we have instead attempted to incorporate aspects which could be viewed as typical of approaches currently used.

In particular, we stipulate that only edge regions comprise the candidate set of regions for allocation in the first search period. This is to mimic the initial reconnaissance search of linear features utilized in most instances. We also define a primary search area, as is common practice. Search regions are ranked for allocation based on Colwell's search priority method [26] and are allocated sequentially to resources via a shortest edge path connection to the end of the path. The method is trialled with each of the proposed resource criteria. Only one search of a search region is permitted within a single period and regions are allocated until the specified path limit is attained, with this being viewed as a hard constraint. The path limit selected is six hours, which is advocated as the maximum search time before human fatigue sets in.

Additionally, as it appears to be common practice to only execute a search when some minimum POD level is attainable, the arbitrary POD value of 20% is set as this minimum requirement. An arbitrary upper limit of 90% is placed on a single search of a triangular region in accordance with perceptions that repeated sweeps of lower POD are more desirable [21]. In line with Perkins' critical separation theory [137] the beginning target POD level is set to 50%, the theoretical figure obtained when resources are spaced at this distance. This is arbitrarily incremented by 5% for subsequent search periods until the $PODLIMIT$ is attained.

Benchmark Method:

PRIMARY_AREA	initiated at the beginning of each search period
*CANDIDATE_REGION*1	edge search regions only
*CANDIDATE_REGION*2	edge and triangular search regions
MULTIPLE	FALSE, no re-searching of a region within the same period
TIME_LIMIT	six hours and *STRICT*=TRUE (a hard constraint)
PATH_FRACTION	1.0
MINPOD	20%
PODLIMIT	90%
STARTPOD	50%
POD_INCREMENT	5%
ALLOCATION	*parallel*
RESOURCE_CRITERION	*greatest_hours*, *least_hours* and *index*
ALLOCATE_TASK	*path_limit*
COMPARISON	search priority value
CONNECTION	*path_end*

In order to facilitate comparisons between the path construction methods the same parameter settings for POD determination and path limits are initially used for all path methods.

8.8.5.2 Single Task Method

The second path generation method is highly myopic and, as such, responds to the inherent dynamism of the SAR problem at an implementation level. This method sequentially allocates only one search region to a resource at any one time, selecting the region with the highest valued *COMPARISON* criterion. As a resource searches only one region and then waits for new instructions, this method fully incorporates and reacts to any new information gained since the initial allocation of search assignments. At any search phase, all regions are considered candidates for searching and no primary search area is defined; regions are selected solely on the *COMPARISON* criterion. The three region criteria which contain a probability of area and detection component are each trialled for this search path method. Multiple searches of regions within a search period are permitted, *i.e.*, a search intensification strategy is followed.

8.8.5.3 Primary Search Area Method

The third search path construction method allocates for searching only those regions which fall within the primary search area. The method allocates search regions to resources one resource at a time and halts when the path limit of a resource would be exceeded by any additional assignments. The main difference in approach is that, rather

Single Task Method:

PRIMARY_AREA	not defined
*CANDIDATE_REGION*1	edge and triangular regions are candidates in all periods
*CANDIDATE_REGION*2	edge and triangular regions are candidates in all periods
MULTIPLE	TRUE
ALLOCATION	*one_resource*
ALLOCATE_TASK	*single_task*
COMPARISON	search priority value, POS and PSR

Primary Search Area Method:

PRIMARY_AREA	initiated at the beginning of each search period
*CANDIDATE_REGION*1	edge and triangular regions are candidates in all periods
*CANDIDATE_REGION*2	edge and triangular regions are candidates in all periods
MULTIPLE	FALSE
ALLOCATION	*one_resource*
ALLOCATE_TASK	*path_limit*
COMPARISON	search hours, access time, combined search and access hours, PSR, POS, search priority, elevation and degree of exit vertex
CONNECTION	*cheapest_insertion*

than adding regions to the end of the path generated so far, regions are inserted into the path at the position which will result in the least total increase in path length. As the primary search area is constructed to incorporate only those regions which can be searched (approximately) within the time frame of the search period, this method heuristically finds the quickest time to search these regions, rather than applying a strict priority ordering. In light of this, the method does not permit the re-searching of regions within a period. As the regions with highest POA values are the first to be included within the primary search area, priority is inherently incorporated within the method. All criteria for selecting the next region to insert into the path of the resource are trialled.

8.8.5.4 Path Scan Method

The fourth path generation method is the one that allows the most scope for varying combinations of region selection and allocation. The approach is parameter-dependent and creates paths by simultaneously allocating search regions to resources until their path limits would be violated by further additions. Search paths are constructed by adding regions to the end of the current path via the shortest edge path connection. The region to be added to the search path is selected based on the *COMPARISON* criterion. At each stage the resource to be allocated a search region is the resource that has the greatest number of unallocated search hours remaining. When selecting regions

Path Scan Method:

PRIMARY_AREA	not defined
*CANDIDATE_REGION*1	edge and triangular regions are candidates in all periods
*CANDIDATE_REGION*2	edge and triangular regions are candidates in all periods
MULTIPLE	FALSE
ALLOCATION	*parallel*
RESOURCE_CRITERION	*greatest_hours*
ALLOCATE_TASK	*path_limit*
HYBRID	*first_half*, when this parameter is used
COMPARISON	PSR, POS and search priority
*COMPARISON*2	search hours, access time, combined search and access time, exit vertex degree and elevation
CONNECTION	*path_end*

based on a priority criterion this approach ensures that the region is searched earlier rather than later. A region is to be searched only once within a period and all regions are considered candidates for allocation. The path scan method is similar in concept to the path-scanning algorithm developed by Golden *et. al.* [72] to solve the CCPP; the main difference is that paths are constructed in parallel rather than one path being constructed at a time. In the approach of Golden *et. al.*, paths are also generated for each criterion considered, with the final path being the best solution resulting from these.

The method introduces the concept of generating hybrid paths, constructed in two portions from two different criteria. In particular if *HYBRID* is initialized the method divides the total search hours to be allocated in half and allocates the first half of these via one criterion and the second via another. The criteria used to allocate the first portion of the path are those which maximize a measure of the probability of success. The second portion of the path is then constructed via criteria which select regions close to the resource's position or which enable as many regions to be searched in the time remaining as possible. This may be viewed as a diversification strategy; searching regions which may otherwise not be selected, due, for example, to a lower POA value. When the *HYBRID* parameter is not set, a complete path is generated via one of the priority based criteria.

This chapter has investigated: ways in which the spatial search of a region can be executed; heuristic methods to partition the TIN into individual search regions; and a number of heuristic methods of resource allocation, including a method for defining a primary search area. The four particular path generation methods defined in Section 8.8.5 have distinct characteristics and comprise the set of initial methods on which some preliminary computational experiments will be conducted in Chapter 12.

We now examine the dynamic and stochastic elements of a SAR operation in more detail, as a precursor to developing some initial dynamic, real-time search strategies.

CHAPTER 9

Dynamic Search Strategies

"Our ability to collect and use data in real time is restricted by our ability to absorb and synthesize the information being put in front of us. Real-time operations impose tight constraints on the amount of time we are allowed to make a decision. ... Despite the long history of modelling and optimizing long-range planning problems, these are often the problems that humans handle best. By contrast, humans struggle with the complex decisions that must be made in a very short period of time, and yet these tend to dominate day-to-day operational planning."

— DROR AND POWELL [44, PAGE 11]

9.1 Real-Time Decision Problems

Séguin *et.al.* [146] address ***Real-Time Decision Problems (RTDPs)*** and the ***Real-Time Decision Systems (RTDSs)*** which are developed to support or replace human decision-makers, in the real-time solution of such problems. The authors [146, page 173] state that:

> "The study of RTDPs is not yet well established in the OR community, although agencies, companies and governments are facing increasing pressure to work and to react in real time."

Séguin *et.al.* do, however, predict an expansion of this research area "in the near future".

9.1.1 Attributes of RTDPs

RTDPs are defined by Séguin *et.al.* [146, page 162] as occurring within a "dynamically evolving environment" where resource limitations exist and the information available on which decisions are to be made may often be incomplete or uncertain. The outcome of a given action may also be uncertain and hence such problems "typically include stochastic as well as dynamic components". The SAR problem is encapsulated by this description and in particular possesses the following attributes of RTDPs as identified by Séguin *et.al.*.

- An optimal desired solution.
- Multiple and conflicting objectives.
- Fast response time.
- Short and medium term planning with actions typically forecast for a 6 or 12 hour time horizon.
- Multiple (reactive, incremental, deliberative) courses of action.
- Uncertain outcomes of action — in terms of whether or not the subject or a clue will be detected.
- Input data can be certain, *e.g.*, the location of resources, and uncertain, *e.g.*, the knowledge of the subject's intentions and confusion factors resulting from the detection of clues. Data may be numerous or scarce in nature, and will generally be incomplete, especially in respect to compiling information which will lead to the discovery of the subject.
- Future events are extrapolated and predicated, although such events may be ignored, to an extent, in terms of short term planning.
- The environment and working conditions are time variant, and are also unstable to some degree.

9.2 Solution Approaches

In scenarios displaying dynamic elements, mechanisms which update information are essential, as are rules to cope with such information [141]. Rules must be included to determine the necessity of updating existent routes and to resequence, or reassign, tasks in order to achieve the best overall solution. In addition, Psaraftis [141, page 229], emphasizes the need for faster computation times to meet real time requirements, the implication of which is that "rerouting and reassignment decisions tend (by necessity) to be made in a heuristic and 'local' fashion".

Psaraftis [141, page 232] gives the generic design features of solution approaches to dynamic problems as being:

- interactive;
- user-friendly with graphical aids;
- flexible;
- hierarchically designed using a 'first-cut' analysis followed by more detail; and
- able to update routes and schedules efficiently at any time "without compromising 'key decisions' already made."

Psaraftis states that many solution approaches to dynamic problems have attempted to adapt static solution methods into a dynamic framework. This often involves significant redesign, and is often in the form of local operations that are applied after the problem has been initially solved by a static algorithm. Heuristics are often incorporated within the planning principle of a 'rolling horizon'.

Psaraftis specifically considers the dynamic vehicle routing problem and presents local insertion and k-interchange heuristics as fast approaches which have been shown to be particularly suited to such applications. Psaraftis [141, page 234] states that:

> "If the time horizon of the initial input is short or if subsequent input updates are numerous, the overall schedule would be less influenced by its initial solution and more by the subsequent local improvements. Such a scenario would drastically reduce the role of the static core in a dynamic situation, and shift the emphasis to the efficiency of the local operation method."

Dror and Powell [44, page 12,13] review the current state of stochastic and dynamic models in transportation and provide three different perspectives for solving stochastic models.

- A priori — where decisions are made "prior to knowing the outcomes of certain random parameters". This approach requires a design or strategy to be developed that will perform well over a range of outcomes.
- "Evaluation of simple operating practices in a stochastic, dynamic environment."
- "Development of operational models that directly assist decision-making in a real-time environment" and involve the use of a rolling horizon in decision-making and a possible interface with on-line databases.

The development of models which directly assist the decision-maker in solving real-time problems, is addressed in detail by Séguin *et.al.* in their description of a RTDS. The development of such a system to aid decision-makers in solving the SAR problem would be advantageous and aspects of such a system are now considered within the framework presented by the authors.

9.3 Real-Time Decision Systems

Séguin *et.al.* [146, page 163] consider that a system developed to solve a RTDP "must be able to decide by itself whether there is time to deliberate and if so, to what extent." The authors also give the following characteristics of a RTDS:

- an ability to recognize a change in the state of the world;
- an ability to focus on the key issues;
- an ability to dynamically adapt to changing scenarios;
- an ability to incorporate new information;
- an ability to cooperate with any other systems; and
- an ability to respond with a solution despite minor system failures.

In particular Séguin *et.al.* give the four basic functional components of a RTDS as:

- information management and data fusion;
- situation assessment;
- evaluation of alternatives; and
- decision — which can be classified by three levels of response characterized by increasing decision-making time,
  - reactive planning,
  - incremental planning, and
  - deliberative planning.

We proceed to illustrate how these functional components are addressed by the SAR problem.

9.3.1 Information Management and Data Fusion

Information about the current state of the world[1] is gathered via: GPS units carried by the search resources; radio communication between search resources and search management; and weather reports. GPS units provide situational information allowing the positions of search resources to be determined at any point in time. Radio communication and briefings at the search base relay information, and weather reports are used to estimate travel times and search urgency.

9.3.2 Situation Assessment

Séguin *et.al.* [146, page 164] state that "the significance of a particular event must be interpreted with respect to the current state of the world, as well as its past history and predicted future." The significance of an event will determine the level of reaction and response from the system. In the SAR problem the situation is assessed by considering:

- the relevance of any clue found (ROC value);
- the knowledge of the subject's movements (POA values);
- the past, current and forecasted weather;
- the level of search urgency;
- the value of POS_{cum}, the growth of POS_{cum} with each search phase and the elapsed duration of the search;
- the positions of the search resources and their future movements;
- the availability of search resources, considering fatigue and ability;
- the priority class of each search region and its current POD_{cum} value;
- the traversability of streams and rivers; and
- the available light — affecting visibility.

9.3.3 Evaluation of Alternatives

Alternatives for responding to situations that arise are evaluated by preconditions, with respect to feasibility. The state of the world after an alternative has been executed is described by postconditions [146].

Postconditions in the SAR problem include:

[1] The search environment and the external influences acting upon it.

- updated POA values and cumulative POD levels for each region;
- the regions which have been searched;
- the time resources have spent in the field;
- the positions of search resources; and
- ROC values and the level of overall knowledge.

Preconditions that determine the feasibility of alternatives include:

- resource availability;
- region traversability;
- fuel and capacity restrictions of helicopters or transportation vehicles;
- past, current and predicted weather;
- useful search hours for the resources (fatigue levels); and
- the number of hours of daylight remaining.

In the SAR problem, if a relevant clue is detected, alternatives that can be evaluated include reallocating search resources to regions indicated as now having a higher priority. If the relevance of the clue is unclear (as in an uncertain environment) then it may be advisable to continue with the current search assignments until additional information becomes available. If the weather deteriorates, or is predicted to deteriorate, this will also impact upon decisions. Alternatives under this scenario include withdrawing search resources or allocating more resources to concentrated areas of high priority in order to achieve coverage of these areas within the time available.

Evaluations of alternatives are based upon the increase in POS value, the rate of coverage, the time of reassignment, loss of coverage if current search tasks are abandoned and the situation assessment factors listed above.

9.3.4 Decision

> "A decision consists in either reacting to the current situation or doing nothing ... If the decision is to react, one must decide how and when to react. Accordingly, a decision does not necessarily result in an immediate action. Rather, the action is incorporated into a plan of actions that extends over a rolling time period known as the planning horizon", [146, page 164].

Decision-making in the context of the SAR problem is considered for the three levels of response detailed by Séguin *et. al.*[146]. These levels of response can also be viewed as non-compensatory (local and selective) strategies versus compensatory (holistic) strategies.

9.3.4.1 Reactive Planning

Reactive planning is "concerned with short-term actions, whose effects are local in nature. These actions may be adequate if the modifications in the state of the world are small", [146, page 164]. An example of such planning in the SAR environment is to allocate a search resource to immediately search a region in the vicinity of their current position. Such a reallocation results in a modification to the current search path but the resource continues to execute their original tasks. For such a reallocation to be achievable it is necessary to have initial search paths whose durations are significantly less than any constraining time limits. A further example of reactive planning is illustrated when a region initially allocated as a component in a search resource's path becomes impassable due to flooding, and local path modifications are needed to bypass the region.

9.3.4.2 Incremental Planning

Incremental planning involves "slightly more elaborate modifications of the current plan and is appropriate when the current state of the world does not depart too much from its expected state at the time the plan was first devised", [146, page 164]. An example of such planning in SAR is the allocation of a new search task to a search resource where this task is inserted within their current search path at some future time point.

9.3.4.3 Deliberative Planning

Deliberative planning is a "mandatory revision of the initial plan whenever the current state of the world departs significantly from its predicted state, hence making the initial plan less effective or even inapplicable", [146, page 164]. Deliberative decision-making is employed when the current strategy deteriorates to a point where any new strategy would be more effective. Indicator levels can be used to monitor a strategy and its associated deterioration. In the SAR operation a deliberative planning response may correlate to the allocation of new search paths to search resources which differ entirely, or in some part, from those initially assigned. Such planning could also involve the exchange of search tasks between search resources and/or the reordering of those tasks already assigned. Scaling-down or suspending the search operation would also fall within this category of decision-making.

Withdrawing search resources in the onset of adverse weather conditions is likely to fall under either reactive or incremental planning, depending upon the suddenness of its onslaught. We assume that there exists enough forewarning to recall search resources to the search base before adverse weather strikes, effectively categorizing this action as an incremental planning response.

Mandelbaum and Buzacott [113, page 17] state that "flexibility is required in a system or process so that it is able to respond to change in the system's environment or to change in the decision-maker's perception of reality."

Dynamic models require responsiveness to changing situations and a measure of flexibility inherent in the system to respond to change. Flexibility can be modelled as the size of remaining period actions after an initial decision has been taken [113], and as such can be achieved within the SAR operation by initially routing resources to say, 80%, of the search time limit. Incorporating this measure of flexibility within each of the search paths can allow tasks that may arise with high priority to be allocated to resources during the execution of the operation.

9.3.5 System Architecture

Séguin *et.al.* provide a framework of a general conceptual architecture for a RTDS with three levels of abstraction. The framework depicts the inter-relationships between the system components in each level of the system, with the functional components of RTDSs described above being incorporated within these system components. The three layers of the general architecture described by the authors consist of an interface layer containing an effector, a characterizer and a verbalizer. The next layer of abstraction is the choice layer consisting of a selector and a projector. The top layer is the generation layer and consists of the planner component. Figure 9.1 depicts such a framework for the SAR problem following the general architecture outlined by Séguin *et.al.*

The function of each of the components in the general architecture and their role in the decision-making process is now described. As indicated by the label of 'interface layer', the components at this level interface with the external environment — both relaying information from the environment to other components and relaying actions to the environment. In particular the characterizer detects physically measurable, significant events which occur in the environment. The verbalizer analyses information from the characterizer and effector, and assesses the current situation before triggering an appropriate action by calling either the projector or the selector. The effector effects actions from the selector by relaying implementation instructions and also executes fast reactive actions when required.

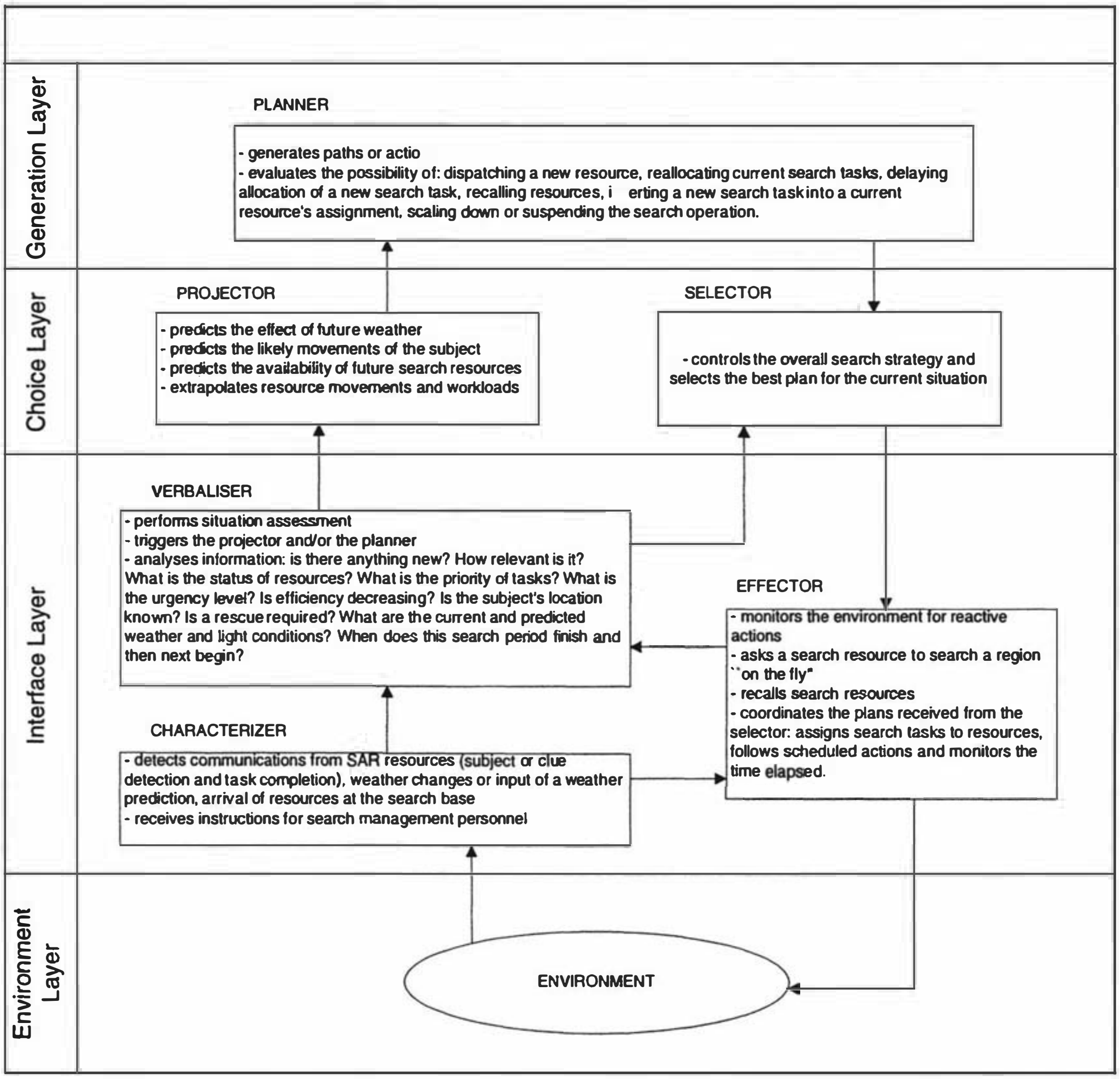

Figure 9.1: A general architecture for the SAR problem

The function of the components comprising the choice layer is to form decisions on inputs from the verbalizer. The projector gathers additional information on uncertain events by simulating, extrapolating, predicting or estimating from known information, while the selector selects the best plan for the current situation in line with the global strategy of the problem.

In the top layer — the generation layer — the planner evaluates alternative responses and generates plans or actions from the information it receives via the projector.

The time required for decision-making increases with each level of abstraction, hence when fast, reactive decisions are required the only components utilized are those within the interface layer. When the actions currently being executed satisfactorily address the present situation the components in the two lower layers — the interface and choice layer — are invoked. When the situation at hand cannot be satisfactorily met by the current actions all three layers in the architecture are required for more detailed and deliberate decision-making [146].

9.4 Relevance of Solution Approaches in the Literature

Séguin *et.al.* [146, page 165] review the Artificial Intelligence literature and its relevance in providing solution approaches for RTDPs. In particular they view the solution of an RTDP as a series of "time-constrained searches in a sequence of problem spaces." The authors examine techniques that reduce the number of states that are generated for each problem space. In reducing the number of possible states, the time needed to search through a sequence of problem spaces is reduced and becomes more predictable. Such techniques include:

- Pruning — eliminates portions of the problem space which are known to be, or have a high likelihood of being, "useless to the search process." Pruning can be either static or dynamic, depending on when such knowledge becomes available. The technique "guarantees a reduction of worst-case execution times without compromising solution quality", [146, page 166].

- Ordering — considers first the states which have the greatest likelihood of quickly reaching a terminal state; a 'best-first approach'.

- Approximating — "potentially less accurate solutions are deemed acceptable in exchange for a reduction in variability and lower worst- and average-case execution times." Approximations are used in reaching a solution *e.g.* quickly computed upper or lower bounds that are not strictly valid.

- Scoping — determines how much of the solution space will be examined given the time available. Scoping is more limited for highly unpredictable environments.

Séguin *et.al.* [146, page 166] note that "every algorithm, method or solution concept proposed for solving RTDPs makes use (to some extent)" of these methods.

Possible applications of these techniques to the SAR problem are now considered.

- Pruning:
  - eliminate solutions that assign a search resource to search two regions that are of similar priority but located at a significant distance from each other;
  - eliminate those search regions from the list of candidate regions for allocation which can not be reached by the resource within the time limit constraints.
- Ordering: first examine solutions where a search task is allocated to the nearest search resource.
- Approximating travel times when a weather change is expected a some point in the next time interval.
- Scoping: allocating search tasks to resources 6 hours in advance.

Séguin *et.al.* [146] also consider algorithm classes found within the artificial intelligence literature.

The authors outline the advantages and disadvantages of applying these algorithms to solving RTDPs and suggest that a combination of approaches be developed to balance out their individual strengths and weaknesses.

Séguin *et.al.* also examine four generic Operations Research metaheuristics: simulated annealing; Tabu search; Greedy Randomized Adaptive Search Procedures (GRASP); and genetic algorithms, examining their relevance to solving RTDPs. The authors [146, page 172] conclude that these metaheuristics "can easily be adapted to real-time decision problems", although to be competitive they must be fine-tuned and specifically designed for the problem at hand. The authors also propose that neural network approaches could be applied to RTDPs, and investigate the potential benefits and applications of parallelism in solving these real-time problems.

Some specific and dynamic strategies are now detailed to address the ever-changing nature of the SAR operation, in order to develop efficient search path allocation methods.

9.5 Dynamic Strategies

9.5.1 Strategy Components

The frequency with which strategies and search paths need to be changed gives some measure of the robustness of that strategy. Ideally we desire a solution strategy which is readily adaptable, obtainable and efficient. Strategy implementation is phased and evolves over time. The strategy is redefined and directed by the information revealed and the strategy responds to this information and the current level of urgency. The stratagems themselves are a set of tools and techniques designed to respond to differing and specific situations. Strategies can be viewed as 'blocks' which 'slot-in' at differing thresholds of activity in the world. Such thresholds or indicator levels are quantitative values or states of the system that trigger a strategy change.

The decision-making process in the SAR problem exemplifies a preference-learning phenomenon. As additional information comes to hand the decision-maker's perception of the decision problem and his preference structure alters; this is illustrated by changing POA values as the search progresses, and a changing emphasis on the scenario being followed. The problem solution can be approached sequentially such that, when an environmental change is detected which leads to new preferences, the current strategy is altered to produce a new strategy. Preferential strategies depend on spatial and temporal aspects, and the relevance of new information.

Spatial aspects of the SAR problem include:

- the subject's location probability distribution at any given time;
- the level of coverage that each search region has received;
- resource locations at a given time as predicted from the search times and travel times modelled on the TIN;
- the locations of detected clues and their relevance to the search operation; and
- accessibility and traversability of regions within the search area.

Temporal aspects of the SAR problem can be measured by the following:

- the elapsed duration of the search operation;
- the length of time that search resources have been active — indicating the efficient search time remaining for each active search resource;
- the time remaining in the search period;

- the time remaining before a change in weather or light conditions; and
- the time required for each search resource to return to the search base, or to the closest hut or sub-base.

Specific performance measures that can used to compare different dynamic strategies include the following:

- total POS gained over a given planning horizon;
- search effort utilized;
- the number of new and relevant clues discovered;
- the number of resource reallocations that were required; and
- the amount of duplication of access paths in the form of deadheading.

These measures monitor the stability and effectiveness of the different methods. Such productivity-related measures indicate the performance over differing periods and the effectiveness of redeployment under new and changing information. The progress and learning of heuristics within the model can be visually graphed via plots of solution quality against execution time, and the mapping of strategy over time with respective outcome.

In SAR applications it is likely that there will exist a large number of search regions. To prevent cognitive overload a heuristic simplification measure can be used. For example, clustering techniques can be applied to group search regions together to reduce the decision criteria needed. Clustering can be achieved using a similarity measure of closeness between the centre of gravity of each search region, weighted by POA similarity.

One solution approach is to fine-tune a first-cut analysis. This approach can be defined by the following steps:

1. sequence search regions based upon the objective of the search phase;
2. allocate search regions to search resources based upon this sequence;
3. look for improvements to these initial allocations with respect to the objective function, flexibility and hedging;
4. future responsive revision which may result in beginning over.

9.5.2 Changing Knowledge

The most critical dynamic factor in the model, in terms of creating the most dramatic change to the search operation, is changing knowledge. As the current state of knowledge is updated by new knowledge gathered, POA values are updated, changing the preference structure underlying the current search approach and task allocations. The priority of different search regions and differing scenarios may be dramatically altered by a single new piece of information, and result in a change in search strategy and direction. New information may also come in the form of strengthening the current search scenario being followed. The significance of the new information gathered will stimulate the level of reaction resulting.

Changing knowledge is generated by:

- search regions which have just been searched;
- clues that have been detected and their associated relevance; and
- interviews with observers or people who know the subject (this element and source of knowledge is not directly considered within the current framework).

Algorithm 9.1 outlines a possible dynamic response to a change in POA values as a result of increased knowledge.

The algorithm assumes that all available resources are currently active in the field. The approach addresses the change in preference values by a reactive, incremental and deliberative response depending on the relative level of change. The greater the difference from the information that was initially planned on, the more deliberated the response. Two thresholds (*poa_threshold* and *time_threshold*) are utilized within the algorithm. The *poa_threshold* measures the level of relative POA value which must be exceeded before current search tasks are interrupted in preference to searching regions now identified as having a higher priority. The *time_threshold* performs a similar function in determining whether enough time remains in the search period to warrant the allocation of new search paths to all resources when the subject location probability distribution is significantly altered, as in the instance of evidence for a change in search scenario. If not enough time remains in the period to effectively coordinate and communicate new search allocations and allow time for some searching to occur, current allocations are maintained with new allocations coming into place at the commencement of the next search period. Alternatively, the current search period could be terminated at the receipt of the information and a new period entered.

In instances where an additional allocation is made to a resource and the time duration of the resource's path subsequently exceeds the resource's useful search hours, any tasks

Algorithm 9.1 function *POA_response*()

> **if** *there is no change to the ordering of search regions* **then**
> no change to current search paths
> **else if** (*subject location probability distribution significantly altered*) **then**
>   // *deliberative response*
>   **if** (*time remaining in period* < *time_threshold*) **then**
>   **if** ($urgency \neq 1$) **then**
>   allow resources to complete their current search task
>   allocate a new search path to each resource from the exit vertex of their current task
>   **end**
>   **else**
>   interrupt current search tasks
>   allocate new search paths for each resource from their current position
>   **end**
>   **end**
>   **else**
>   allocate new search paths at the commencement of the next search period
> **end**
> **else if** (*unassigned region i exists of higher priority than currently allocated regions* **and** *time remains to search region this period*) **then**
>   // *reactive response*
>   select first and last point of possible insertion in each resource's path
>   // *so as to insert between tasks of similar priority level*
>   **if** (POA_i > *poa_threshold*) **then**
>   interrupt current region search
>   **else**
>   complete current search task
> $k \leftarrow$ resource which can search region i the soonest
>   insert region i into the path of resource k by the shortest path connection
>   **end**
>   **if** *resource's path now exceeds path_limit* **then**
>   delete search tasks at end of path which extend past *path_limit*
> **end**
> **else if** (*allocated region i exists of higher priority than unsearched regions scheduled prior to it*) **then**
>   // *incremental response - resequence region before regions of lower priority*
> $m \leftarrow$ resource initially allocated to search region i
>   **for** $k = 0$ **to** *resources* **do**
>   insertion position $\leftarrow$ first position after unsearched region of higher priority
> $k \leftarrow$ resource which can search region i the soonest
>   **if** ($k \neq m$) **then**
>   attempt to swap unsearched regions from the path of resource k to the path of resource m which will not violate the path limit of resource m
>   **if** *if path of resource k now exceeds path_limit in length* **then**
>   delete search tasks at end of path which violate *path_limit*
> **end**

end

exceeding this limit are deleted from the search path. If, alternatively, the limit constraint is a soft constraint, then these regions can still be searched, but with a lower search performance resulting.

A practical way of coordinating small changes in POA values is via a dynamic task queue. Search tasks can be initially sequenced in POS order. Any new task or tasks that increase in POS value and require immediate attention can be inserted further up the sequence. Such an instance might occur due to new information or the inability for a resource to complete a task initially allocated to it. These tasks then form a queue to be actioned immediately or re-sequenced for later allocation. Such a task queue that changes dynamically as the search progresses is similar in principle to the laddering approach used in the United Kingdom.

9.5.3 Changing Weather Conditions

Deterioration in current weather conditions is first considered. The effect of such a change will impact upon:

- the urgency level of the search;
- the traversability of some streams and rivers;
- the terrain traversal speeds;
- the utilization of helicopter or fixed wing resources;
- the safety of search resources; and
- the time required to complete scheduled search assignments.

Which of these events will occur is dependent upon specific thresholds. The following thresholds and rules are used to respond to such changes.

- If weather conditions reach level 4:
  - streams and rivers become impassable in two hours time; and
  - ground search resources are withdrawn two hours prior to this weather change (or as soon as such a change is predicted if predictions are not possible two hours out from the event).
- If weather conditions reach level 3:
  - helicopter and fixed wing resources are withdrawn.

The degree to which the urgency level will increase is a factor of the subject's medical condition, age, experience, clothing, weather conditions and the length of time that they are overdue, and is specifically addressed in Chapter 11.

Changes in the terrain environment will not only be dependent upon weather changes that have just recently occurred but will also be dependent upon the weather conditions predicted for future time periods. Specifically, predicted weather changes due to occur within a *PREDICT* hour 'look ahead' period are considered. The following rules can be used to determine responses to deteriorating weather conditions being predicted within this time frame.

- Flooding: if a prediction of weather level 4 conditions exists and a search resource is due to traverse a region after it is predicted to have flooded, the resource's search path is altered to avoid this. If the region requires searching rather than being used merely as an access route, the path is shuffled such that the region is searched prior to the time when it is due to flood. Algorithm 9.2, Algorithm 9.3 and Algorithm 9.4 detail specific responses to flooding at the three levels of response defined by Séguin *et. al.* It may also be desirable to create a 'safety net', *eg.* half an hour, around the predicted time of flooding to allow for inaccuracies in prediction and to ensure that search resources do not become stranded by weather deteriorating more quickly than expected.

- Withdrawal: if the weather conditions are due to deteriorate to level 4 then the current search paths are modified to return each resource to the search base within the next *PREDICT* hours. If this is not possible then resources can instead be routed to the closest hut or sub-base, where these are defined. Rather than utilizing the shortest path back to the search base a path can be selected which covers unsearched regions. Helicopter withdrawal of the search resources is not considered to be viable, as helicopters would have been withdrawn prior to the withdrawal of ground searchers (at weather level 3).

- Air-based resources: if a prediction of weather level 3 conditions exists and scheduled search paths cannot be completed within the *PREDICT* hour limit these paths are modified to return to the base within this time. Any uncompleted assignments can be modified for searching by ground resources. Transferral of ground search resources that are due to occur beyond the *PREDICT* hour time limit are rescheduled to be achieved in the first portion of the flight path, where this is possible, otherwise ground transportation is utilized.[2]

[2]The initial simulation model assumes that only ground search and resource transportation are available; air search and transferrals are considered as a natural extension to this model.

Algorithm 9.2 function *reactive_modify*()

> **for** *all edges* (i, j) *in current search paths which will flood* **do**
> delete edge from search path
> delete access-only edges adjacent to edge (i, j) in the path
> reconnect remaining search path segments by shortest edge paths
> **end**
>
> **end**

If weather conditions improve, the effect upon the search operation is in the following areas:

- If previous flooding has subsided, streams and rivers previously impassable are now candidates for inclusion in current search paths.
- If the weather conditions are now viable for air resources and these are available, search paths are created to utilize these. Such paths may replace scheduled ground search and transportation routes.
- The urgency level of the search is unaffected as the preceding detrimental weather may have already decreased the subject's survival chances, as in the cases of hypothermia or injury, and the subject will need to be found quickly.

The effects which changing weather and light levels have on POD are as follows:

- visibility is altered; and
- search spacing is altered to maintain the same POD levels, or alternatively, POD levels are altered to maintain the same search task time durations.

If search spacing is to be decreased, then the time duration of search tasks will be increased; hence, tasks previously assigned may not now be able to be completed within search hour limits. Similarly, if conditions improve, then further search allocations may be possible due to quicker traversal speeds.

9.5.4 Resource Availability

It is assumed within the initial simulation model that the number of resources used is constant over a planning horizon. Hence, the availability of resources affects only the initial assignments and is not a dynamic component of the problem. However, if resource availability does change dynamically, in terms of more resources arriving throughout the search period which are available to search, the unsearched portions of search paths

Algorithm 9.3 function *incremental_modify*()

```
for all edges (i,j) in current search paths at risk of flooding do
    if edge (i,j) is labelled a hazard then
    reschedule edge (i,j) to be searched prior to time of flooding
    else if (edge (i,j) only access to scheduled search region m) then
    re-schedule region m to be searched prior to time flooding to occur
    else if (edge (i,j) provides cheapest access to scheduled search region m) then
        obj1 ← objective function value when alternative, viable access to
        region m is utilized
        obj2 ← objective function value when region m is rescheduled
        prior to flooding time, access by (i,j)
        if (obj2 < obj1) then
        reschedule region m prior to time of flooding
        else
            delete edge (i,j) from access path
            delete access-only edges adjacent to edge (i,j)
            reconnect by alternative, viable shortest edge connection
        end
    end
end

end
```

Algorithm 9.4 function *deliberative_modify*()

```
for unallocated edge hazards (i,j) at risk of flooding do
    schedule for searching before time flooding to occur
    if additional resources on hand then
    create a new path for an additional resource including edge (i,j)
    else
    include edge(i,j) in the search path of a resource who can search the edge
    prior to flooding
end

end
```

already assigned to other search resources can be reallocated to the new arrivals as a form of incremental response. Alternatively, new paths can be allocated to each new resource to take into account information obtained since the commencement of the search phase. If a search resource currently active in the field becomes injured, or stranded due to rising water or other unforeseen natural occurrences, their unfinished tasks of higher importance can be redistributed to other active resources or newly arriving resources. Whenever a search resource becomes idle we look to allocate the resource new tasks or reallocate search tasks from other resources to that resource, if that resource is not fatigued.

Aircraft viability is also dynamic, but this is considered under the section pertaining to changing weather conditions.

This section has outlined a number of possible strategies to respond to the dynamism inherent in the SAR problem that could be further developed. Not all of the strategies are utilized within the initial SAR simulation model, and those which are, are expanded upon further in Chapter 11.

Chapter 10

Search And Rescue Simulation

10.1 Introduction

Séguin *et. al.,* [146, page 163] state that simulation "is not a solution approach but it may be useful to calibrate models and parameters, to test algorithms, and to determine the specification, performance or quality of solution methods and prototypes."

The technique of simulation allows a flexible model of a particular system to be developed, which can then be manipulated to provide insights into the system's operation. Important parameters can be isolated, by altering their values and relationships under a tightly controlled environment. The simulation itself is an isolated unit and cannot be affected by exogenous factors. This gives considerable power to the technique, allowing causes and effects to be identified, leading to a greater understanding of the system.

In the literature, we have found the research most closely associated, in approach, with the simulated search for a subject located on the TIN to be that of Thomas and Hulme [164]. Thomas and Hulme simulate the situation of a helicopter searching for a subject lost in "an uniformly featureless area" via Discrete Event Simulation (DES). The search area is modelled by a two-dimensional hexagonal grid and is assumed to remain constant throughout the search. Each vertex of the grid is connected to its neighbours by a uniform edge length. The authors consider three representative strategies of the subject; remaining stationary, walking randomly, and moving along an estimated bearing towards the helicopter when he detects it (remaining stationary otherwise). The bearing moved along includes an error parameter that models the accuracy to which the subject can determine the helicopter's location. The parameters of the helicopter and subject considered by the search simulation are; the time to move between neighbouring vertices, their velocity, and their radius of detection. The probability that the helicopter will detect the subject if he is within its radius of detection and the time the helicopter spends searching a vertex, are also parameters of the simulation.

Three search methods are modelled on the hexagonal grid and tested over each of the subject's search strategies. Thomas and Hulme also consider an adaptation of these methods to a multi-region approach where the search area is split into subareas with a search conducted over each. The subject is randomly positioned on the grid with the helicopter beginning at a point on the perimeter of the grid. If all vertices are searched without success a second sweep is then executed.

Thomas and Hulme [164, page 49] state that while simulation approaches allow only a limited number of strategies and patterns to be investigated, and are time intensive in analysis, they can also be advantageous. In particular,

> "when analytic results are very difficult to obtain as is the case with asymmetric search games, simulation is a very useful way of trying to identify general comparative results as well as suggesting attractive strategies to investigate in more detail."

It is with these aims in mind that we develop a simulation model for a land based SAR operation. In particular, a DES model of an operation is developed, incorporating a subset of the heuristics of the previous chapters as sub-routines. The DES approach was chosen as the state of the SAR operation changes only at discrete time points. The simulation model itself allows for pertinent solution method parameters to be altered. By controlling these parameters over a number of different scenarios and problem instances, we investigate the effects of such changes with the objectives of identifying critical factors and proposing outcomes for different organizational procedures.

10.2 Problem Description

The problem that we simulate consists of a SAR operation conducted over bush-covered terrain. We concern ourselves only with the problem of a subject who remains stationary throughout the search operation.

The operation commences after an alert has been raised for an overdue subject. A search base is established at the scene with an appointed SAR management team and search resources are selected from available personnel. Resources are assigned specific search tasks based upon a priority listing of topographic areas within the total search area. Searching commences after an initial briefing of the search resources. Any information gathered, together with the latter redeployment of search resources or the allocation of new search task assignments, are communicated between the search resources and the

SAR management team via radio. When the subject is detected he is transported out of the region. If the subject is injured, transport is via helicopter if environmental conditions are favourable, or otherwise via stretcher. If the subject is uninjured, they return to the search base with the search resource which located them. All other resources in the field return to the search base at this time. If the subject remains undetected at the end of a defined period, the operation is suspended.

The operation is divided into search periods of $PERIOD$ time length, with the search being evaluated at the end of each search period and new search tasks being allocated at the commencement of each period. Search resources are replaced when they become fatigued.

The primary objective of the SAR operation is to locate the subject as quickly as possible *i.e.*, to maximize POS. Secondary objectives to this include minimizing redundant route collision between search resources, minimizing the unallocated search hours of search resources, maximizing the number of clues detected, and minimizing the number of task reassignments due to strategy changes.

10.3 The Simulation Model

The DES models the SAR operation over time by representing changes in the model's variables at discrete time points. The basic components of a DES are defined below, detailing these with respect to the SAR operation.

10.3.1 System State

The system state variables are those variables which together provide enough information to describe the status of the SAR operation at any particular time. Within this model these variables are:

- whether the subject has been detected as yet;
- the search tasks assigned to each search resource for searching in this period and the POD level that they will be searched at;
- the location of each search resource;
- a weather and light indicator, and variables indicating the time and level of the next predicted weather change;
- the level of urgency;
- the time remaining in the current search period;

- the cumulative POS achieved for the operation to date;
- those regions which have been searched and to what level (POD_{cum}), and those regions still to be searched ;
- the POA value assigned to each search region;
- the regions which comprise the primary search area, if this is defined;
- the level of visibility over each terrain characterization;
- the level of traversal difficulty over each terrain characterization;
- the total time that has passed since the commencement of the operation and specifically the number of search hours utilized;
- the total time each individual search resource has spent in the field;
- the worst weather conditions experienced thus far;
- the status of regions susceptible to flooding; and
- the information gained from clues detected, including the time at which the last clue was detected.

10.3.2 Entities

The resources which the SAR operation utilizes and the objects it consists of are described as entities of the DES. Those entities requiring representation in this model are:

- the subject;
- search resources — in this model once ground searchers are initially assigned to teams the searchers comprising a team are considered as a single entity;
- the SAR management team;
- the search base;
- search tasks; and
- clues.

Transport vehicles could also be included but we do not utilize these within our model.

10.3.3 Attributes

Each entity of the DES is described by a number of attributes.

- The subject:
  - actual path taken;
  - present location;
  - intended path (if known);
  - responsiveness level — dependent upon the time that the subject has been outdoors, the worst weather experienced to date and any injuries the subject may have (detailed in Section 5.1.10); and
  - profile of experience, equipment, known medical conditions or injuries, and time spent outdoors.[1]
- Search resource:
  - type,
    - ground searchers, and
    - helicopter;
  - location and current assignment;
  - future search task assignments;
  - detection ability;
  - useful search hours;
  - time spent in the field to date; and
  - the number of ground searchers comprising a team and the number of resources used in total

  Specialist skills could also be included as an additional attribute.
- The SAR management team:
  - location.
- The search base:
  - location.
- Search tasks:

[1]These factors together with weather conditions give an indication of the urgency of the search.

- POD level and the spacing of search resources required to achieve this;
- predicted POS and planning criteria value;
- number and type of resources required; and
- search method,
  i) visual sweep search,
  ii) sound sweep search,
  iii) linear feature search, and
  iv) the use of an edge of the search region graph as the means of access to another search task.

- Clues:
  - position;
  - type;
  - detectability by a search resource;
  - direction indicator to a vertex of the TIN; and
  - variable indicating whether detected in the course of the operation or not.

10.3.4 Events

Events are described as instantaneous occurrences which alter the system state. Events which are considered are the:

- commencement of searching;
- beginning and end of a search period;
- detection of the subject;
- detection of a clue;
- completion of an edge traversal by a search resource;
- completion of the search of a triangular region by a search resource;
- completion of a redeployment of a search resource during an edge or triangular region traversal;
- commencement or completion of a rest break by a search resource;
- commencement or completion of a communication with search management by a search resource;

- completion of a corporate briefing of search resources;
- change in weather conditions;
- the flooding and later re-setting of regions susceptible to flooding;
- beginning and end of dawn conditions;
- beginning and end of dusk conditions (with the end of dusk conditions coinciding with the onset of nightfall);
- arrival of rescue resources or a helicopter at the subject's location;
- completion of the loading of the helicopter in a subject retrieval;
- scale-down or suspension of the search operation, as a result of a decision made by the SAR management team;
- the subject becoming unresponsive; and
- scale-up of the search operation as a result of the urgency indicator reaching level 1.

10.3.5 Activities

Activities are defined as those actions within the model which require a known amount of time to complete.[2] Activities considered here are:

- travel of the search resources to the search base;
- execution of an assigned task by a search resource;
- search path determination and subsequent briefing of resources;
- replacement of a search resource in the field once they have exhausted their useful search hours;
- search resource redeployment;
- search resource rest break;
- search resource communication with the base; and
- rescue of an injured subject after detection.

[2] Although this time may vary with stochastic factors.

10.4 Assumptions of the Model

Nebiker [120, page 10] states that:

> "In simulation we say a good model is like a good political cartoon; it identifies and enhances salient points and minimizes or eliminates unimportant ones."

Similarly the "law of parsimony" is defined as "the assertion that no more causes or forces should be assumed than are necessary to account for the facts", [2, page 867].

As the large number of differing factors, which may be considered for any SAR operation, can become quickly unmanageable, we have limited the number of factors and scenarios that will be considered. We have attempted to justify such assumptions by including the most significant causes that will affect the search outcome and excluding those that would have a small impact over a sufficiently wide range of parameter values. The following assumptions explicitly define the limits that we have placed upon this simulation model.

- The subject consists of a single entity and he is not positioned within a region which could become flooded.
- The subject is located within the search region as defined by the given TIN.
- All searchers have identical fitness, experience and skill attributes. As all SAR personnel are required to achieve a set level of fitness we assume that any additional variation is negligible. This is described in New Zealand Land SAR documents as a desired goal to be achieved by the year 2000 [79].
- There exists an unlimited pool from which to draw search resources.
- Communication is possible over all areas of the terrain being searched.
- Communication occurs only between the search management and a single search resource at any given time. No communication schedule, as such, is utilized, with communication occurring only in response to specific events.
- Each search resource searches for a maximum of six hours in any one day, and for a maximum of four consecutive days. This defines a resource's useful search hours to avoid fatigue-induced errors and is in line with currently upheld recommendations. Hence fatigue is not explicitly factored but only implicitly factored in useful search hour allocation.
- The allocation of search tasks only occurs if the time available for allocation exceeds the parameter $MINLIMIT$ (currently set to two hours).

- Search regions once defined do not alter over the duration of the search and are initially defined as being equivalent to a single entity of the TIN, *i.e.*, either an edge or a triangle.
- Decisions to scale-down or suspend a search are made at the end of a search period and hence only affect search assignments to be made for the following search period.
- Search bases are located at vertices of the TIN.
- Search resources reach an interior search base by foot, and all exterior vertices are accessible by a four-wheel drive vehicle. Specific paths to TIN perimeter points are not considered.
- Helicopters are only utilized for subject retrieval when the weather conditions are suitable, not for transportation or searching purposes.
- No search dogs are utilized as this is currently not a common practice in New Zealand searches.
- Visibility restrictions exist as outlined in Chapter 4.
- When a triangular region is searched the edges of that region are not searched. It is assumed that each edge constitutes a region in itself and that a triangular region search searches only the interior of that region.
- A region is searched by a maximum of one resource at any given time.
- Sound sweep search methods, when utilized, are only conducted over triangular regions of the TIN, not edge regions.

10.5 Inputs To The SAR Simulation

The inputs to the SAR simulation are:

- visibility measures and traversal times (delineated for both 'access-only' and search traversals) for each terrain classification;
- time of day that the search operation commences;
- random number streams to enable replicability of experiments — in particular such streams are required to determine,
  - the subject path, when generating a random path or when the subject shows evidence of confusion,

  - the detection of the subject and clues,
  - clue positions,
  - subject responsiveness, and
  - weather input data;
- a rule to determine when the simulation will halt;
- the starting weather conditions and an input file of times at which the weather conditions will improve or deteriorate from the current level, by one step;
- TIN,
  - number and classification of each edge type, including susceptibility to flooding,
  - vegetation classification for each triangle,
  - level of variability in elevation and vegetation,
  - total area covered; and
  - location of huts;
- POA distribution over the regions comprising the TIN;
- solution method to be used;
- management directives to be followed;
- the actual path taken by the subject and the position of any clues laid;
- the subject's intended path if known;
- status of the subject; and
- POD values associated with each combination of search method, terrain type, clue or subject type, environmental conditions, and search resource type.

10.5.1 Simulation Parameters

We define the following set of parameters which are used to simulate a SAR operation, along with the specific resource allocation parameters outlined in Section 8.8.

PERIOD The time length of a search period. This can also be interpreted as the planning horizon of the search operation.

DAYSTART The time of first light.

DAYEND The time of nightfall.

SUSPEND The number of days of searching which are to be completed before suspension of the operation is considered, in conjunction with the minimum number of days which must have passed since the discovery of the last clue detected, **CLUE_DAYS**.

SCALE_DOWN The number of days of searching which are to be completed before scaling down the search operation is considered.

KNOW A Boolean parameter indicating whether the intended route of the subject is known.

START_RESPONSE A Boolean parameter indicating whether the subject is responsive at search commencement.

START_INJURED A Boolean parameter indicating whether the subject is injured.

START_URGENCY The urgency level set at the commencement of the search operation.

POAROW The POA value initially assigned to the ROW region.

ROC_0 The value used to assign the initial $ROC_{cum,i}$ value for each search region i to facilitate the update of POA values after clue detection.

SELECT_BASE Whether an exterior or an interior vertex of the TIN is selected as the search base.

NIGHT_SEARCH A Boolean parameter indicating whether night searching is conducted.

SOUNDSWEEP A Boolean parameter indicating whether the sound sweep searching technique is utilized.

SOUNDPERIOD The number of periods that sound sweep searching is applied in from search commencement.

MINLIMIT The minimum number of search hours which must be available before search paths are generated.

TEAMNUM The number of resource teams assigned to the field at the commencement of the search operation.

TEAMSIZE The number of resources allocated to each team. Teams of size 4, 6 and 8 are considered.

SCALE_EXTRA The number of additional teams assigned to the field when the operation is scaled up.

REST_STOP The length of a rest break for a searching resource.

BRIEF_TIME The time needed to brief resources.

COMM_EXTRA The additional communication time required when two pieces of information are communicated with a resource at the same time.

STRETCHER_FACTOR The multiplication factor of traversal difficulty when executing a stretcher retrieval of an injured subject.

10.6 Simulation Objectives

Using the vehicle of simulation we investigate some of the decisions made by SAR personnel in the field. These include:

- determining the location of the search base;
- identifying priority regions and ordering these for searching;
- assigning search tasks to resources;
- adjusting operational procedures under differing terrain and weather conditions, and urgency levels;
- adjusting operational procedures upon receipt of new information;
- redeploying search resources to defined areas; and
- calling in additional resources or scaling down a search when necessary.

These are decisions which must necessarily be made in real-time and under differing degrees of knowledge, for example, the knowledge of the intended route of the subject. Due to the uncertainty which is implicitly present, we attempt to determine those search heuristics that are the most robust, efficient and successful over differing scenarios. The simulation model can be viewed as a learning tool for developing search strategy to guide the search operation to a successful outcome.

10.7 Implementation

At the conclusion of each simulation run, relevant data is recorded to a file to enable a graphic portrayal of the search paths taken by each resource over the period of the operation. In addition to plotting search paths the graphics function also plots the position of the subject and the search base.

The actual simulation of the SAR operation follows the broad schema outlined in Figure 10.1.

10.7.1 Time Mechanisms

As the SAR simulation dynamically responds to time-triggered events, there need to be mechanisms in place which record the actual times involved in the differing activities of the simulation, in order for events to be successfully initiated. We use the following time record and advance mechanisms:

day_clock Actual elapsed time of the simulation.

time_clock The number of whole days for which the simulation has been running. This is used to determine whether or not the search operation will be scaled down or suspended in light of other information such as the urgency level.

resource_start_time The time that a resource becomes active in the search operation.

resource_path_clock The total number of consecutive hours that a search resource has spent in the field during the simulation. This mechanism records the time spent searching: in accessing regions; in redeployment; in communication with the search base; in rest breaks; and in down time between consecutive search periods. This clock allows for the differing traversal times of regions in poor light or bad weather to be recorded accurately.

resource_down_time The total amount of time spent between periods by an active resource who is not currently searching due to a temporary search suspension.

stop_clock The exact time in the simulation at which the subject is detected.

period_start The time that the current search period commenced or, if between periods, the time that the next search period is due to commence.

period_end The time that the current search period is due to finish.

lost_days The number of days of searching lost due to adverse weather conditions.

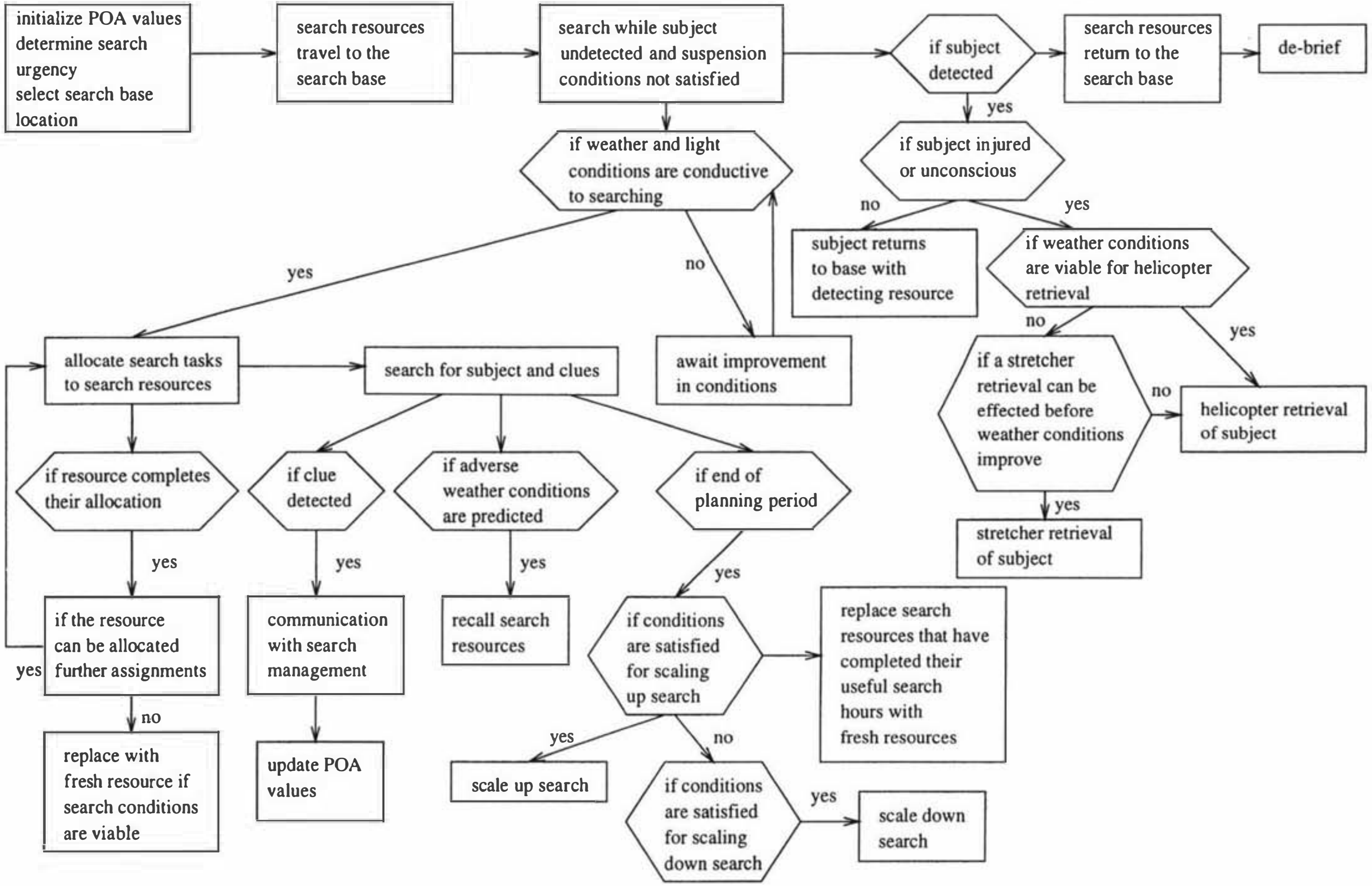

Figure 10.1: A broad schema of the SAR simulation model.

weather_clock The time of the next weather change.

flood_time The time at which regions susceptible to flooding will flood.

recall_time The time at which search resources are recalled to the search base in adverse weather conditions.

10.7.2 Path Representation

The tasks assigned to a search resource are represented by a vertex path, a task path and a task-type path. The task path stores the labels of the search regions in the order of assignment while the vertex path consists of the entry and exit vertices of each region. Pictorially these can be represented as in Figure 10.2 where the vertex path is: 0–1–5–6–7–10–9 and the task path is: 11–14–16–19–12–16.

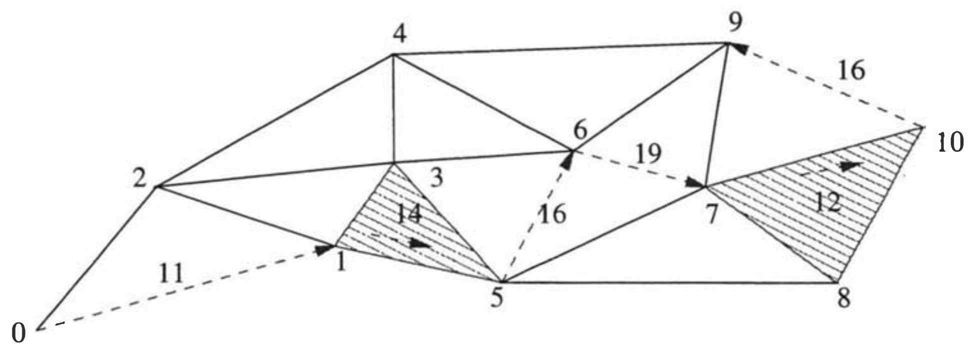

Figure 10.2: Path representation.

The task-type path records which search method is to be utilized over the corresponding search task. The number of tasks comprising a path and the total time predicted to complete these is also recorded.

10.7.3 Resource Positions

As well as maintaining accurate information as to the time that any search resource has spent in the field, it is also necessary to be able to determine their progress and position at any given time. In order to achieve this, a position indicator for each resource is held and updated as each region in the resource's path is traversed. The indicator records the labels of the entry and exit vertices of the region that the resource is currently searching, along with the index of that region in the search path. Any change in path allocation can then be determined easily and search management can see clearly those resources in the best position to redeploy.

10.7.3.1 Recording Task Times In Changing Conditions

When weather or light conditions alter to create a new state, traversal costs also alter changing the time to complete current and future tasks, and any detection events pertinent to that resource. It is possible that more than one such state change may occur over the period of the task execution, as in the case where a weather change occurs just prior to dusk. In these instances it is necessary to have a mechanism for recording the time which has elapsed while on the task and the fraction of the task that has been completed. As the *resource_path_clock* indicator is updated with each communication or rest break interruption, it is insufficient to record this information; hence a new variable *cost_change* is initialized to record this data.

Such a state change is illustrated in Figure 10.3 for a resource searching over both an edge and a triangular region. A change in weather or light conditions has occurred at the point represented by '*'. Each state is shaded in a different pattern.

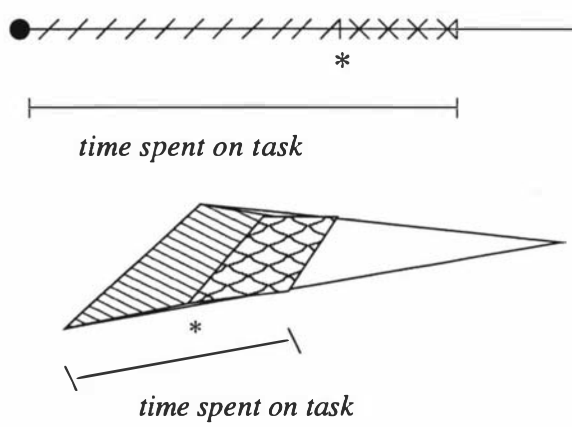

Figure 10.3: State changes over search task execution.

Where the search task is an 'incomplete' assignment, *e.g.*, the resource searches only a portion of the task before being redeployed and this redeployment involves the resource returning over the searched portion of the region, the *cost_change* variable is also utilized. It is used to record the unsearched fraction of the region to ensure that later calculations to determine the correct time of assignment completion are accurate. This situation is illustrated in Figure 10.4. In this instance the *resource_path_clock* of the resource is incremented at the point of turn-around.

At any point, the time spent searching by a resource since the previous *resource_path_clock* update is easily calculable, enabling the total fraction of the task, and hence the time remaining to complete the assignment, to be determined.

If the resource was to have detected either a clue or the subject while executing their

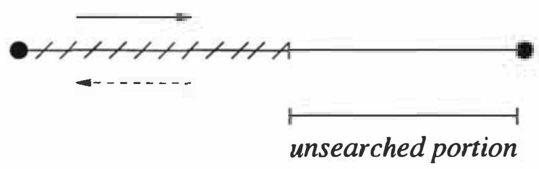

Figure 10.4: Incomplete assignments.

current task, then the actual time of detection is updated to account for an increase or decrease in resource speed, and an increase or decrease in the visibility and sound measures.

Rules and heuristic approaches are developed in the next chapter to simulate a response to each of the events that may occur within the simulation model and to execute each function of the operation, thus automating the decisions which would normally require a human decision-maker.

Chapter 11

Simulation Modules

The simulation of a SAR operation provides an environment within which to test and compare heuristics designed to cope with: changes in weather conditions; visibility limitations; the redeployment and recall of search resources, as appropriate; the scaling-up or scaling-down of the search, as factors dictate; search task determination and resource allocation at each planning period; and the call-in and assignment of new search resources, as required.

The following sections examine, in closer detail, the heuristics and rules developed to simulate each stage of a SAR operation.

11.1 Initialization

The initialization of the data and indicators for the SAR simulation are executed as follows.

11.1.1 TIN Initialization

The TIN data is downloaded from files describing all its elements and characteristics, including the type and vegetation classifications of each edge and triangular region, the location of any huts, and the set of all edges which will flood in adverse weather conditions (level 4). The visibility and traversability measures for each vegetation classification present in the TIN are initialized for the starting weather and light conditions, and the traversal times along all edges are determined, together with the resulting shortest path matrix between all vertex pairs.

11.1.2 Search Region Initialization

Initially each edge and triangle entity is defined as a search region and is assigned:

- an identifying label;
- a set of defining vertices;
- density (vegetation classification);
- area, or length (if more applicable);
- POA value;
- POD_{cum} value; and
- ROC_{cum} value.

An array *search_region* records whether or not each region is assigned to a search resource in the current search period. Once a region has been covered in its entirety, either searched or traversed as access, the cumulative POD for that region is then greater than zero.

11.1.3 Indicator and Variable Initialization

The following indicators and variables are set to equal null in the function **initialize()**:

- all clocks and day counts, period commencement and completion times, and total search down time;
- period counter, and for each period the cumulative POS and coverage achieved, and the number of completed tasks;
- subject detection indicator and detecting resource;
- search operation scaling (up or down), recall and suspension indicators;
- resource task paths, position and *cost_change* arrays, POD, predicted POS and spacing variables;
- resource indicators for returning to the search base, reaching the end of an allocated search path, resting while searching, communicating with search management, being in the field, needing replacement, being on route to the subject or waiting with the subject, predicted to detect a clue;
- dusk, dawn and night indicators;

- the total number of resources used in the operation;
- rescue by helicopter indicator and the set of resources needed to assist with a stretcher retrieval of an injured subject;
- corporate briefing indicator;
- flood conditions indicator; and
- variables which track the time that the first and last clue found were detected, and the time at which the *lost_region* is first allocated to a resource for searching.

Additionally, the following variables are also initialized:

- the first simulation event — this is defined as the commencement of the search operation;
- the starting weather and light conditions; and
- the time that the last clue was detected is recorded as the time of operation commencement, where the 'clue' records all available information on which the search is planned.

The position indicator, resource spacing, the first rest stop event, and the initial search task completion event are initialized for each search resource in the function **initialize_resource_position()**. In addition, this function calls the function **detect_clue()** to determine if the resource will detect a clue left by the subject while executing this task and at what time this event is predicted to occur.

The first event called, **begin_search_operation()**, initializes variables for the commencement of the search operation and executes the procedures outlined in Algorithm 11.1.

11.2 Search Base Allocation

11.2.1 Location Of The Search Base

In this model we consider only the vertices of the TIN as possible search base locations. The vertices of the TIN are the only points in the search region graph which are used as entry and exit vertices of regions. Hence locating the search base at a vertex provides both greater coverage and flexibility.

Candidate search base vertices are selected from the set of exterior (convex hull) vertices or interior (within the convex hull) vertices as two base placement rules. Vertices on the convex hull are viewed as immediately accessible, whereas the vertices lying interior

Algorithm 11.1 function *begin_search_operation*()

```
urgency ← START_URGENCY()
initialize dawn event at DAYSTART
initialize dusk event at DAYEND − 0.5
day_clock ← DAYSTART
weather_change()
set time at which subject will become unresponsive
if (KNOW = 1) then
    input subject's intended path
    update POA values for search regions based on this information
end
read in subject's actual path from input file
read in clue data from input file
base_allocation()
if (urgency = 1) then  scale_up(SCALE_EXTRA)
generate paths to base for each resource
end
```

Algorithm 11.2 function *base_allocation()*

```
for i = 0 to n do
    dist_i ← shortest, average distance to five regions having highest POA values
    count_i ← number of times i appears as intermediate vertex in all shortest paths
end
for i = 0 to n do
    drank_i ← rank in decreasing order of dist_i(ties have same rank)
    vrank_i ← rank in increasing order of count_i(ties have same rank)
end
for all candidate vertices(interior or on convex hull) do
    if vertex not in danger of flooding then
    obj_i ← w1 × vrank_i + w2 × drank_i
end
base ← vertex with maximum obj_i value
end
```

to the convex hull are assumed in the model to be accessible only by foot. A secure location is desired which will not experience flooding in adverse weather conditions. Hence any vertices which may flood or which may be undesirable with respect to some other factor not directly considered here, such as a lack of clear ground space or communication restrictions, are not considered viable candidates for base location.

A candidate base location is evaluated upon two criteria; the proximity of the vertex to regions of high POA and the occurrence of the vertex as an intermediate component in the shortest paths calculated between all vertex pairs. The second criterion forms a pseudo-measure of centrality in terms of access paths to regions. Both criteria seek to measure, and minimize, the distance resources based at the search base will have to travel in order to commence their search tasks. We arbitrarily assign the weights $0.7(w2)$ and $0.3(w1)$ respectively to each of these criteria to obtain an overall additive measure of desirability. Weight $w2$ is greater then weight $w1$ in order to assign greater importance to the vertex with closest proximity to the set of regions having the greatest likelihood of containing the subject. This base selection process is outlined in Algorithm 11.2.

11.2.2 Transport Of Searchers To The Search Base

Once the location of the search base has been determined the next phase in the simulation model is to transport the ground search resources into the base. In those instances where the base is located at an interior vertex, the search resources make their way from the exterior vertex, closest to the search base, on foot via the shortest edge path. The function **transport_to_base**() plans the actual path to an interior base. The POD level over edges comprising this path is 100%, as for all edge traversals. Transport to exterior vertices is not explicitly considered in this model, but access is assumed to be viable via 4-wheel drive vehicle, or similar.

An alternative modelling option could consider only one exterior vertex as accessible *i.e.* the TIN would have only one entry and exit point when helicopter access is not utilized; all bases would then be accessed from only this vertex. Under this option an additional criterion could be added to the base selection, namely proximity to this access point.

11.3 Search Urgency

The urgency level of the search is initially set by considering the information available to the search management team at the outset of the operation. While the urgency level can be increased at a later stage in the search in response to the duration of the search operation and deteriorating weather conditions, we assume that the urgency level will not decrease from its original level within the simulation environment. In reality, however, this may occur in response to new information which contradicts the initial information upon which the urgency level was set, for example the subject may turn out to be well equipped or to not have a medical condition. The urgency levels considered are 1 (high urgency), 2 (moderate urgency), and 3 (no initial urgency).

11.3.1 Urgency Action Level

The urgency action level of the situation is determined from Algorithm 11.3, where responses to nine classifying questions are sought from the search management team. The algorithm is based upon the urgency chart found in LaValla and Stoffel [103], but with additional categories added to evaluate the areas incident history (as Wade originally did [86]) and the length of time that the subject is overdue. Each response is at a level of 1,2 or 3, with level 1 signifying the most urgent scenario. Summing the responses over all categories results in a relative measure of urgency.

LaValla and Stoffel [103, page 68] state that: "the combination of the factors affecting urgency will help determine not only how quickly to respond, but the nature and level of response as well." We utilize the urgency response levels used by Handel in his Relative Urgency Response Factors software adapted from Wade's rating system [86]. (This software also considers responses to nine questions.) These are detailed in Table 11.1.

Table 11.1: Urgency response level.

Factor Sum	Response Level	Urgency Level
8–12	emergency response	1
13–18	measured response	2
19–24	evaluated response	3
25–27	missing person	

We define the level of urgency as 1 (the highest possible level), if the responses sum to less than or equal to 12. An urgency level of 2 is assigned if the sum of the responses falls between 13 and 18. If the responses sum between 19 and 24 an urgency level of 3 is initialized. If the factor sum is greater than 24 a search operation is not actioned at this time.

If the urgency level is initially set to level 1 then we assume that the level remains at 1 throughout the operation. If the urgency level is initially set to level 2 or 3, then the urgency level may be upgraded to a higher level as the search progresses without detection, taking into consideration the number of days that the operation has been in progress and the weather conditions — both predicted and experienced. In particular, we

Algorithm 11.3 function *urgency*()

```
if subject very young or very old then
age ← 1
else
age ← 2 or 3
if subject has known medical condition, or know/suspect to be injured then
medical ← 1 or 2 depending on seriousness
else if (subject healthy or a known fatality) then
medical ← 3
if subject is traveling solo or separation of party has occurred then
number ← 1
else
number ← 2 or 3
if subject is inexperienced and does not know area then
experience ← 1
else if (subject is inexperienced but knows area) then
experience ← 1 or 2
else if (subject is experienced but does not know area) then
experience ← 2
else if (subject is experienced and knows area) then
experience ← 3
if subject carrying inadequate equipment then
equipment ← 1
else if (subject carrying questionable equipment) then
equipment ← 1 or 2
else if (subject carrying adequate equipment) then
equipment ← 3
if time elapsed since subject due out of area is more than 2 days then
time_elapsed ← 1
else if (time elapsed since subject due out of area is 1–2 days) then
time_elapsed ← 2
else if (time elapsed since subject due out of area is less than 1 day) then
time_elapsed ← 3
if terrain contains known hazards then
terrain ← 1
else if (terrain contains few or no hazards) then
terrain ← 2 or 3
if there exists past or current hazardous weather then
weather_level ← 1
else if (hazardous weather is predicted within the next 8 hours) then
weather_level ← 1 or 2
else if (hazardous weather is predicted past the next 8 hours) then
weather_level ← 2
else
weather_level ← 3
rate the seriousness of previous incidents in the area
history ← 1,2 or 3

level ← age + medical + number + experience + equipment + time_elapsed
+ terrain + weather_level + history
return(level)

end
```

Algorithm 11.4 function *urgency_upgrade*()

if *number days* ≥ 2 **or** *worst weather level* = *4* **or** *time to weather level 4* < 8 *hours* **then**
  if *initial urgency level* = *2* **then**
    urgency $\leftarrow 1$
  if *initial urgency level* = *3* **then**
    urgency $\leftarrow 2$
end
if (*number days* ≥ 4 **and** *initial urgency level* = 3) **then**
urgency $\leftarrow 1$

end

arbitrarily upgrade the urgency level under the circumstances given in Algorithm 11.4.

11.4 Search Periods

The search period is defined over a constant time length, $PERIOD$. However, if light conditions fail over this period, the search phase may be terminated before $PERIOD$ hours have elapsed. This will occur due to a detrimental weather change prior to, or during, dawn, dusk or nightfall. If no night searching is to take place then the search period will terminate at the onset of nightfall. A period is required to be at least $MINLIMIT$ hours in length before search assignments are allocated to resources. Figure 11.1 shows a representation of two consecutive search periods with are separated in time. The down time component occurs due to either poor visibility or a decision by management not to search during the night.

As the number of daylight hours is dependent upon the time of year we have utilized the 12 hours from 6am to 6pm as the daylight hours for this simulation environment.

Figure 11.1: Visual representation of consecutive search periods.

11.4.1 Period Commencement

At the commencement of a search period the function **begin_search_period()** of Algorithm 11.5 is called to facilitate search task allocation over this phase of the search operation. The length of the search period is determined and variables are initialized to

Algorithm 11.5 function *begin_search_period*()

```
periodct ← periodct + 1
determine target POD level for the period
determine time of period completion
initialize end of period event
determine path_limit
if (path_limit ≥ MINLIMIT) then
    determine maximum POD level each region can be searched at within path_limit
    initialize search allocation planning variables
end
if PRIMARY_REGION set and subject not detected then
primary_search_region()
define set of candidate regions for searching
if (periodct = 1) then
    for k = 0 to resources do
    start_k ← search_base
    if subject can be visually detected from the base then
    initialize rest stop for find_team
    else if (path_limit ≥ MINLIMIT and recall = 0) then
    assign search tasks with corporate briefing
end
else
    down_time ← day_clock − period_end
    if (down_time > 0.0) then
        urgency_level()
        if (urgency = 1) then
        scale_up(SCALE_EXTRA)
    end
    if no resources in the field and subject undetected and suspend = 0
     and recall = 0 then
        if (path_limit ≥ MINLIMIT) then
        assign search tasks with corporate briefing
    end
    for k = 0 to resources do
        if resource in the field then
            resource_path_clock_k ← resource_path_clock_k + down_time
            resource_down_time_k ← resource_down_time_k + down_time
            re-initialize rest stop for resource k
            re-initialize the completion time of current search task
            if subject undetected then
            detect_clue()
            if (k = find_team and down_time > 0.0 and
            subject undetected) then update time to detection of subject
        end
        else if (subject undetected and suspend = 0 and recall = 0) then
        assign search tasks to resource k with individual briefing
    end
end
if (find_team = −1) then
determine if subject detection will occur over newly generated paths
value ← objective()
record period assignment information to output file
if (suspend = 1) then suspend_search()

end
```

aid in planning the search allocations. A primary search area is defined if required and the target POD level for the period is calculated.

The limit on the time duration of a search path assigned to any resource is set as a parameter of the simulation with respect to the type of path heuristic selected. However, if the predicted weather level at the start of the period is level 4, then the number of hours until this weather change is to occur are taken into account when defining the path limit. In particular, search paths are only allocated up until two hours prior to this weather change. This is an application of risk management. For example, if the path limit is set to 6 hours but a weather change to level 4 conditions is predicted to occur in 5 hours time, then the path limit is reduced to 3 hours.

If the path limit exceeds $MINLIMIT$ and resources are not being recalled to the search base due to search suspension or adverse weather conditions, search paths are generated for all active resources. When a forecast of level 4 weather exists and the flooding of regions is due to occur within the current search period, no regions at risk of flooding are allocated to resources, either as regions to search or as access paths to other regions. If all resources are located at the search base then a corporate briefing of the resources follows, otherwise the search assignments are individually communicated with each resource.

In line with the dynamism inherent in the model, the spacing at which each region search is to be conducted to achieve the planned POD level is not determined until the region is about to be searched. This ensures that the current weather and light conditions are taken into consideration.

If there was down time between this period and the preceding period, the *resource_path_clock* variables for all resources in the field are adjusted in light of this. The search is also scaled up if the urgency level has increased to level 1. The events pertaining to each resource are also re-initialized for all resources currently located in the field.

Within *begin_search_period()* the objective value of the proposed search assignments is calculated, and this is recorded to an output file, together with the assignments and related information.

11.4.2 Period Completion

The function **end_search_period()**, detailed in Algorithm 11.6, coordinates the carry-over between consecutive search periods. Within this function performance measures for the preceding period are recorded and the urgency level of the operation is increased if the conditions justify this.

Past events are deleted from the simulation event array and future events are either deleted, if no longer required, or re-scheduled after the commencement of the next search period. The time of commencement is determined by the function **next_period()** in Algorithm 11.7, which takes into consideration weather and light conditions, and whether or not night searching is to be undertaken.

At the end of the search period a resource stationed in the field and not scheduled to return to the search base has all tasks beyond their current task deleted from their search path. If the resource needs to be replaced by a fresh resource they are instructed to return to the base at the beginning of the next search period, otherwise they will complete their current task at the beginning of the next period and then be allocated further search tasks.

At the completion of a search period the decision to suspend the search operation is made if the subject remains undetected, and the number of search days (taking into account any days lost due to adverse weather conditions) has reached $SUSPEND$, and the last clue (relating to the subject) to be detected was found more than $CLUE\_DAYS$ ago. The function **suspend_search()** is initiated at the beginning of the next search period to achieve this when some search resources remain in the field.

The search operation is scaled-down if the number of search days has reached $SCALE\_DOWN$ and the last clue to be detected was found more than $CLUE\_DAYS$ ago. The exception to this rule is if the search operation has been scaled-up within the preceding 48 hours or the urgency level has just reached level 1. The scaling-down of the operation is effected by the function **scale_down()**.

The scaling-up of the operation occurs under two scenarios; the first is if the number of resources needed to effect a stretcher carry of an injured subject is greater than the number of currently active resources, the second is the first time the urgency level reaches a value of 1. The second scenario is actioned at the beginning of the search operation if the urgency of the search begins on level 1, otherwise at the end of the search period in which the urgency level first attains level 1. In this scenario the number of additional resources sent into the field is described by the simulation parameter $SCALE\_EXTRA$. The number of active resources is increased by the function **scale_up()**.

11.5 Event Functions

The DES is coordinated by an event array that stores the time of each event within the simulation. When a new event is initialized its correct time sequencing within the event array is found and the event is inserted at this point. The only events which may occur between search periods are **weather_change()**, **flood_regions()** and **unresponsive_subject()**, as search resources are stationary at this time. The event array is

Algorithm 11.6 function *end_search_period*()

```
record to file cumulative POS and total coverage achieved for the period
if end of a day then
    time_clock ← time_clock + 1
    urgency_level()
end
for k = 0 to resources do
    if k not located at a vertex then
    start_k  ← vertex resource k is travelling towards
    else
    start_k ← vertex resource k located at
    if resource not returning to the search base then
    delete all scheduled tasks past the resource's current task
end
next_period()
if weather change is to occur between periods then
update cost_change array
update simulation event array
if search suspension conditions satisfied then
    if no resources in the field then
    suspend search
    else
    suspend ← 1
end
else
    if scale down conditions satisfied then
    scale_down()
    for k = 0 to resources do
        if resource k is to be replaced then
            if resource k is at the base then
            replace with fresh resource
                else if (resource not already returning to base) then
                instruct resource k to return to base at beginning of next period
            end
        end
    end
end
if urgency = 1  and not yet scaled up search then
scale_up(SCALE_EXTRA)

end
```

Algorithm 11.7 function *next_period()*

> **if** *(weather_level* $= 4$ **and** *dawn or dusk conditions)* **or** *(weather_level* $= 3$ **and** *night conditions)* **then**
>   *// resources remain in field until weather or light conditions improve*
>   *period_start* $\leftarrow$ next weather or light improvement consistent with night searching position
> **end**
> **else if** (*no night searching* **and** *(nightfall* **or** *time to nightfall* $< MINLIMIT$)) **then**
>   *// commence searching at next day light where visibility is viable*
>   **if** (*weather_clock* $<$ *dawn_start* **and** *new_weather* $= 4$) **then**
>   *period_start* $\leftarrow$ *dawn_end*
>   **else**
>   *period_start* $\leftarrow$ *dawn_start*
> **end**
> **else**
>   *// immediate carry-over to next search period*
>   *period_start* $\leftarrow$ *day_clock*
> **end**
>
> **end**

updated at the end of each search period by deleting past events and reinitializing events to be carried over to the next period. A similar compression of the array is executed when a maximum number of events have been initialized in any single search period. Any superseded events are also deleted within the event array. The functions which execute the events consist of the following set, together with the previously defined functions **begin_search_operation()**, **begin_search_period()**, and **end_search_period()**:

sign_found() is called when a clue is found by a search resource. The relevance of the clue is determined and the POA values for all regions are updated accordingly. The *ROC_cum* value for the region containing the clue is updated and the clue is recorded as detected, with this information and the time of discovery also being recorded to the output file of the simulation. The discovery is communicated with search management and the function then determines whether the resource will detect any further clues within this region, initializing such an event if it will occur.

end_task() is called when a search resource completes the search or traversal of a region on their search path. If the region has been completely traversed it is recorded as such and the POD_{cum} variable adjusted. The POA value of the region is also updated for non-detection. The total POS and coverage achieved for the period is increased by the POS and coverage resulting from the task. The resource's *resource_path_clock* is updated. If the search resource is at the end of their current path and is not returning to the search base or travelling to the subject, the function **end_of_path()** is called; otherwise, the position vector for the search resource is updated to represent the next task in their search path. The time to complete this task is determined and the next *end_task()* event is initialized. If the resource is about to enter the region containing the subject for the purposes of participating in their retrieval, the actual time to reach the subject is determined. The function also determines whether or not the resource will detect the subject or a clue while completing their next task. If the next region in the path is an edge region which will flood in level 4 weather conditions, a check is made to determine whether it can be successfully completed (with an additional margin of half the traversal time again) before such conditions are reached. If this is not possible and this region is the last task in the path, or all remaining tasks in the path will also flood, **end_of_path()** is called, otherwise the resource is redirected around the edge via the shortest path.

begin_communication() is called at the beginning of a communication event between a resource and the management team located at the search base. The procedure initiates the end of communication event and, if the resource is currently searching or traversing a region, the time to complete this task is updated to allow for the communication time and the previous task end event is replaced. Similarly, the time to detection of a clue or the subject is also adjusted, as is the arrival at the subject's location. The resource's path clock is updated in **end_communication()** and the communication flags are reset.

begin_rest_stop() halts and rests a search resource for *REST_STOP* (currently set at five minutes) for every hour of searching, or every 30 minutes of a stretcher carry. The procedure generates the updated time of completion of the resource's current task and initiates the end event of the rest break. If the resource being rested is predicted to detect the subject or a clue, but detection has not yet been achieved, the time to detection is also updated. The resource's path clock is incremented, the flags reset and the next rest stop event initiated in **end_rest_stop()**.

weather_change() is called at the beginning of the operation and again with each change in weather conditions. If the SAR operation is just commencing, the starting weather conditions are found from the input weather file. Future weather conditions comprise an increase or decrease in the weather level by one level as determined from the weather file. The timing of future weather changes is also found from the input file and the 'predicted' conditions are stored in the variable *new_weather*.

The function **changes()** is called to co-ordinate the environmental changes and the *weather* variable is set. The light level is also altered by one level in parallel to the changing weather conditions. If the visibility conditions deteriorate to light level 4, searching is terminated for this period by initiating **end_search_period()**. If the weather conditions reach level 4 *flood_time* is initialized two hours from now. When weather conditions of level 4 are predicted, a recall of all active search resources is initiated. This recall is set to occur two hours prior to when the weather conditions will deteriorate. If this deterioration is due to occur during the night (and hence between search periods) the resources are recalled two hours prior to nightfall. If both the prediction and occurrence of level 4 conditions occur between search periods then resources are recalled at the beginning of the next search period. If the predicted weather conditions are level 4 and *flood_time* will occur within this search period, the function **flood_check()** (defined later in this chapter) is also called for search resources en route to the search base or subject, in order to re-route their current search paths around edges which will flood at this time. The urgency level of the search is updated if such conditions are satisfied.

recall_resources() All search resources in the field are recalled to the search base, if they are not already en route there or en route to the subject. If flooding has not yet occurred, resources are recalled via the shortest-coverage edge path developed in Section 11.9.1, otherwise, by the shortest edge path available.

dawn() coordinates the visibility changes experienced throughout the dawn period — the 30 minutes following *DAY_START* . The function sets the light level indicator at one level higher than the current weather level indicator. If the light level exceeds level 4 search resources are not sent out until after the dawn period, otherwise the function **changes()** is called to alter the traversal times and visibility measures for this period. The function initializes the procedure **dawn_end()** in the event array list.

dawn_end() decreases the light level by one to account for the end of dawn conditions which result in improved visibility. The function **changes()** is called to facilitate the update of traversal times and visibility for the altered conditions.

dusk() updates simulation data for dusk conditions where the light level is increased by one to account for more difficult visibility conditions. If this increase results in a light level greater than 4, searching is halted for the day due to poor visibility. In this instance the function **end_search_period()** is called; otherwise, searching continues and the function **changes()** is called to facilitate the changes in simulation input data.

nightfall() updates traversability and visibility conditions for night searching, in those situations where night searching is utilized, by calling **changes()**. Night search conditions apply from DAY_END to DAY_START and are one light level more difficult than dusk conditions. If conditions are too treacherous (equivalent to light level 4 conditions) searching is suspended until DAY_START the following day, or until weather, and hence visibility, conditions improve.

scale_down() scales down the search operation by removing x (currently set at two ground search teams) resources from the number of active resources. The resources removed are those which have the highest index labels. If the resource is currently in the field then they are redeployed to the search base at the beginning of the next search period. Redeployment is via the shortest edge path if the resource has searched for six hours or more, or via a shortest-coverage path otherwise.

scale_up() scales up the search operation by $SCALE\_EXTRA$ additional resources initializing their respective indicators, positions and clocks.

suspend_search() executes the suspension of the search operation by recalling any resources still in the field by the shortest-coverage path.

arrive_at_subject() is utilized when the stretcher retrieval of an injured subject is required. The function registers when rescue resources have arrived at the subject's location and when all rescue resources are present the subject is stretchered-out by calling the function **subject_carry_out()** (defined later in this chapter).

helicopter_arrival() is called when the helicopter arrives at the injured subject's location in order to transport them out of the area. The procedure allows 20 minutes to load the injured subject into the helicopter at which time the function **helicopter_complete_loading()** is actioned to return the search resource, who has been waiting with the subject, to the search base via the shortest available edge path.

next_event() activates the next event in the DES. The next event is labelled $event\_first$.

unresponsive_subject() alters the responsiveness state of the subject from responsive to unresponsive.

flood_regions() occurs at $flood\_time$ and records all regions susceptible to flooding as now flooded and impassable.

Floyd's shortest path algorithm is utilized to find the shortest edge paths between all vertex pairs in the TIN.

11.6 Utility Functions

The following functions are utilized within the simulation of the SAR operation in response to events that have occurred.

end_of_path() is activated when a search resource reaches the end of their allocated search path. The path covered is output to the graphics file. If a hut is located at this vertex and has not yet been searched for clues this is done now. If the resource's remaining useful search hours, the time remaining in the search period and the time remaining until a combined recall of resources, all exceed $MINLIMIT$, then further search tasks are allocated to the resource by calling the function **single_path()**. Otherwise the resource is recalled to the search base if they are not required to transport an injured subject out of the area. The resulting path is communicated to the resource.

changes() updates traversability and visibility conditions under changing weather and light levels. The procedure calculates the updated shortest paths between all vertex pairs in the TIN, redetermines the visibility classes of each edge region and reclassifies any edges that are now impassable due to flooding. If environmental conditions alter during a search period, the completion time of tasks currently under way are updated due to the changing cost structure and, if a resource is scheduled to detect either the subject or a clue while executing their current task, this detection time is also updated for the altered conditions.

11.7 Communication

Communication between the search resources and the search management team located at the search base occurs when each of the following events arise:

- a search resource detects the subject;
- management recalls a search resource to the search base;
- management redeploys a search resource, assigning them a new search path; and
- a search resource locates a clue.

Within this model communication only occurs due to these events and it is assumed that communication is possible from all points on the TIN. The time for communications will obviously differ, depending on the complexity of the information to be conveyed. We have used the arbitrary time lengths presented in Table 11.2 for the purposes of this simulation.

Table 11.2: Time durations of communication events.

Event to be communicated	Time length of the communication
Resource locates the subject	4 minutes
Management recalls a resource	2 minutes
Management redeploys a resource	5 minutes
Resource locates a clue	4 minutes

If two pieces of information are communicated at the same time the communication time is extended by *COMM_EXTRA*. When information needs to be relayed to all resources in the field this is communicated in a sequential fashion to each resource in turn. Any resources that may currently be in a communication with search management, or have a communication event scheduled to occur, are communicated with first in this scenario.

11.7.1 Search Interruptions

The act of searching is interrupted in the simulation by either a communication occurring between the search management and a resource, or by the resource stopping for a rest break. It is possible for a search resource to receive a communication during a rest break but any directives communicated will not be actioned until the conclusion of the break.

Rest breaks and communications are represented by binary indicator variables identifying whether such an interruption is active or not, and by beginning and end flags which identify the length of the interruption. The total length of an interruption needs to be calculable in order to determine the actual time a resource has spent searching over the current task, and hence the search time required to complete the search task.

The possible interruptions can be represented in time sequence as depicted in Figure 11.2.

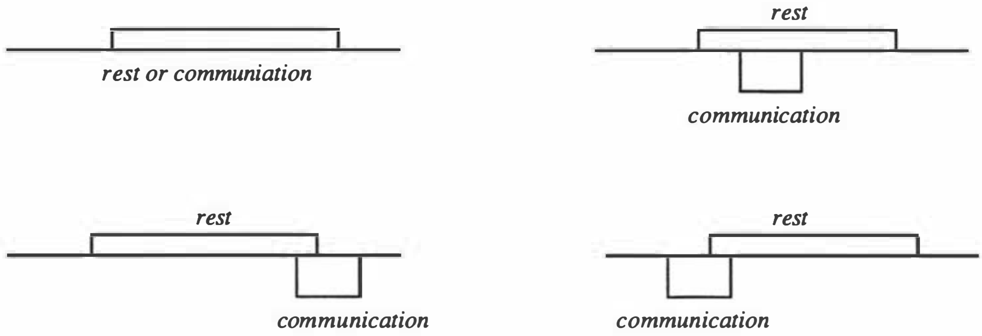

Figure 11.2: Search interruptions.

11.8 Resourcing

A search resource is replaced at the end of a search period if they have searched for six hours,[1] or their remaining useful search hours are less than $MINLIMIT$. A resource is replaced at the end of their search path if they have searched for six hours, or there are less than $MINLIMIT$ hours remaining before the end of the search period, or their remaining useful search hours are less than $MINLIMIT$. If, upon reaching the end of their path there is less than $MINLIMIT$ hours remaining before a combined recall, then the resource returns to the search base but is not replaced by a fresh resource. Resource replacement also does not occur if a recall of resources is scheduled this period. When a resource is replaced after having searched for six hours or after the subject has been detected, they are routed back to the search base via the shortest edge path. Otherwise the resource is redeployed to the base via a shortest-coverage path.

Once a resource reaches the search base they are then replaced by a fresh resource who may or may not have searched previously. While we do not explicitly keep track of this we assume that a fresh resource is one who has stood down for at least one search period and who has not searched for more than four consecutive days. An unlimited resource pool is assumed in the simulation environment, with a constant number of resources in the field at any one time. The number of resources in the field is a parameter of a particular SAR simulation.

Resources stay fixed in the same location between search periods. In the event of storm conditions, poor visibility or night conditions when night searching is not an option, there will be periods of downtime between the search periods. Hence moving during this time could be considered detrimental to the searchers' safety under the current management criteria.

[1]Six hours is considered to comprise a search resource's useful search hours.

At any given time a resource is involved in one of the following activities:

- travelling to the search base;
- being briefed at the search base;
- travelling to a search region;
- searching a search region;
- returning to the search base;
- communicating with search management via radio;
- resting through a rest break;
- travelling to reach the subject;
- waiting with an injured subject;
- executing a stretcher-carry of an injured subject; or
- waiting between periods for conditions to improve before recommencing searching.

11.9 Resource Redeployment

A search resource will be redeployed to a new position when any of the following events occur.

1. The search is suspended.
2. The search is scaled down and the resource is no longer required.
3. The search resource is replaced by a fresh resource.
4. Weather conditions deteriorate.
5. The subject has been located.
6. The resource completes their assigned search path.
7. A segment in their original search path will become impassable due to flooding at $flood\_time$.

In these instances the destination vertex of the redeployment is the search base in cases 1, 2, 3 and 4. It is also the destination in case 5, if the resource is neither the resource to detect the subject nor required for a stretcher-retrieval of the subject; otherwise, the destination is the position of the subject. In case 6, if the useful resource hours or period time remaining are not enough to allow for further search assignments, the resource is also redeployed to the search base; otherwise, the resource is assigned a new search path and the destination vertex becomes the entry vertex of the first task in the path. When case 7 occurs the resource is redeployed from the entry vertex of the segment that will flood to the exit vertex of that segment.

For all resource redeployments a path needs to be found between the resource's position at the time of redeployment and the destination vertex of that redeployment. When the subject has been detected or the resource has extinguished their useful search hours this path is the shortest available edge path from the resource's position to the destination vertex. When the subject remains undetected and the resource has useful search hours in hand, this path is a shortest-coverage path between the two vertices. The only exceptions to this are the allocation of a new search path in case 6 which is governed by search path generation heuristics, and the adjustment of a search path in case 7 which is governed by the function **flood_check()**. Deadhead traversal speeds are utilized for all redeployments.

11.9.1 Shortest-Coverage Path

The objective of a shortest-coverage path is to not only minimize the time of the redeployment but to also maximize the possibility of detecting the subject. This is perhaps best illustrated when case 4 occurs and search resources are recalled to the search base in deteriorating weather conditions. Resources are recalled due to the increasingly hazardous conditions and hence it is desirable to redeploy them in a timely fashion; yet it is just such conditions which increase the risk to the undetected subject's life and argue for the exploration of unsearched regions as part of that redeployment.

As a shortest-coverage path is initially called in response to a time-driven management decision, we have developed a heuristic to determine a path that is primarily weighted towards a shortest time path. The time length of the path is constrained by a redeployment path limit that is defined with consideration to the reason of redeployment, the current urgency level, and the remaining useful search hours of the resource to be redeployed. In particular, the path duration limit is defined in Table 11.3 and is effected by the condition resulting in the smallest duration limit.

Table 11.3: Redeployment path duration limits.

Condition	Path Duration Limit
Redeployment in adverse conditions	$flood\_time - day\_clock$
Urgency level of one	remaining useful search hours + 2 hours
Urgency level of two or three	remaining useful search hours + 1 hour

While these duration limits are arbitrary, the justifications behind them are: in conditions hazardous to resources their effectiveness is reduced by the conditions, fatigue sets in earlier, and a greater risk of injury is entertained; that fatigue will set in for a resource once they complete their useful search hours and searching much longer that this will be ineffective. If conditions are not hazardous then additional effort is made to locate the subject in situations of highest urgency, with the duration limit equal to the number of useful search hours plus 2 hours.

The shortest-coverage path heuristic finds the shortest edge path from the resource's starting position, vertex a, to the redirection vertex, vertex b. An edge path was selected over a path traversing both edge and triangular regions, primarily for ease of navigation. The heuristic first finds the shortest path between vertices a and b which consists only of those edges not yet searched,[2] if such a path exists. If the time length of this path falls within the path duration limit then this path is the redeployment path selected.

If the path exceeds the time duration limit or does not exist, a second type of path is found which may include searched and non-searched edge regions. This path is found by first transforming the edge costs to reflect whether or not an edge has been searched during the operation. Edges yet to be searched have their traversal times temporarily reduced by a factor to weight them more favourably for inclusion. Dijkstra's algorithm is then used to determine the shortest path between vertices a and b over this adjusted cost network. The weight by which the unsearched edge costs are multiplied is recursively decreased, starting at one half the edge weight and reducing to one tenth the edge weight, until the composite path generated exceeds the redeployment duration limit. In this way the final composite path selected is that which is within the path limit, if such a one exists, and having the greatest number of unsearched edge components.

If neither of these generated paths have an unadjusted time duration which falls within the redeployment path limit, the resource is redeployed from vertex a to vertex b via the shortest available edge path, irrespective of the composition of that path (in searched or non-searched edges) and its final length (which will be shorter than either of the two previously generated cover paths). The heuristic method is described in detail

[2]This subset does not include untracked edge regions which are not required to be searched.

in Algorithm 11.8.

11.9.2 Modelling the Redeployment of Resources

If a resource is redeployed whilst searching a triangular region, the resource's position within the region at the time of redeployment needs to be approximated. This is because the level of detail used to model the actual movement of resources is not fine enough to determine a resource's exact location within a triangular region at any given time. The straight-line distance, *dist*, between the resource's approximated position and the desired exit vertex for the redirection, *exit*, is approximated as:

$$dist = (1 - fraction) \times edge\_length(entry, exit) \times 1.2 \quad \text{if } exit \neq entry$$
$$dist = fraction \times edge\_length(entry, exit_0) \times 1.2 \quad \text{if } exit = entry$$

where $exit_0$ is the vertex that the resource was initially to complete the task at, and *entry* is the vertex the resource entered the region at. The fraction of the task completed before the redeployment occurs is represented by *fraction*. The multiplication factor of 1.2 is an approximation of the depth of strip searching into the region interior. Such a redeployment is illustrated in Figure 11.3.

The speed approximation utilized for searcher redeployment from within a triangular region is the quickest deadhead speed to the exit vertex when considering the gradient from either of the other two region vertices.

11.9.3 Recall of Search Resources

The function **return_to_base()** (detailed in Algorithm 11.9) redeploys a resource from their current position to the search base via a shortest-covering path. When the resource is searching a region which has a *POD_cum* value less than 1.0, the search is completed if the time to achieve this and return to the base is within the redeployment duration limit. If these conditions do not hold and the resource is in the middle of a task, this task is interrupted and the best redeployment path is found, by constructing a path exiting from each vertex of the region. The path selected is that which has total length less than the redeployment limit and greatest predicted POS return. In instances of ties, the tie is broken by selecting the path which contains the greater number of unsearched regions and in instances where this number is also equal, selecting the path of least time duration. If all path options exceed the redeployment limit the path of least time duration is selected.

11.9.4 Flooding

If weather conditions reach level 4 those edge regions which are 'flagged' as susceptible to flooding will flood two hours after these conditions are attained and become impassable

Algorithm 11.8 **function** *redirect*$(a, b, resource)$

for $i = 0$ **to** $nedges$ **do**
  if *edge i will flood before end of search period* **then**
  $cost_i \leftarrow INF$
end
$redeploy\_limit \leftarrow$ time duration maximum less time to reach a
for $i = 0$ **to** $nedges$ **do**
  if *edge i has been searched* **or** *is an untracked region* **then**
  $tempcost_i \leftarrow INF$
  else
  $tempcost_i \leftarrow cost_i$
end
// determine shortest path from a to b over unsearched edges
$path \leftarrow Dijkstra(a, b, tempcost)$
if *path exists* **and** *path duration* $\leq redeploy\_limit$ **then**
return$(path)$
else if (*length of shortest path from a to b over cost matrix* $\leq redeploy\_limit$) **then**
  // generate redeployment path of mixed edge composition
  $weight \leftarrow 2$
  while (actual path duration $< redeploy\_limit$ **and** $weight \leq 10$
  and path exists) **do**
    for $i = 0$ **to** $nedges$ **do**
      if *edge i not searched* **and** *not untracked* **then**
      $tempcost_i \leftarrow \dfrac{cost_i}{weight}$
      else
      $tempcost_i \leftarrow cost_i$
    end
    $temppath \leftarrow Dijkstra(a, b, tempcost)$
    if *actual path duration* $\leq redeploy\_limit$ **and** *number of unsearched edges in path exceeds number in previous composite paths* **then**
    $path \leftarrow temppath$
    $weight \leftarrow weight + 1$
  end
  if *unsearched* **and** *composite paths not within* $redeploy\_limit$ **then**
    // generate shortest edge path from a to b over unadjusted costs
    $path \leftarrow Dijkstra(a, b, cost)$
  end
  return$(path)$
end

end

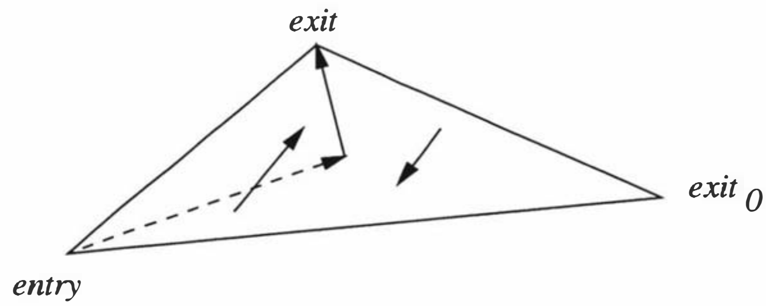

Figure 11.3: Redeployment from within a triangular region.

to ground search resources [3]. The time that these regions will flood is recorded by the variable *flood_time*. If flooding is simulated to occur within the current search period ($flood\_time < period\_end$) no edges with a flood indicator equal to one are considered as candidates for allocation to resources, either as search tasks or components of access paths to other regions. This strategy can be viewed as a conservative risk management policy. To ensure that these regions are not considered as viable routing options their traversal costs are temporarily updated to a prohibitively high cost (∞) to avoid selection. At *flood_time* the edges susceptible to flooding are recorded as flooded with traversal costs equal to ∞. The status of these edges is arbitrarily reversed three days after an improvement to weather level 3 conditions.

Within the simulation model of the SAR operation it is possible that a prediction of level 4 weather conditions may arise after search paths have been allocated for the period and arrive in actuality before resources have completed their search paths. In this instance a recall of resources is set in place to occur prior to flooding. However, if a resource is already en route to the search base or to the subject, they are not affected by this recall and it becomes necessary to re-route such resources around any edge regions in their search paths that will flood. This is achieved by activating the function **flood_check()** at the time this prediction occurs. As the proportion of allocated edge regions that will flood is likely to be small, as is the number of resources likely to be affected, we have selected a myopic approach to re-routing these resources.

The function **flood_check()**, of Algorithm 11.10, investigates all edge regions in the uncompleted portion of a resource's path for possible flooding. If a segment of the path is identified as susceptible to flooding (a segment may consist of a single edge or a connected subset of edges that will all flood) it is individually replaced by a new path segment containing no regions in danger of flooding. The new segment has its beginning

[3] We currently model 50% of those edges representing streams and rivers as candidates for flooding, selecting those edges representing the features with greatest water volume.

Algorithm 11.9 function *return_to_base*(*resource*)

```
    redeploy_limit ← time duration maximum
    if resource has reached the end of their assigned path then
        vertex ← vertex resource positioned at
        path ← redirect(vertex, base, resource)
    end
    else
        rgn ← region resource is located on
        time ← time to complete search of rgn + time to return to base via shortest path
        if POD_cum,rgn < 1.0 and time < redeploy_limit then
            complete search of region
            vertex ← exit vertex of region
            path ← redirect(vertex, base, resource)
        end
        else
            // interrupt task
            for each exit vertex of rgn do
                a ← exit vertex
                path_a ← redirect(a, base, resource)
                totaltime_a ← time to exit vertex a + time to traverse path_a
                if (totaltime_a ≤ redeploy_limit) then
                POS_a ← POS result of redeployment path
            end
            if all redeployment paths exceed redeploy_limit then
            path ← redeployment path of smallest time duration
            else
                if best POS result is shared by more than one exit vertex path then
                    if more then one path comprises the greatest number of
                    unsearched edges then
                    path ← redeployment path of least time duration
                    else
                    path ← redeployment path comprising the greater number of
                    unsearched edges
                end
                else
                path ← redeployment path returning highest POS
            end
        end
    end
    return(path)

end
```

and end vertices as the beginning and end vertices of the segment that will flood and comprises the shortest available edge path. If it is not possible to find an edge path to redirect the resource through, such as in a case where all adjacent edges will also flood, the resource is redirected through adjacent triangular regions, selecting the quickest path.

The shortest edge path connecting the end vertices of the flooded segment was selected over a shortest-covering path connection, as the additional time length of the path would take search time away from the future tasks already allocated to the resource. In particular, these future tasks were considered to be of higher priority for searching (with respect to the path generation strategy), to be initially selected than the edge tasks that would be selected in a shortest-covering path.

If the instance arises where a resource is located on an edge that will flood at $flood\_time$ and the resource cannot complete the traversal of the region by this time, then it is necessary to re-route the resource so that they will not be located on the edge at $flood\_time$. To this end the function returns the resource along the edge in the direction which they came or, if this vertex cannot be reached in time, re-routes the resource into the adjacent triangular region with the least dense vegetation. If the resource is about to enter an edge region which will flood before $flood\_time$ and the region comprises the only region in the resource's path, or all regions in the allocated path will flood, a new path is allocated to the search resource.

11.10 Subject Detection

If a resource searches the *lost_region* and detection occurs, the simulation model calculates the position and time at which this occurs, labelling the locating resource, $find\_team$. After determining the condition of the subject the positions of the other search resources in the field are determined. These resources are communicated with by search management to inform them of the successful conclusion of the search together with further redeployment instructions. These resources will then either return to the search base or will be redeployed, by the shortest available edge paths, to the location of the subject in order to assist with transporting an injured subject when helicopter access is prohibitive. At this time all traversal costs are reduced to their non-searching (deadhead) times *i.e.*, the resources move more quickly over the terrain. However, if a stretcher retrieval is required, the traversal costs are again augmented to approximate the greater traversal difficulty involved, as outlined in Section 11.11.2. The procedures followed upon subject detection are detailed in Algorithm 11.11.

Algorithm 11.10 function *flood_check*(*resource*)

pstn ← current position of *resource* on search path
a ← start vertex of current region
if *resource is located on an edge susceptible to flooding* **then**
  if *time to complete task* > *flood_time* **then**
    // *need to re-route resource so not on edge when it floods*
    identify connected edge segment (a, b) in *resource*'s path which will flood in entirety from vertex *a* at *flood_time*
    if *positioned at beginning vertex of region* **then**
      if *all tasks in path of resource are in segment* (a, b) **then**
      assign a new path to *resource*
      else
      shortest_path(a, b)
    end
    else if (*time to return to a* < *flood_time*) **then**
      return back along edge to vertex a
      shortest_path(a, b)
    end
    else
    re-route *resource* into neighbouring triangular region with least density
  end
end
if *remaining search path not replaced* **then**
  for $j = pstn$ **to** num_i **do**
    if *region is an edge* **and** *edge* $(path_{i,j}, path_{i,j+1})$ *will flood* **then**
      save copy of unchanged path to this point
      a ← vertex at position *j* in *resource*'s path
      identify connected edge segment (a, b) in *resource*'s path which will flood in entirety from vertex *a* at *flood_time*
      shortest_path(a, b)
    end
  end
end
if *current search path is altered* **then**
  reinitialize search path for *resource*
  communicate altered path to *resource*
end

end

Algorithm 11.11 **function** *subject_located*()

```
    find_team communicates to base that the subject has been detected
    if subject not injured then
        find_team returns to base with the subject via shortest available edge path
        for i = 0 to resources do
            if resource i not at base or returning to base and i ≠ find_team then
            resource i returns to base via shortest available edge path
        end
    end
    else if (weather conditions are viable for helicopter retrieval or weather conditions
     due to improve before stretcher retrieval can be executed ) then
            helicopter rescue of subject
            find_team returns to base via shortest available edge path
        end
        else
            // stretcher retrieval of subject
            num ← rescuers_required()
            rescuers ← (num − 1) closest resources to the subject
            for i = 0 to num − 1 do
                t ← rescuers_i
                path_t ← shortest path to subject from pstn_t to pstn_subject
                communicate redeployment path to resource t
            end
            for i = 0 to resources do
                if resource i not in rescuers then
                    recall resource i to base via shortest path
                end
            end
        end
    end

end
```

11.11 Subject Rescue

If the subject is injured or a fatality, the subject is transported out of the search region either by helicopter (if conditions allow), or by stretcher retrieval. Weather is considered viable for a helicopter rescue of the injured subject if the weather conditions are equal to level 1 or level 2. If the weather level equals 3 then the conditions are considered to be viable 50% of the time. It is assumed in the simulation that there are no limits on the helicopter's flight distance.

Currently when a helicopter is used for subject-recovery it is assumed that any position on the TIN is accessible by the helicopter. If this does not hold then it would be necessary to stretcher the subject to the closest point accessible by the helicopter.

If a helicopter rescue is initiated then all resources in the field, apart from *find_team*, are recalled to the search base via the shortest available edge path. The SAR simulation generates the arrival of the helicopter at the subject's position after their detection and condition is notified to the base. A speed of 50km/hr is utilized plus 20 minutes preparation time. It is assumed that the helicopter flies to the subject's location from the search base. Twenty minutes are then allocated to stabilize and load the injured subject into the helicopter before flying on. The detecting resource, $find\_team$, then also returns to the search base by the shortest available edge path.

11.11.1 Redeployment of Resources to the Subject

When the subject is injured or unconscious and a helicopter transferral is prohibitive under the circumstances, stretcher retrieval is required from the subject's location to the search base. (Alternatively this could be the closest accessible point on the exterior of the TIN.) The heuristic **rescuers_required()** (detailed in Algorithm 11.12) determines the number of search resources needed to carry the subject over the distance to the search base. If the stretcher carry is a short carry (we assume this to require less than one hour), four people per subject are required. A medium carry (up to four hours) requires 16 people per subject, while a long carry (exceeding this 4 hour limit) requires two teams, each of 16 people. If the number of resources required is greater than the number available, the operation is scaled-up by the additional number of personnel. These additional resources are allocated shortest paths leading directly to the location of the subject.

The time required to execute the stretcher carry is approximated by the shortest path from the subject's position to the search base multiplied by a factor, $STRETCHER\_FACTOR$. This multiplicative factor is used to approximate the additional time needed in carrying an injured person. Additionally the shortest available path may not be the easiest path for the stretcher teams to traverse.

The closest resources available for the stretcher carry, up to the number of rescue resources needed, are redeployed to the subject's location from their current positions via the shortest available edge path. The edges comprising this path are communicated to each rescue resource. Any other resources remaining in the field are recalled to the search base by the shortest available edge path.

When a resource is redirected to the subject's location for the purpose of stretcher-carrying the subject out of the region, the shortest overall distance is determined from the resource's current location to the subject's position. If the subject is located in a triangular region the straight line distance from the entry vertex of the region to the subject's coordinates is calculated and the speed utilized is the quickest speed over the

Algorithm 11.12 function *rescuers_required()*

```
    // determine the number of search teams needed to stretcher carry subject
    time_to_base ← stpath_{pstn_subject,base} × STRETCHER_FACTOR
    if (time_to_base < 1) then
    num ← 1
    else if (time_to_base < 4) then
        if (TEAMSIZE = 4) then  num ← 4
        if (TEAMSIZE = 6) then  num ← 3
        if (TEAMSIZE = 8) then  num ← 2
    end
    else
        if (TEAMSIZE = 4) then  num ← 8
        if (TEAMSIZE = 6) then  num ← 6
        if (TEAMSIZE = 8) then  num ← 4
    end
    if (num > resources) then
        extra ← num − resources
        scale_up(extra)
    end
    return(num)

end
```

region as approximated by the gradient to either of the non-entry vertices.

11.11.2 Stretcher Retrieval of a Subject

A path is desired which, while still transferring the subject from their location to the search base quickly, also does this by the 'smoothest' possible path (heuristically) to avoid undue discomfort to the subject, which could compound their injuries, whilst preserving the energy of their rescuers. The **gradient-shortest path** is used to describe such an edge path between any two vertices of the TIN.

11.11.2.1 Gradient-Shortest Path Determination

The most desirable path that the stretcher carry is to cover is the one which covers easier terrain, both in the sense of vegetation cover and the type of regions being traversed, including whether the traversal is over level terrain, or in an uphill or downhill direction. The heuristic **subject_carry_out()**, of Algorithm 11.13, is developed to adjust the cost structure of the TIN to reflect terrain desirability. Hence those regions with easier terrain and gradient are reduced by a significant factor, whereas regions of less desirable terrain are reduced by a smaller factor, if reduced at all. Traversal costs are initially multiplied by a factor of $STRETCHER\_FACTOR$ to account for slower traversal.

Algorithm 11.13 function *subject_carry_out*()

```
for i = 0 to n do
    for j = 0 to n do
        cost_ij ← cost_ij × STRETCHER_FACTOR
        if (terrain = 1) then
        cost_ij ← cost_ij/6
        if (terrain = 2) then
        cost_ij ← cost_ij/3
        if (gradient < 0.5) then
        cost_ij ← cost_ij/6
        if (gradient < 1.0) then
        cost_ij ← cost_ij/4
        if (gradient < 1.5) then
        cost_ij ← cost_ij/2
        if level terrain then
        cost_ij ← cost_ij/4
        if downhill terrain then
        cost_ij ← cost_ij/2
    end
end
best ← INF
// select exit vertex of lost_ region which results in best path to the search base
    a ← exit vertex
    time_a ← adjusted time from pstn_subject to vertex a
    time_a ← time_a + Dijkstra_gradient(a, base)
    if (time_a < best) then
    path ← gradient_shortest path from pstn_subject to base via a
end

end
```

The adjusted cost structure is then input into an adapted version of Dijkstra's shortest path algorithm. Within the algorithm the cost structure is altered when considering the augmentation of the path. The gradient change between the preceding edge in the path and the current edge is calculated and the factor by which the path cost is reduced is related to this gradient change as given in Table 11.4. These factors are utilized to reflect a desirable gradient change between edges along the path that the stretcher is carried. Hence where the gradient change is small, the path is favoured more heavily.

Table 11.4: Path cost factors for gradient changes between edge traversals.

Gradient Change	Factor
< 0.5	4
< 1.0	2
≥ 1.0	1

Dijkstra's shortest path algorithm is of complexity $O(n^2)$ (as is the adapted version) and code for the algorithm and a detailed description can be found in [160] Section 3.3.1.

Having developed rules and heuristics to simulate each phase of a SAR operation, the simulation model is complete. We now use this model to simulate a search over three representative search scenarios, drawing some preliminiary conclusions on the relative performance of a restricted number of resource allocation methods and search strategies.

CHAPTER 12

Simulation Experimentation

12.1 Introduction

Having implemented the simulation model it remains to alter the parameters and data input into that model in order to test different search strategies. The intention of this chapter is to provide some preliminary analysis of resource allocation techniques applied over varying search scenarios. As there are many possible combinations of search scenarios and simulation parameters that could be tested, the objective of this computational experimentation is to provide some insights into the relative performance of the four path generation methods described in Section 8.8.5 over a limited number of scenarios. In particular, as the SAR problem is inherently dynamic, high levels of control are needed to identify differences between search scenarios, and a large number of experiments would be required to discern the best level at which to fix particular simulation parameters. In light of this, the analysis performed is only preliminary, with the intention of identifying both expected results and unexpected results which could be investigated in further research.

We now describe the process developed to generate the TIN on which the experiments were conducted.

12.2 TIN Generation

Constructing TIN models that accurately model real terrain data requires specific skills in sampling initial data points and forming a triangulation whose approximated surface falls within a specified tolerance level [106]. As these skills extend beyond the scope of this thesis an artificial computer-generated TIN is used in these precursory experiments. Such a TIN is constructed by generating point locations and elevations, and then applying the TIN construction algorithm of Macedonio and Pareschi [110] over these points. Possible approaches to generating point locations and elevations include the following:

- randomly generate x, y and z coordinates for n points, with each coordinate being independently and uniformly distributed over user-specified ranges;
- generate n points located at each vertex of a lattice, then randomly generate elevations for each point over a user-defined range;
- model mountain ranges and valleys by linear segments and generate clusters of points around these, modelling correlated elevations by generating z coordinates over tightly defined ranges; and
- form a grid of user-defined dimensions, then randomly generate points over this grid using a uniform distribution such that a maximum of m points are generated within each grid square, where $m < n$, then define an elevation range for each grid square and randomly generate z coordinates for each point via a uniform distribution.

The fourth method models the correlation of elevation points among spatially adjacent points that is present in reality, by defining elevation ranges, modelled by a uniform distribution, over each grid square. The method has the flexibility of allowing a greater number of grid squares to be used to achieve greater control over the coordinate values generated. The maximum number of points generated within each grid can also be separately defined for each grid to model the heavier sampling that would occur within some grid squares if those grid squares were characterized by changeable terrain.

12.2.1 Experimental TIN

We applied the fourth construction method to construct a TIN of 30 points on which to conduct the experiments. A 3×3 grid was arbitrarily selected, with the dimension of each grid square being 1 km by 1 km. A maximum of 10 points were generated within each grid square. The z-coordinate range over each grid square is depicted in Figure 12.1.

12.2.2 Vegetation Allocation

Triangles have their vegetation classification determined by the vegetation of the grid square in which the triangle predominantly falls, *i.e.*, the grid cell of 'maximum overlap'. This grid square is determined by investigating which column and row of the grid each vertex of the triangle falls in. If two or three vertices of the triangle share a common column (or row) then the grid square which the triangle is allocated to is determined from

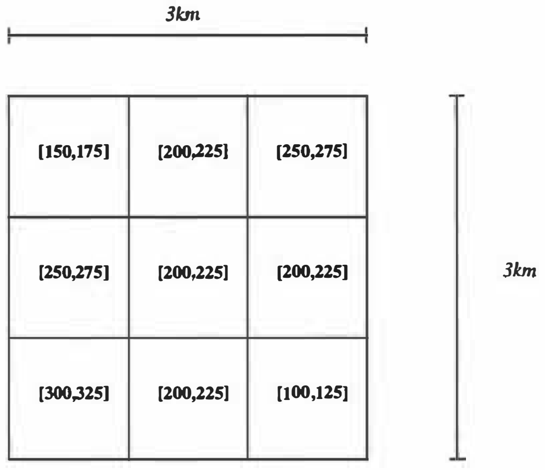

Figure 12.1: The range of z-coordinates (in m) for each grid square of the TIN construction grid.

this column (or row). When each vertex of the triangle falls in a different column (or row), the grid square is determined from the middle column (or row) of the three. The triangle is allocated to the grid square that is described by the selected column and row. This grid square allocation process is illustrated in Figure 12.2 for a triangle spanning several grid squares. The column shared by two vertices of the triangle is column $x2$, while row $y2$ is the central row of $y1, y2$ and $y3$; hence the triangle is allocated to the shaded grid square $(x2, y2)$.

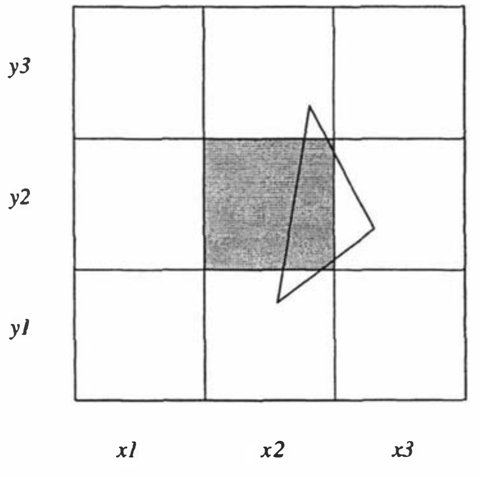

Figure 12.2: Allocating a triangle spanning several grid squares, to a single grid square.

The vegetation over each square of the grid can be input manually or it can be determined by an algorithm. We have developed an algorithmic approach to vegetation

Algorithm 12.1 function *vegetation_allocation*()

> set limits on the total number of grid squares to be allocated each type of terrain classification
> **for** $i = 0$ **to** number of grid squares **do**
> $veg_i \leftarrow -1$
> order triangles in increasing order of area covered
> *type* $\leftarrow$ terrain classification of greatest density
> **while** (*grid squares remain with unallocated terrain classification*) **do**
>   *grid* $\leftarrow$ grid square that next triangle in ordered list is allocated to
>   **if** ($veg_{grid} = -1$) **then**
> $veg_{grid} \leftarrow type$
> $num_{type} \leftarrow num_{type} + 1$
>   **end**
>   **if** num_{type} *equals the grid square limit for the terrain type* **then**
>   *type* $\leftarrow$ terrain classification of next greatest density
> **end**

end

Table 12.1: The percentage limit of the number of grid squares classified by each terrain type.

Terrain Classification	1	2	3	4
% limit	0	50	30	20

allocation which allocates denser vegetation to those grid squares containing triangles of smaller area. Specifically the algorithm follows the steps outlined in Algorithm 12.1.

The algorithm allocates denser vegetation to triangular regions with smaller area, as such vegetation impedes search progress and visibility resulting in greater search duration to attain POD levels equivalent to those resulting from searches over larger areas of less dense vegetation. By correlating thicker bush cover to smaller search regions and sparse bush cover to larger areas, further partitioning of these regions (such that a minimum POD level can be achieved within 6 hours) can often be avoided. To ensure consistency between the type of bush classification allocated to adjacent triangles, vegetation is allocated to squares of the construction grid rather than to individual triangles in increasing order of area. If vegetation were manually entered for each grid square, a similar result could be achieved by generating a larger number of vertices over the grid squares with denser bush cover and generating fewer vertices over the squares having sparse bush cover. The particular limits governing the terrain classification in the experimental TIN are outlined in Table 12.1.

After triangles have been classified by terrain, any regions that cannot be visually searched to a 50% POD level within 6 hours, in ideal conditions (taking the minimum time traversal direction), are further partitioned as described in Section 8.5.5. Further partitioning of the experimental TIN is required, resulting in ten additional vertices; hence, the final number of vertices comprising the TIN is forty.

12.2.3 Edge Types

The edge types described in Section 4.3 are computed on a given distribution with additional restrictions being placed upon those edges which might be considered suitable candidates for each type. To this end, edges are defined by type with the most restrictive being computed first.

The most restrictive edges are the ridge edges. *Ridge edges* are assigned to those edges having the highest elevation and are linked to one another to form a subnetwork of edges, as would normally be encountered. *Water networks* are initially rooted on those edges with the lowest elevation and are then formed into linked components with a least two other like edges. *Tracked edges* are then defined, where at least one track edge has a vertex on the convex hull of the TIN, to form an accessible link with the roadways of the area. Tracked edges are also connected to one another to form a subnetwork within the TIN. *Untracked edges* are then defined as those remaining edges not yet labelled. The terrain density of each edge is determined individually for streams and tracks, and is calculated for ridge and untracked edges as the average of the terrain classification of the triangles adjacent to each edge.

12.2.4 Experimental TIN

The TIN on which the computational experiments are conducted comprises 40 vertices and 177 regions, and is illustrated in Figure 12.3. The set of regions is made up of 69 triangles and 108 edges with the edge set consisting of: 6 ridges, 18 streams, 14 tracks and 70 untracked edges. The terrain classification of each region type is given in Table 12.2. The mix and dispersion of vegetation types are illustrated in Figure 12.3, and the type and contiguity of edge types are illustrated in Figure 12.4. Two huts are located at vertices 18 and 20 and no hazard features are considered. The z-coordinates of the vertices cover a range of 203m, from 113m to 316m. The total area of the triangles is equal to 7.17 km^2 and the edge lengths sum to 65.02 km. The average vertex degree in the TIN is five.

Of the 18 edges representing streams and rivers, half are described as being susceptible to flooding in adverse weather conditions. The edges so described are those which have the higher density classifications, edges{(1,26), (5,23), (15,23), (15,25), (15,26), (17,25),

Table 12.2: The number of regions classified by terrain type.

Region	Terrain Classification			
	1	2	3	4
Triangles	0	6	26	37
Stream edges	0	9	9	0
Track edges	4	6	4	0

(21,26), (22,19), (22,29)}.

In ideal weather and light conditions the total time to traverse all edges (averaging the traversal time in each edge direction) is 98.85 hours, giving an average for an edge of 55 minutes. The total time to visually sweep all triangles to a 50% POD level in ideal conditions (taking the minimum time traversal direction) is 167.40 hours, giving an average for a triangle of 2.43 hours. The total time needed by one search team to traverse all regions comprising the TIN, in ideal search conditions, is therefore 266.25 hours.

The maximum time to complete an edge traversal (in the shortest direction) is 3.07 hours for ridge edge (2,27) that has a terrain density classification of 4. The maximum time required to complete a visual sweep of a triangle to achieve a 50% POD (in the shortest direction) is 5.97 hours for triangle (2,13,30) that has a terrain density classification of 4.

12.2.5 POA Distributions

An historical distribution of POA values is modelled on the TIN via a 'seeding' procedure. This procedure seeks to mimic the localized nature of high POA values in those instances when lost people tend to be found in concentrated areas within a given terrain map. This may be due to the terrain, including existing hazards, or the type of recreation activity most commonly performed in that area. The method 'seeds' a high POA value of 20% at a set number of locations (*seed_num*) within the TIN (these regions are randomly selected). The regions adjacent to these seed regions are also allocated relatively high POA values. The particular POA value of any region adjacent to a seed region is randomly generated over a uniform distribution of range [5,19] percent. All other regions in the TIN are assigned a POA value of 0.0 and POA values are normalized to sum to one.

Edges of the TIN which represent untracked features are considered only in their capacity as delineators between triangular regions. Hence these regions are assigned null POA values and are not allocated to resources for searching. It is also assumed for the purposes of simulation that the subject is located within the regions comprising the TIN and hence the POA value of the ROW region ($POAROW$) is set equal to 0.0.

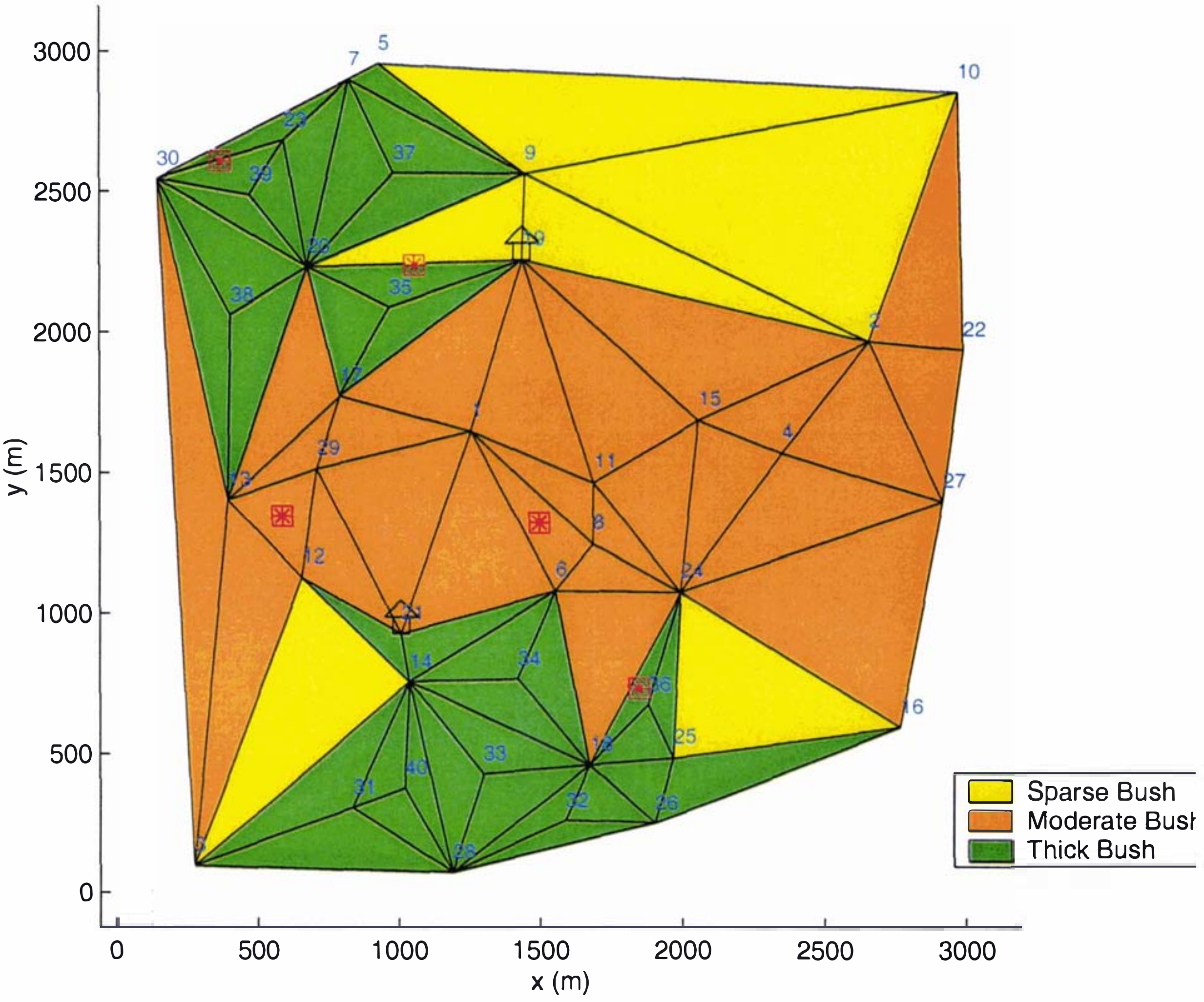

Figure 12.3: The TIN used in experimental computation.

The particular POA distribution of the TIN used in the computational experimentation is generated by five randomly selected seed regions: edges (22,29) and (18,19), and triangles (0,5,7), (11,12,28) and (17,23,35). These seed regions are depicted in Figure 12.3 by an '$*$'.

12.3 SAR Problem Instance

A SAR problem instance describes a particular SAR scenario and is defined by the inputs to the SAR simulation described in Section 10.5. In addition to the TIN described above, on which all the experiments are conducted, the following inputs are used.

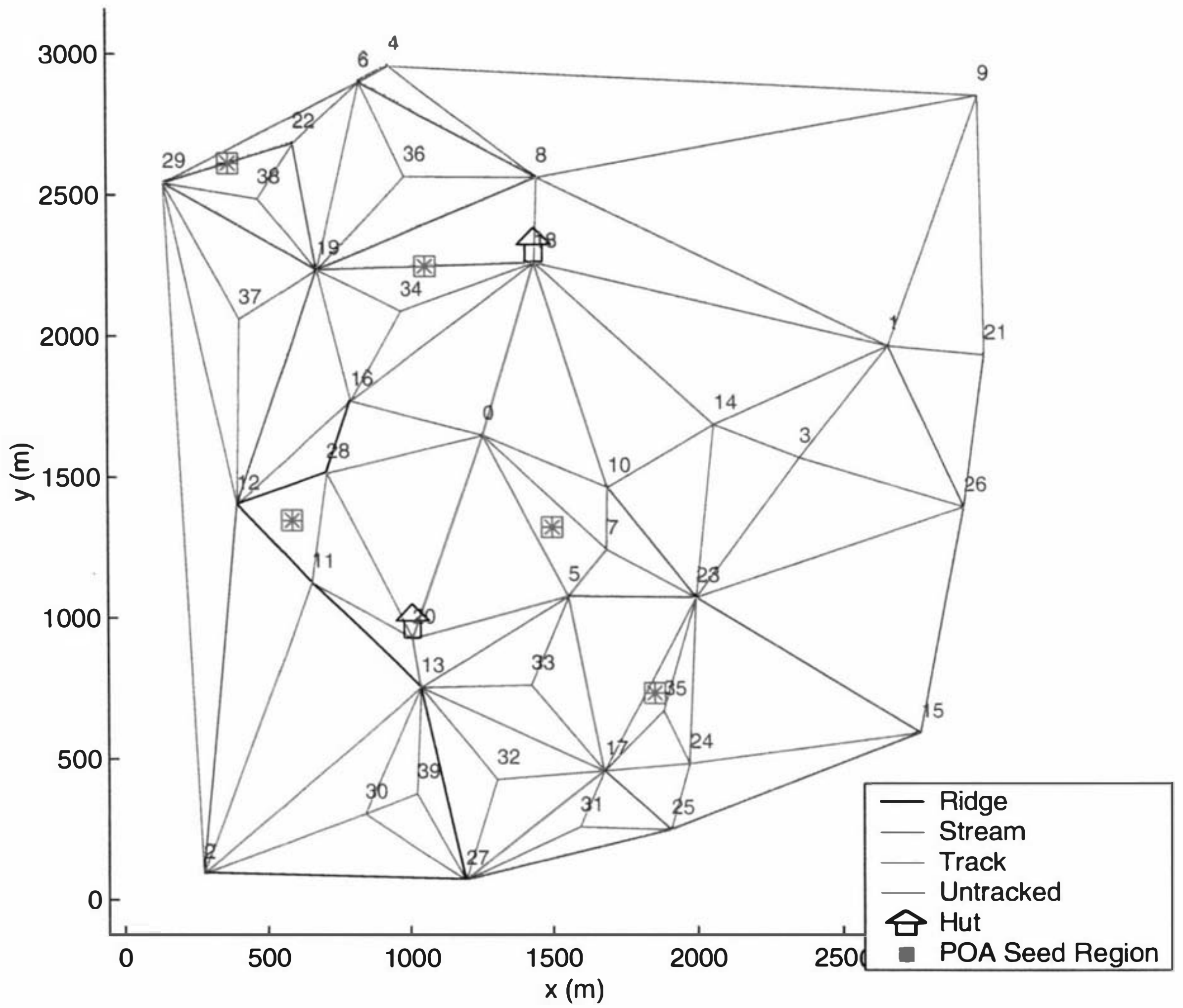

Figure 12.4: The edge classifications of the TIN used in experimental computation.

12.3.1 Subject Path and Clue Placement

The path taken by the subject is determined as described in Section 5.2. The resulting path can be described by:

- the decision process guiding which region is entered at each decision point along the path;
- the starting vertex;
- the resulting time length;
- the number of clues laid;

- the number, type and distinctiveness (whether a region is passed through more than once) of the regions that the path moves through;

- the type of region that the subject is finally positioned within and the type of vegetation that covers that region;

- the area covered by the region in which the subject is located and the subsequent time required to search the region to a given POD level under differing search conditions;

- the initial POA values attached to the regions along the subject's path (this will impact upon whether these regions fall within the primary search area, when this is defined); and

- whether the intended path is known to search management or not, and whether or not this was actually followed.

Within the experiments conducted 50 clues are laid for each problem instance and the final position of the subject is set by $LOST\_FRACTION = 0.5$. The other parameters governing the construction of the subject path are varied across experiments to investigate differences arising from such things as the type of region that the subject is located within, and whether or not the subject moved in accordance to the initial subject location probability distribution.

12.3.2 Subject Status and Detection

The subject is regarded as stationary in all experiments with responsiveness being varied to test the utilization of the sound sweep searching method. Responsiveness is affected both by the parameter setting depicting the subject's initial condition, and then by search duration and weather conditions, with responsiveness under these conditions being determined from the pseudo random number stream:

71 0 5 16 43 71 23 19 63 40

This stream is used in all experiments where the subject is initially responsive.

A stream of pseudo random numbers is used to determine subject detection. The particular stream used in each experiment is:

19 85 90 73 31 30 41 0 93 81

The first number in the stream is 19; hence, if the lost region is completely searched the first time that it is allocated for searching, with a POD greater than 19%, detection of the subject will result. However, if the first search of the region is interrupted due to resource redeployment, the region may need to be searched several times before subject detection will result, depending on the POD level of each subsequent search.

The same random number streams are used to determine subject responsiveness and detection in each of these preliminary experiments in order to avoid the multiplicity of runs which would otherwise result, thus enabling a straight forward comparison of the four resource allocation methods when varying selection criteria, search strategies and weather conditions.

12.3.3 Weather Scenarios

Two weather scenarios are used in the experiments. Each scenario describes the initial weather conditions and the times at which these conditions subsequently improve or deteriorate. The first weather scenario commences with weather conditions of level 2 and improves before deteriorating four days into the search. Resources are recalled at time 160.00 and the flooding of those regions susceptible to flooding occurs at time 167.54. When weather scenario one is used an initially responsive subject becomes unresponsive at time 96.00. The second weather scenario is more changeable, beginning with level 3 weather conditions and deteriorating quickly to level 4 conditions. Resources are recalled at time 14.54 and again at 85.54. The flooding of those regions susceptible to flooding occurs at 18.54 and an initially responsive subject becomes unresponsive at the onset of level 4 conditions at time 16.54. These weather patterns are illustrated in Figure 12.5 and Figure 12.6.

12.3.4 Simulation Parameters

In order to study a subset of the factors governing search resource allocation some simulation and method parameters are fixed at some level over all experiments. Of the set of simulation parameters defined in Section 10.5.1 and method parameters detailed in Section 8.8.2, the following are fixed over all experimental runs:

- the starting search urgency level ($START\_URGENCY$) = 3 (low urgency);
- the length of planning period ($PERIOD$) = 6 hours;
- the time of day that the operation commences (DAY_START) = 6am;
- the length of a rest break ($REST\_STOP$) = 5 minutes;
- the duration of briefing time at the base ($BRIEF\_TIME$) = 15 minutes;
- the additional communication time when two pieces of information are communicated at the same time ($COMM\_EXTRA$) = 2 minutes;

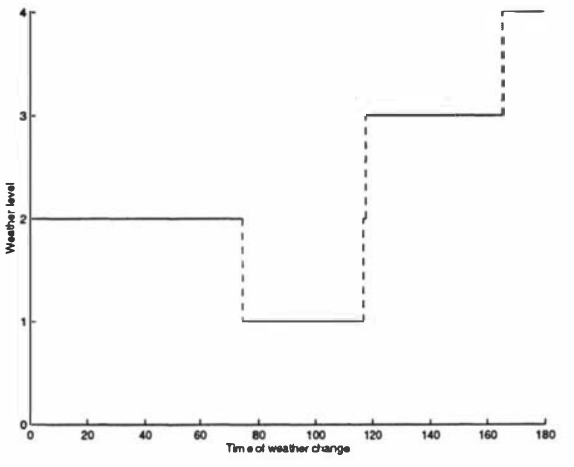

Figure 12.5: Simulated weather conditions under weather scenario one.

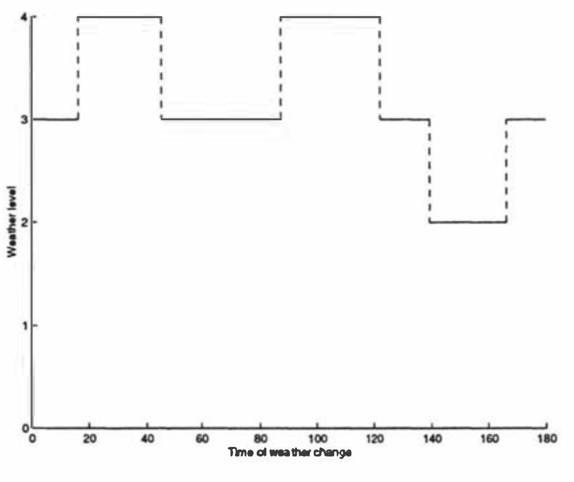

Figure 12.6: Simulated weather conditions under weather scenario two.

- the number of days until search suspension is considered ($SUSPEND$) = 7 with the number of days which must have passed since the first clue was discovered ($CLUE\_DAYS$) = 2;
- the number of days without detection before the operation is scaled down ($SCALE\_DOWN$) = 4;
- the initial ROC value (ROC_0) = 0.40;
- the factor by which traversal costs are multiplied when a stretcher retrieval of the victim is required ($STRETCHER\_FACTOR$) = 3;
- fixed target POD levels are utilized each search period, beginning at ($STARTPOD$) 50% and being incremented ($POD\_INCREMENT$) by 5% each successive period to model search intensification as the operation progresses, in a similar manner to the increasing search thoroughness of the three search phases currently used in land SAR;
- the minimum POD level that must be achieved in a search ($MINPOD$) is set at 20% and the maximum POD level of a search of a triangular region ($PODLIMIT$) = 90%;
- the minimum time that must be available before search tasks are allocated to resources ($MINLIMIT$) = 2 hours; and
- the maximum duration ($TIME\_LIMIT$) of a search path allocated to a resource may not exceed 6 hours in length and this is a hard constraint ($STRICT$ =TRUE).

Further parameters that specify particular methods and strategies of searching are varied within the experiments and include:

- whether night searching is conducted or not;
- whether the sound sweep method of detection is employed;
- the resource allocation method used in each search period; for example, a sound sweep search may only be performed in the initial search periods followed by visual searching; and
- whether a region can be re-searched in the same search period.

The number of resources available for allocation to search tasks is also fixed over each search period and only ground searchers are utilized. The number of ground searchers that comprise a team ($TEAMSIZE$) is fixed at 4 over all experiments. The number of teams ($TEAMNUM$) is also set to 4, resulting in a total of 16 searchers being active in the field at any one time. This number is reduced by 2 teams if an operation is scaled down and increased by 2 teams ($SCALE\_EXTRA$) if an operation is scaled up.

The stopping rule of each experiment is the detection of the subject or the suspension of the search operation.

12.4 Computational Experiments

In total we generated three problem instances over which to test the four resource allocation methods and a limited number of search management strategies. The three problem instances vary in the type of path followed by the subject, whether he recorded his intentions, the type of region that the path terminated in, and the subject's responsiveness. In each instance the search was simulated under the more favourable weather conditions described by weather scenario one. For the third problem instance the search was also simulated under the second weather scenario in order to provide some insight into the different search results which could be expected under such conditions.

While all four allocation methods are used to determine search paths from both an interior and an exterior search base, the remaining search methods and strategies tested are only executed over a subset of the problem instances. For example, as the subject is located on an edge of the TIN in problem instance C and sound sweeps are only conducted over triangular regions, the sound sweep method is not tested for this particular problem instance.

The problem instances were created to address the specific research question of how do the resource allocation methods and search strategies respond to:

- no known subject route intentions;
- a subject moving away from their intended route;
- the subject being located in a region of relatively high or low POA;
- the subject being located in a region that requires few or many hours to search;
- a responsive or unresponsive subject;
- a subject showing evidence of confused movement;
- adverse weather conditions; and
- few or many clues being detected;

To this end, problem instance A is described by the path of a subject moving randomly, having left no route intentions, who is finally positioned in a dense triangular region. While this region requires high search effort to search and is difficult in this sense, it also has a relatively high historical POA value, which would be expected to offset the lack of route intentions. Problem instance B differs in that the subject follows a path directed by the historical POA values embedded in the TIN and displays confusion, re-entering a region previously visited. In this instance the subject does leave route intentions but moves off this intended path, finally being positioned within a triangular region of less area and density than problem instance A. This problem is easier in that an entry point into the area is known and the path followed is not random. However, it is complicated by the fact that the subject moved in a confused manner, and away from his intended route. In this respect information gained from the intended route and clues laid on the path provide conflicting evidence to the subject's actual position. The third problem instance, problem instance C, also models a subject moving randomly having left no route intentions but his path terminates on an edge region with a POA value of 0.0. The subject is unresponsive, and when the search is simulated under adverse weather the problem was anticipated to be the most difficult of the three to resolve with quick detection of the subject.

All reported computations were performed on a 75MHz Pentium running RedHat Linux 5.1 with 8M memory. All implementations are in C compiled with GNU egcs-2.90.27. Time is measured using the `gethrvtime` system call.

12.4.1 Performance Measures and Data Collected

The performance measures and data collected in the computational experiments consist of the following.

12.4.1.1 Subject Related

The data collected which relates to the subject is:

- the state of the subject when detection occurred; and
- the time that the subject became unresponsive.

12.4.1.2 Resource Related

Those performance measures and data relating to the search resources consist of:

- the total number of resources used in the operation;

- what times, if at all, withdrawal of resources occurred due to adverse weather conditions; and
- the time duration of the resource that spent the longest time in the field.

12.4.1.3 Period Related

The performance measures and data collected for each search period are:

- the ratio of POS/time achieved in each planning period;
- the predicted POS value of the planning period;
- the cumulative POS value up to the current planning period;
- the down time between the end of the previous planning period and start of the current planning period;
- the search regions comprising the primary search area and whether or not the lost region is one of these; and
- the total unallocated search hours across resources.

12.4.1.4 General

Further general performance measures and data are also collected and include:

- the time at which visual detection of the subject occurred, if at all;
- the time at which the subject was physically located;
- the time at which the search operation was scaled down, if at all;
- the time at which the search operation was scaled up, if at all;
- the total down time over the operation;
- the urgency level at the completion of the search;
- the worst weather conditions experienced;
- the minimum POD level assigned to any search region;
- the maximum POD level assigned to any non-edge search region;
- the number of clues detected;

- the time at which each clue was detected and the resulting POA value of the *lost_region*;
- the time that the *lost_region* was first allocated for searching and the number of times that it was allocated for searching over the duration of the operation;
- POD_{cum} of the region that the subject is located in;
- POD_{cum} for regions on the subject's path;
- the number, time and rationale of task reassignments;
- the time at which edges susceptible to flooding become impassable;
- the number of times that each edge is traversed - this can serve as a measure of redundancy as each edge is searched only once, the remainder of traversals being a component of an access or redeployment path; and
- the CPU time required to generate the first search path allocation.

12.4.2 Data Tabulation

A set of performance measures that capture the overall performance of a search, are tabulated for each resource allocation method examined and each problem instance. Within these data tables the following abbreviations are used:

grtst	greatest
smlst	smallest
hgst	highest
hrs	hours
NS	night searching
W	subject is capable of walking out with the searchers
H	helicopter retrieval of the subject
S	stretcher retrieval of the subject
ND	non-detection of the subject
$*(x)$	the *lost_region* needed to be allocated x times before subject detection resulted - when unannotated the subject was detected upon the first allocation

The results and analysis for each problem instance are now presented. This is followed by an analysis of the relationship between clue and subject detection, and a comparison of the performance of each of the four resource allocation methods.

12.5 Problem Instance A

The first problem instance incorporates a subject moving by a random path, without confusion. The path of the subject is generated by the parameters:

- $PATHTYPE =$ *random*;
- $CONFUSION =$ FALSE;
- $SPATHSTART =$ 27;
- $SPATHLENGTH =$ 5.

The path begins from the exterior of the TIN at vertex 27 and moves through three regions; one edge and two triangles. Specifically, the path is described by:

vertex path:	27 - 2 - 29 - 19
region path:	0 - 141 - 171
initial POA values:	0.000 - 0.012 - 0.016

where the region labels identify: ridge edge(27,2), triangle(2,12,29), triangle(29,37,19).

The region in which the subject is located, triangular region 171, has a density of level 4 and an area of 0.089 km^2 The region can be searched to 50% POD in ideal search conditions, selecting the quickest direction, in 5.65 hours. Three inorganic clues are laid in regions 0 and 141; these comprise 6% of the total clues, the remainder being organic. The search operation is conducted under the weather conditions described in weather scenario one. The subject is initially responsive and uninjured ($START\_RESPONSE$=TRUE, $START\_INJURED$=FALSE).

The intended route of the subject is unknown ($KNOW$=FALSE), hence the search base selected under the external base criterion is located at vertex 27 and the base location interior to the convex hull is located at vertex 0.

12.5.1 General Results for Problem Instance A

The performance of each of the resource allocation methods detailed in Section 8.8.5 is now examined with respect to the time taken to detect the subject under this problem instance. The search outcomes resulting from the paths constructed by each of these four methods are detailed in Table 12.3 for deployment from the interior base and in Table 12.4 for deployment from the exterior base.

When the benchmark resource allocation method is used to determine the resource's search paths, both the strategy of searching throughout the night and the strategy of only searching in daylight hours are considered. Ranking the resource allocation criteria in

Table 12.3: Resource allocation methods for problem instance A from the interior base located at vertex 0.

Method	Time of visual detection	POA_{lost_region} at completion	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark									
least hrs, NS	40.399	0.038	0	-	37.812	-	0.00	25	W
index, NS	40.491	0.038	0	-	37.904	-	0.00	26	W
grtst hrs, NS	82.490	0.102	3	74.867	79.034	4.167	0.00	53	W
grtst hrs, no NS	57.608	0.037	0	-	54.732	-	24.0	21	W
index, no NS	57.614	0.040	0	-	54.738	-	24.0	22	W
least hrs, no NS	63.896	0.043	0	-	61.309	-	24.0	25	W
Single Task									
POS	34.055	0.133	9	10.686	31.115	20.429	0.00	20	W
PSR	40.175	0.147	3	28.790	37.504	8.714	0.00	26	W
search priority	40.419	0.139	3	29.126	37.832	8.706	0.00	24	W
Path Scan									
PSR	33.786	0.089	3	28.282	30.653	2.371	0.00	20	W
search priority	34.177	0.125	3	16.558	31.301	14.743	0.00	20	W
POS	58.155	0.148	3	32.030	55.484	23.454	0.00	36	W
Primary Area									
hgst degree exit vertex	16.933	0.022	0	-	12.939	-	0.00	8	W
smlst search hrs	33.690	0.035	0	-	30.556	-	0.00	20	W
search priority	39.998	0.037	0	-	37.057	-	0.00	24	W
smlst search & access hrs	40.146	0.037	0	-	37.270	-	0.00	24	W
PSR	40.330	0.037	0	-	36.760	-	0.00	24	W
smlst access hrs	57.690	0.037	0	-	54.556	-	0.00	36	W
POS	64.677	0.037	0	-	61.047	-	0.00	40	W
hgst elevation	-	0.045	0	-	-	-	82.8	93	ND

Table 12.4: Resource allocation methods for problem instance A from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
index	16.244	0.073	9	8.421	13.262	4.841	0.00	8	W
grtst hrs	57.905	0.040	0	-	54.862	-	0.00	35	W
least hrs	64.246	0.045	0	-	61.556	-	0.00	39	W
index, no NS	16.244	0.073	9	8.421	13.262	4.841	0.00	8	W
grtst hrs, no NS	57.699	0.039	0	-	54.572	-	24.00	20	W
least hrs, no NS	58.249	0.038	0	-	55.559	-	24.00	20	W
Single Task									
POS	32.773	0.137	9	10.486	30.000	19.514	0.00	19	W
search priority	40.565	0.159	9	23.596	37.875	14.279	0.00	24	W
PSR	40.648	0.180	9	30.298	37.958	7.660	0.00	24	W
Path Scan									
search priority	34.533	0.120	3	15.767	31.843	16.076	0.00	20	W
PSR	64.287	0.074	9	35.120	61.597	26.487	0.00	35	W
POS	64.482	0.156	9	31.118	61.792	30.674	0.00	37	W
Primary Area									
smlst search & access hrs	39.561	0.111	9	26.396	36.433	10.037	0.00	24	W
smlst access hrs	39.972	0.031	0	-	36.929	-	0.00	24	W
search priority	39.981	0.035	0	-	36.938	-	0.00	24	W
POS	40.226	0.037	0	-	37.536	-	0.00	24	W
smlst search hrs	40.283	0.034	0	-	37.593	-	0.00	24	W
hgst degree exit vertex	40.838	0.034	0	-	36.686	-	0.00	24	W
PSR	81.154	0.044	0	-	78.457	-	0.00	50	W
hgst elevation	135.973	0.0793	9	73.889	132.000	58.111	8.46	85	S

increasing order of subject detection time results in a consistent ordering for the exterior base under both night searching strategies but the ordering that arises from the interior base deployment reverses itself between these strategies.

The method that results in the earliest subject detection (index ordering from the exterior base) is also the method to first detect a clue and to detect the maximum number of clues. This method allocates the triangular region (2,12,29) on the subject's path, to a search resource upon completion of their initial allocation, as an alternative region (adjacent to the search resource's position) when all edges in the primary search area have already been allocated for searching. Edge regions alone are the specified candidate allocation set in the first search period. The edge region (27,2) on the subject's path is also searched as part of a redeployment path to the search base that covers unsearched edges. The benefit of this diversity stratagem is particularly evident in this instance where the initial POA value of this edge is 0.0 and, as such, would make it an unlikely candidate for search allocation based on traditional search criteria. Those methods that detected no clues were allocated no regions on the subject's path to search.

In general, one would expect a strategy of night searching to dominate a strategy of no night searching due to the increased number of hours available to search. In this respect an unexpected result occurs for all criteria when deploying resources from the exterior base and for the resource allocation criterion of greatest hours remaining, when deploying resources from the interior base. Under these criteria the strategy of no night searching actually results in the detection of the subject at an earlier, or exactly the same, time as that resulting from night searching. Upon investigation the reason for these results was found to be a product of the modelling decision to allow only one resource to search a region at a time and the setting of a minimum POD value that must be achievable before committing a resource to search a region. In the problem instance described, the region in which the subject is located cannot be searched by one resource to a level greater than MINPOD (20%), within six hours, in the restricted night light conditions; hence the region is not considered a candidate for searching during this period.

The experimental results show that the POS region criterion is the fastest to detect the subject under the single task method from both base locations. The other two criteria tested under this method - search priority and PSR - both produce similar results to one another and between base locations. Of the four methods tested, the single task method produces the most consistent search results, independent of base location.

Search operations were simulated for the path scan method over three region criteria and three resource criteria. The results of these simulations are detailed in Table 12.5 and Table 12.6 for each base location. It is interesting to note that, when deploying resources from the interior base, the region criteria, when ranked in order of subject detection time, maintain the same ordering irrespective of resource criterion. This observation does not hold for the exterior base. The PSR region criterion produces superior results when resources are deployed from the interior base, while the search priority criterion is the better performer when deploying resources from the exterior base. The resource selection criterion that results in the quickest subject detection over all criteria and base location, is the greatest number of search hours remaining for allocation. The greatest number of remaining search hours is, however, not superior for each region criterion over both base locations. For example, the index-ordering criterion produces a faster subject detection time for the POS region criterion when the base is located on the interior of the TIN. When the base is located on the exterior of the TIN, no pattern between resource criteria and earlier detection times is evident over the three region criteria.

Experimental runs for the primary search area method consider each of the region selection criteria detailed in Section 8.8.2. The highest elevation criterion is the worst performer across both base locations. Applying this criterion resulted in the only instance, across all methods, of subject non-detection, and detection of the subject after he became unresponsive and required stretcher-retrieval from the search area. Search priority was the criterion that performed the most consistently, irrespective of the search base location, while the other criteria tested tended to favour deployment from one location. The majority of criteria detected the subject faster when deploying from the exterior base. The criterion that resulted in the greatest diversity in subject detection times between base locations was the PSR criterion; the difference in these detection times was 40.8 hours.

Except for the initial allocation of search resources at the commencement of the search operation, where all resources are located together at the base, the SAR simulation usually allocates tasks to search resources individually as they complete their previous search path. Hence, the resource criteria parameter is not required for the majority of allocations, except at the beginning of a search period when more than one resource is inactive and awaiting assignment. Given this, attempts to differentiate search results by resource criteria will likely be ineffective unless all resources are allocated their search assignments at the same time or the subject is detected within the first period.

12.5.1.1 Hybrid Path Scan Methods

Hybrid variants of the path scan allocation method were also investigated. These variants allocate a resource a search path generated in consideration of two region criteria - one governing the construction of the first half of the path, the other, the second half. Experiments were conducted for paths whose first half was constructed via each of the three priority criteria, and whose second half was constructed via each of the criteria

Table 12.5: Path scan method of resource allocation for all resource criteria, for problem instance A from the interior base located at vertex 0.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
greatest hrs									
PSR	33.786	0.089	3	28.282	30.653	2.371	0.00	20	W
search priority	34.177	0.125	3	16.558	31.301	14.743	0.00	20	W
POS	58.155	0.148	3	32.030	55.484	23.454	0.00	36	W
least hrs									
PSR	34.201	0.128	3	16.973	31.325	14.352	0.00	20	W
search priority	58.195	0.150	3	32.617	55.523	22.906	0.00	36	W
POS	58.231	0.148	3	32.823	55.559	22.736	0.00	36	W
index									
PSR	34.042	0.090	3	18.561	31.165	12.604	0.00	20	W
search priority	34.228	0.131	3	14.817	31.352	16.535	0.00	20	W
POS	40.490	0.112	3	35.448	37.903	2.455	0.00	24	W

Table 12.6: Path scan method of resource allocation for all resource criteria, for problem instance A from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
greatest hrs									
search priority	34.533	0.120	3	15.767	31.843	16.076	0.00	20	W
PSR	64.287	0.074	9	35.120	61.597	26.487	0.00	35	W
POS	64.482	0.156	9	31.118	61.792	30.674	0.00	37	W
least hrs									
search priority	40.254	0.119	3	32.432	37.564	5.132	0.00	24	W
POS	40.382	0.112	3	34.488	37.692	3.204	0.00	24	W
PSR	58.280	0.155	9	24.453	55.298	30.845	0.00	34	W
index									
PSR	40.271	0.125	9	24.238	37.289	13.051	0.00	23	W
search priority	40.697	0.171	9	11.443	38.007	26.564	0.00	23	W
POS	58.026	0.141	3	33.150	55.044	21.894	0.00	33	W

Table 12.7: Ranking of secondary selection criteria for the hybrid path scan method for problem instance A.

Criterion	Base location		Overall Rank
	Interior base	Exterior base	
Search priority	2.0	1.0	1.5
POS	1.7	1.7	1.7
PSR	3.0	1.3	2.2
Highest elevation	2.7	2.0	2.3
Smallest access hours	3.0	3.0	3.0
Smallest search hours	4.0	6.3	5.2
Smallest search and access hours	5.3	5.0	5.2
Highest degree exit vertex	6.3	6.0	6.2

described in Section 8.8.2. Paths were generated from both the interior and exterior search base. The results of these experiments are depicted in Table 12.8 and Table 12.9

It can be seen from these results that, when paths are generated from the exterior base, that the hybrid method provides superior detection results to the path scan method using only a single region selection criterion for both the initial POS and PSR criteria. Some improvement is also noted for the search priority criterion, particularly when the second half of the path is constructed with respect to the region having highest elevation. The time to visual detection in this instance is less than half that of the non-hybrid path and is the best result for visual only searching from the exterior base, for all of the methods tested over this problem instance.

Such improved results are not evident when paths are generated from the interior base, except for the POS criterion, where the majority of secondary selection criteria produce improved detection times. The paths generated from the PSR criterion, in particular, perform poorly with respect to the non-hybrid PSR path, even though the first clue was often detected sooner. The earliest clue detection occurred at 10.533 (PSR/highest elevation criterion), almost 18 hours earlier than for the single criterion method. The ranking of the secondary selection criteria for both base locations and three initial criteria is found in Table 12.7.

When formulating the hybrid path generation method it was thought that 'packing' the second half of the path with search regions which satisfied some minimum time or maximum access criterion, would result in higher quality paths, leading to faster subject detection. It is then, with interest, that we note that the secondary selection criteria that result in faster subject detection are those with a priority component. In fact, the best performer is the search priority criterion that could be viewed as containing the best of both considerations - a priority and a time component.

12.5.2 Sound Sweep Method

When the sound sweep search method is used in the first two periods of searching (up until 18.00) from the exterior search base, the results of the search are detailed in Table 12.10. These results are compared with those obtained by visual searching alone, in Table 12.11. The incidence of clues detected when conducting a sound sweep within these two periods is less, due to the probability of clue detection being lower.

While the benchmark method did not detect the subject while sound sweep searching, the time until detection was still faster than when using visual searching alone, except for the index ordering resource criterion. This trend of detecting the subject more quickly when employing the sound sweep technique is also evident for the single task method, and for the majority of region selection criteria for the primary search area and path scan allocation methods.

Sound sweep searching over the first two search periods that is then followed by visual searching is more than twice as effective (in terms of time to subject detection) as solely visual searching in five (30%) of the methods tested: single task with POS; path scan with POS; and primary search area with highest degree exit vertex, PSR and POS. It is of particular interest that, in all instances where the POS region selection criterion was used, better results were obtained when utilizing sound detection. In each of these instances the subject was detected via sound in these initial periods.

If the status of the subject is altered to be unresponsive from the outset of the operation ($START\_RESPONSE$ = FALSE), the methods that did not detect the subject during the sound sweep phase return the same results as for a responsive subject. Those resource allocation methods that detected the responsive subject via sound detection (at a time prior to 18.00) do, however, return different and interesting results. These results are presented in Table 12.12. One would have expected that the use of a technique which relied on the response of the subject for detection to occur would perform poorly in comparison to the use of a technique that did not require a response, when the subject was unresponsive. This is true for the primary search area method with highest degree exit vertex criterion, which did not detect the subject before search suspension, but it is not the case in all instances. In particular, the primary search area method with POS and PSR selection criteria, and the path scan method with POS selection criterion, detect the subject earlier when employing the sound sweep technique than when utilizing visual detection searching alone. In these cases the region in which the subject is located is searched twice, once via sound and once visually, with detection occurring with the visual

Table 12.8: Hybrid path scan methods of resource allocation for problem instance A from the interior base located at vertex 0.

Region criteria mix	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Search priority/									
POS	34.037	0.093	3	16.754	31.161	14.407	0.00	20	W
hgst elevation	34.080	0.084	9	10.049	31.204	21.155	0.00	20	W
smlst access hrs	34.128	0.095	9	10.049	31.456	21.407	0.00	20	W
smlst search hrs	40.056	0.120	3	34.055	37.116	3.061	0.00	24	W
PSR	40.366	0.130	3	29.342	37.779	8.437	0.00	24	W
hgst degree exit vertex	40.385	0.119	3	34.255	37.797	3.542	0.00	24	W
smlst search & access hrs	40.577	0.120	3	32.187	37.989	5.802	0.00	24	W
POS/									
hgst elevation	16.086	0.108	3	11.397	13.146	1.749	0.00	8	W
smlst search hrs	40.317	0.082	3	34.055	37.646	3.591	0.00	24	W
smlst access hrs	40.727	0.149	9	10.653	37.703	27.050	0.00	24	W
PSR	58.150	0.151	3	32.107	55.478	23.371	0.00	36	W
search priority	58.152	0.151	3	32.766	55.480	22.714	0.00	36	W
hgst degree exit vertex	58.285	0.144	3	32.480	54.716	22.236	0.00	36	W
smlst search & access hrs	64.278	0.163	3	38.538	61.607	23.069	0.00	40	W
PSR/									
search priority	33.825	0.121	3	14.681	30.691	16.010	0.00	20	W
smlst search & access hrs	34.101	0.089	3	28.006	31.225	3.219	0.00	20	W
smlst access hrs	34.103	0.090	7	17.268	31.227	13.959	0.00	20	W
POS	34.313	0.117	3	29.072	30.696	1.624	0.00	20	W
highest elevation	34.476	0.140	3	10.533	31.888	21.355	0.00	20	W
smlst search hrs	40.150	0.116	3	34.291	37.479	3.188	0.00	24	W
hgst degree exit vertex	57.780	0.134	3	36.567	54.646	18.079	0.00	36	W

Table 12.9: Hybrid path scan methods of resource allocation for problem instance A from the exterior base located at vertex 27.

Region criteria mix	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Search priority/									
hgst elevation	16.602	0.071	9	9.648	12.605	2.957	0.00	8	W
POS	33.793	0.047	9	16.018	30.750	14.732	0.00	20	W
smlst access hrs	34.020	0.105	9	9.684	30.977	21.293	0.00	19	W
PSR	40.814	0.159	9	11.459	38.124	26.665	0.00	23	W
hgst degree exit vertex	58.208	0.134	3	38.182	55.226	17.044	0.00	36	W
smlst search hrs	58.247	0.094	2	41.773	55.265	13.492	0.00	35	W
smlst search & access hrs	58.407	0.130	3	37.466	55.424	17.958	0.00	34	W
POS/									
hgst elevation	34.500	0.135	9	10.440	31.810	21.370	0.00	18	W
search priority	40.177	0.116	3	32.587	37.487	4.900	0.00	24	W
smlst search & access hrs	40.321	0.081	2	35.726	37.631	1.905	0.00	24	W
PSR	56.773	0.142	3	32.868	54.000	21.132	0.00	33	W
smlst access hrs	56.773	0.099	7	22.683	54.000	31.317	0.00	33	W
smlst search hrs	61.938	0.113	9	35.663	60.190	24.527	0.00	36	W
hgst degree exit vertex	64.270	0.121	9	33.200	61.580	28.380	0.00	38	W
PSR/									
search priority	40.233	0.116	3	34.315	37.543	3.228	0.00	23	W
smlst access hrs	40.478	0.137	9	15.931	37.788	21.857	0.00	24	W
POS	40.568	0.111	9	28.731	37.878	9.147	0.00	24	W
hgst elevation	40.590	0.155	9	10.157	37.899	27.742	0.00	23	W
smlst search & access hrs	58.053	0.174	9	30.157	55.010	24.853	0.00	33	W
hgst degree exit vertex	58.182	0.066	9	33.977	55.199	21.222	0.00	34	W
smlst search hrs	62.052	0.219	9	30.254	60.196	29.942	0.00	36	W

Table 12.10: Sound sweep search method for a responsive subject in the first two periods, for problem instance A from the exterior base located at vertex 27.

Method	Time of subject detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
least hrs	32.773	0.090	3	22.679	30.000	7.321	0.00	19	W
index	34.022	0.120	9	10.839	30.979	20.140	0.00	20	W
grtst hrs	39.793	0.043	0	-	36.749	-	0.00	24	W
Single task									
POS	16.046	0.075	7	13.290	14.370	1.080	0.00	8	W
search priority	32.773	0.099	3	14.324	30.000	15.676	0.00	19	W
PSR	40.688	0.156	9	17.144	37.998	20.854	0.00	24	W
Path scan									
POS	16.031	0.031	0	-	12.207	-	0.00	8	W
search priority	40.452	0.140	9	11.436	37.762	26.326	0.00	23	W
PSR	40.509	0.119	9	14.892	37.819	22.927	0.00	23	W
Primary area									
hgst degree exit vertex	9.842	0.021	0	-	6.000	-	0.00	4	W
PSR	14.971	0.034	0	-	13.126	-	0.00	8	W
POS	16.801	0.036	0	-	13.082	-	0.00	8	W
smlst search hrs	39.916	0.120	9	26.182	36.000	9.818	0.00	24	W
search priority	40.823	0.081	9	26.135	36.695	10.560	0.00	22	W
hgst elevation	57.810	0.040	0	-	54.766	-	0.00	35	W
smlst access hrs	58.554	0.171	9	15.981	55.864	39.883	0.00	35	W
smlst search & access hrs	63.482	0.096	9	48.709	60.354	11.645	0.00	39	W

Table 12.11: Sound vs. visual searching strategies from the exterior search base.

Method	Time to subject detection		Ratio
	Sound then visual search	Visual search	
Benchmark, least hours	32.773	64.246	0.510
Benchmark, greatest hours	39.793	57.905	0.687
Benchmark, index	34.022	16.244	2.094
Single task, POS	16.046	32.773	0.490
Single task, search priority	32.773	40.565	0.808
Single task, PSR	40.688	40.648	1.001
Primary search area, highest degree exit vertex	9.842	40.838	0.241
Primary search area, PSR	14.971	81.154	0.184
Primary search area, POS	16.801	40.226	0.418
Primary search area, smallest search hours	39.916	40.283	0.991
Primary search area, search priority	40.823	39.981	1.021
Primary search area, smallest access hours	58.554	39.972	1.465
Primary search area, smallest search & access hours	63.482	39.561	1.605
Primary search area, highest elevation	135.973	57.810	2.352
Path scan, POS	16.031	64.482	0.249
Path scan, search priority	40.452	34.533	1.171
Path scan, PSR	40.509	64.287	0.630

search.

As the POD of a sound sweep is calculated with respect to a responsive subject, the POD level and subsequent POS value of a search are over-stated when the subject is in fact unresponsive. Hence, if the region in which the subject is located is searched via a sound sweep and the subject is unresponsive, the POA value of the region is reduced by a greater factor than is warranted. This is detrimental as it reduces the relative likelihood of the subject still being within the region, hence decreasing the likelihood of the region being re-searched. In light of this, the results displaying faster detection times when this occurs, over visual-only searching, are even less anticipated.

An examination of the output of these simulated searches shows that the search paths generated by the two region criteria for the primary search area method (which detected the unresponsive subject faster when initially employing sound sweep searching), detected clues indicating that the subject had moved in the direction of the *lost_region*. This contributed to higher POA values for the *lost_region* which was subsequently included

Table 12.12: Sound sweep search method for an unresponsive subject for those methods detecting a responsive subject via sound within the first two periods, for problem instance A from the exterior base located at vertex 27.

Method	Time of subject detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Single task									
POS	57.853	0.083	9	10.486	14.370	3.884	0.00	35	H
Path scan									
POS	40.537	0.062	9	17.900	12.207	(5.693)	0.00	23	H
Primary area									
POS	33.716	0.058	9	17.048	13.082	(3.966)	0.00	20	H
PSR	40.019	0.097	9	26.114	13.126	(12.988)	0.00	22	H
hgst degree exit vertex	-	0.060	6	109.343	6.000	(103.343)	81.31	92	ND

within the primary search area as a candidate for allocation, earlier than when visual searching alone was employed. No such clues were detected by the visual-only searches where the regions comprising the subject's path were not searched due to low relative POA values. The path scan method with POS criterion detected the same total number of clues under sound searching as it did under visual-only searching, but detected clues that revealed subject movement towards the *lost_region* at two different stages of the search. At the time of the latter detection the relative POA value of the *lost_region* was subsequently higher than its POA value at the same stage of the search under solely visual searching. The region was then allocated for searching at that time, with successful search detection resulting.

Four periods of sound sweep searching were also trialled for each allocation method from the exterior base, using the search priority criterion for each. These results are displayed in Table 12.13 for a responsive subject and in Table 12.14 for an unresponsive subject. Under this strategy the detection of a responsive subject is achieved more quickly than when sound sweeping for only 2 periods, for all but the benchmark method. However, quicker detection occurs for only half of the methods when the subject is unresponsive.

12.5.3 Re-searching Regions

Further experiments were conducted setting $MULTIPLE = TRUE$ for each path method which, in its general description, does not allow for re-searching of a region within the same search period. When this constraint is relaxed the resulting search outcomes for paths originating from the exterior base are detailed in Table 12.15. Permitting re-searching can be viewed as a way to mimic the search of a region at a higher POD level than is obtainable from either the use of a target POD level or from the search of a region by a single resource. The strategy of re-searching only responds to new information more quickly if initial search allocations are completed within the period while further search allocation hours are available; in this sense it is distinct from the single task method.

When re-searching was permitted, the search results for the benchmark method exhibit almost identical or superior detection times to those when re-searching was not permitted until a later period. No more clues were detected by this process and the first clue was detected at the same time as when re-searching did not occur.

The search paths generated by the primary search area method, when re-searching of regions was permitted, resulted in identical or slower detection times for the majority of the region criteria tested. The exceptions to this were the PSR and POS region criteria that detected the subject faster for all of the methods that used these criteria, when

Table 12.13: Sound sweep search method for a responsive subject in the first four periods using the search priority region criterion, for problem instance A from the exterior base located at vertex 27.

Method	Time of subject detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Primary area	22.739	0.036	0	-	19.332	-	0.00	12	W
Single task	27.775	0.113	2	14.324	24.000	9.676	0.00	16	W
Path scan	34.623	0.126	8	11.436	31.933	20.497	0.00	19	W
Benchmark, least hrs	40.586	0.041	0	-	37.896	-	0.00	23	W

Table 12.14: Sound sweep search method for an unresponsive subject in the first four periods using the search priority region criterion, for problem instance A from the exterior base located at vertex 27.

Method	Time of subject detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Primary area	33.858	0.038	0	-	19.332	-	0.00	19	H
Path scan	34.623	0.126	8	11.436	31.933	20.497	0.00	19	H
Benchmark, least hrs	40.586	0.041	0	-	37.896	-	0.00	23	H
Single task	58.509	0.099	9	14.324	24.000	9.676	0.00	35	H

multiple searches of regions could be conducted within the same search period.

The single task method permits re-searching; however, when this is not permitted the resulting subject detection times are detailed in Table 12.16 and are similar to those that result from permitting re-searching. The only criterion to detect the subject faster without re-searching regions within the same search period is the search priority criterion. This result is explained by the fact that, unlike the other search results for this method, the first clue detected indicated that the subject moved towards the *lost_region*, hence increasing the region's POA value. The total number of clues detected under each strategy is the same but the first clue is detected earlier, or at the same time, when re-searching is not permitted. This observation is explained by the diversity of search regions allocated via this strategy. The time that then elapsed between this clue detection and the subsequent allocation of the *lost_region* for searching, is less for the majority of criteria under the strategy of re-searching.

12.5.4 Cumulative POS

Figure 12.7 through to Figure 12.10 depict the probability of having detected the subject at the end of each search period for each of the path generation methods when the search base is located on the exterior of the search area. The selection criteria employed by each method are graphed such that the curve depicted by the solid line is the fastest to detect the subject, and the curve depicted by the dashed line, '- -', is the slowest to detect the subject. For clarity the three region selection criteria resulting in the fastest subject detection are the only three graphed for the primary search area method. Figure 12.11 illustrates the POS_{cum} curves of the region selection criterion within each of these four methods, to first detect the subject.

Examination of these graphs reveals smaller POS_{cum} increases over periods of searching which extend throughout the night (indicated by search periods three and four, seven and eight, *etc.*). The flatter curve over these periods is due to the lower POD values obtainable under night search conditions. As expected POS_{cum} growth is greatest in the initial search periods where searchers are allocated to regions with high POA values.

In the majority of the experiments depicted, the path construction method that detected the subject first was also the method with smallest total POS_{cum} at the time that the subject was detected; the primary area method is the only exception to this observation.

Table 12.15: Permitting re-searching of a region within the same search period, for problem instance A from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
index	16.152	0.079	9	8.421	13.462	5.041	0.00	8	W
grtst hrs	39.945	0.040	0	-	36.902	-	0.00	24	W
least hrs	64.290	0.045	0	-	61.600	-	0.00	40	W
Path scan									
search priority	39.824	0.137	3	32.095	36.781	4.686	0.00	23	W
POS	40.588	0.131	3	33.422	37.898	4.476	0.00	23	W
PSR	63.558	0.124	9	32.640	60.142	27.502	0.00	37	W
Primary area									
search priority	39.981	0.037	0	-	36.938	-	0.00	24	W
POS	39.982	0.037	0	-	36.939	-	0.00	24	W
smlst search hrs	57.998	0.035	0	-	54.955	-	0.00	36	W
smlst search & access hrs	58.241	0.034	0	-	54.678	-	0.00	36	W
PSR	63.659	0.117	9	48.456	60.531	12.075	0.00	39	W
hgst degree exit vertex	63.948	0.035	0	-	60.905	-	0.00	38	W
smlst access hrs	83.489	0.038	0	-	79.050	-	0.00	52	W

Table 12.16: Not permitting re-searching of a region within the same search period for the single task allocation method, for problem instance A from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
search priority	34.704	0.044	9	16.346	32.014	15.668	0.00	20	W
POS	38.343	0.152	9	10.486	36.090	25.604	0.00	22	W
PSR	40.762	0.166	9	16.905	38.072	21.167	0.00	24	W

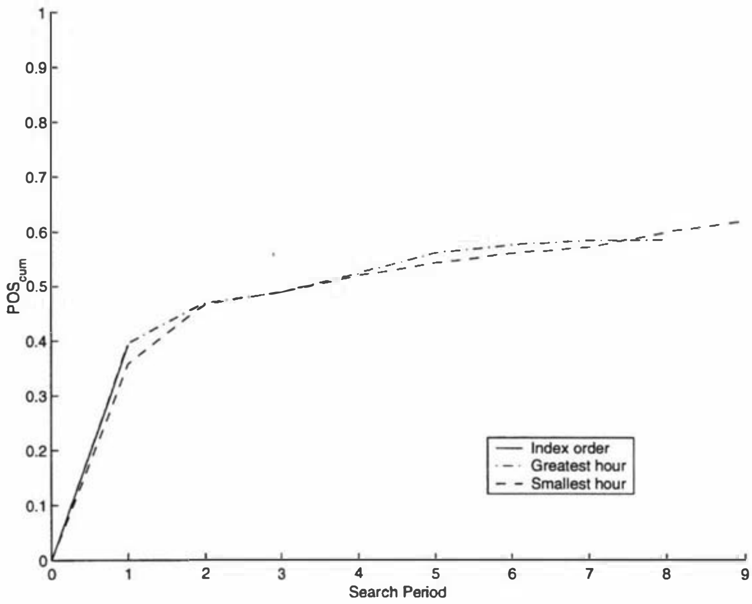

Figure 12.7: The growth of POScum under the benchmark method from the exterior base with night searching strategy.

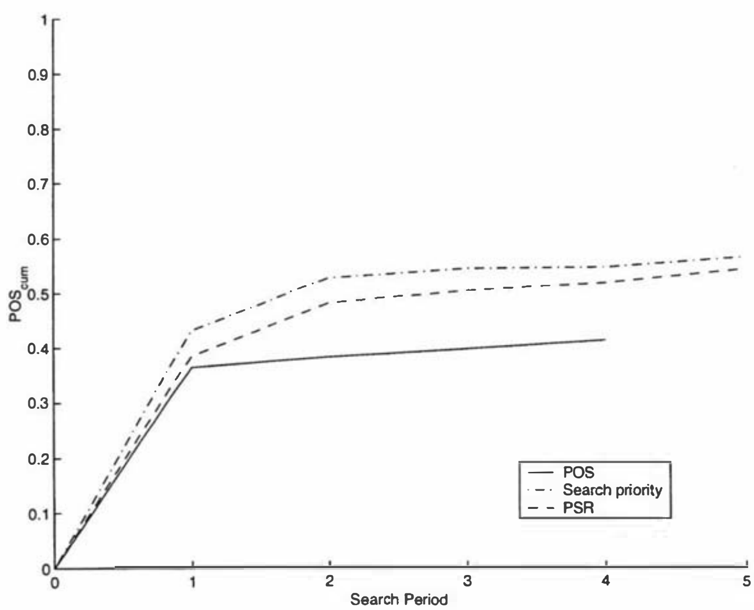

Figure 12.8: The growth of POScum under the single task method from the exterior base.

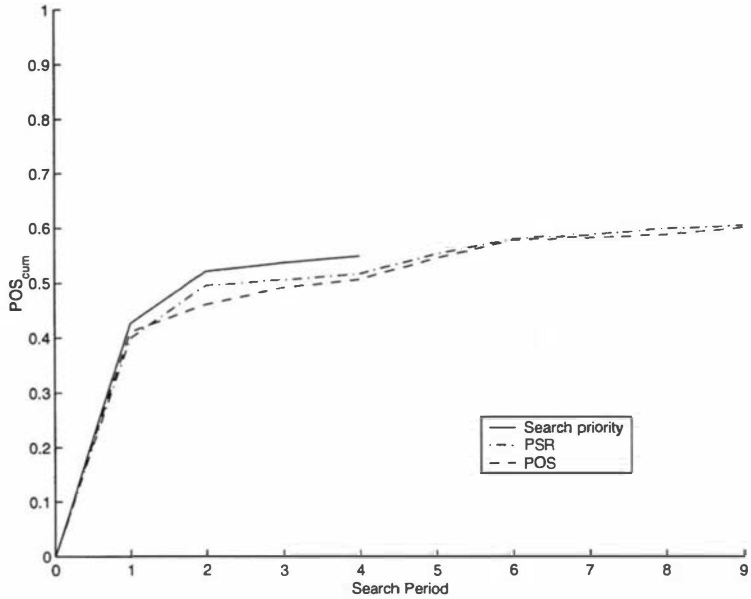

Figure 12.9: The growth of POS_{cum} under the path scan method from the exterior base.

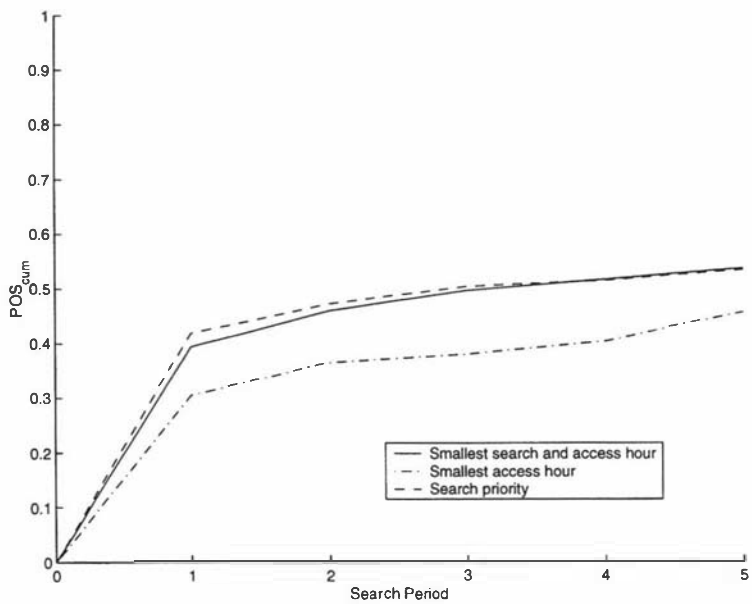

Figure 12.10: The growth of POS_{cum} under the primary search area method from the exterior base.

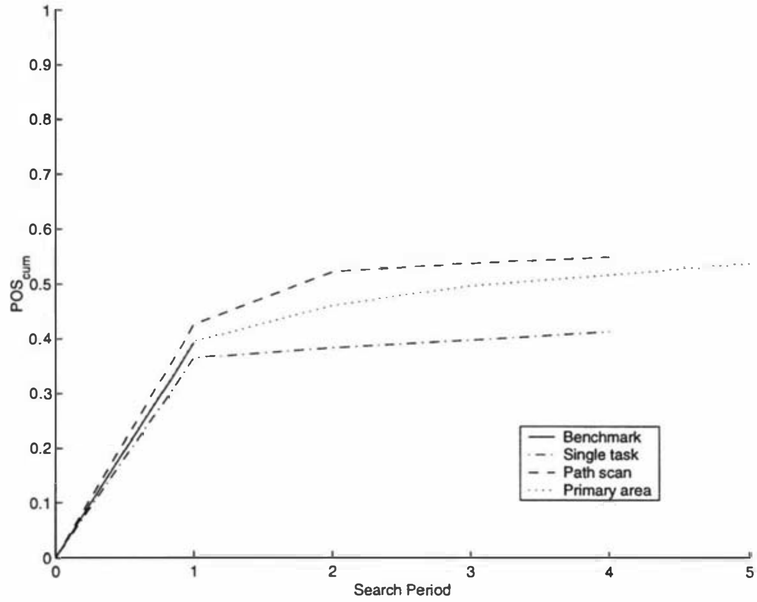

Figure 12.11: Comparison of the growth of POS_{cum} over resource allocation methods from the exterior base for problem instance A.

12.5.5 Base Location

When comparing search results between base locations, no one location is shown to lead to the detection of the subject faster for the majority of resource allocation methods. The path scan method was the only method to detect the subject faster, over all region criteria, from a single base location and this occurred from the internally-located search base. This observation also held when the basic path scan method was extended to hybrid variants. In this instance the search paths generated from the internally located search base detected the subject more quickly, when comparing individual region selection criteria, for the majority of criteria. In general, the resource allocation methods that selected regions for searching from a primary search area detected the subject faster when deploying from the exterior base location, while the methods that did not define an initial target search area obtained better search results from the interior base location.

12.5.6 Method Comparison

In order to compare the relative performance of the four resource allocation methods and the hybrid variants of the path scan method, we compare their best, median and worst case solutions to the best solution obtained over all methods. This form of analysis is

Table 12.17: Performance ratios for visual search resource allocation methods for problem instance A when deploying from the exterior base.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Benchmark	1.000	3.565	3.955	48.002
Single task	2.018	2.497	2.502	7.875
Path scan	2.126	3.958	3.970	29.949
Primary area	2.435	2.478	8.371	96.412
Hybrid - search priority	1.022	2.513	3.596	41.805
Hybrid - POS	2.124	3.495	3.957	29.770
Hybrid - PSR	2.477	2.499	3.820	21.819

used as the optimal solution to each of the problem instances is both unknown and unobtainable due to the complexity of the problem. The median performance measure was chosen over the average performance measure as it is less affected by extreme solutions.

The ratio to the best solution found is calculated for the best solution for each allocation method such that:

$$\text{best performance ratio} = \frac{\text{best path solution}}{\text{best overall solution}}$$

The median and worst performance ratios are similarly calculated.

Comparisons are made for each base location for visual-only searching, and then for both visual and sound searching strategies when resources are deployed from the exterior base. The best solution obtained by visual searching from the exterior base was the detection of the subject at 16.244 when resources were allocated by the benchmark with index order resource criterion. When deploying resources from the interior base, the best solution obtained was the detection of the subject at 16.086 when resources were allocated by the hybrid path method. This method generated the first portion of search paths on the basis of the POS criterion and the second portion on the basis of the region with highest elevation. The performance ratios for visual searching are detailed in Table 12.17 and Table 12.18.

When sound searching is utilized the subject is detected in the fastest time over all the methods tested, at time 9.842. This result occurred when resources were allocated search assignments by the primary search area method with highest degree exit vertex criterion. The performance ratios for the comparison of visual and sound searching strategies are detailed in Table 12.19 and the methods included for sound searching are those where the sound sweep method was used in the first two search periods followed by visual searching. When the use of sound search techniques is compared with the use of visual-only search techniques, a reduction in median and worst performance ratios is

Table 12.18: Performance ratios for visual search resource allocation methods for problem instance A when deploying from the interior base.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Benchmark	2.511	2.517	5.128	42.091
Single task	2.117	2.498	2.513	6.364
Path scan	2.100	2.125	3.615	24.369
Primary area	1.053	2.501	-	-
Hybrid - search priority	2.116	2.490	2.523	6.540
Hybrid - POS	1.000	3.615	3.996	48.192
Hybrid - PSR	2.103	2.133	3.592	23.955

Table 12.19: Performance ratios comparing sound and visual searching resource allocation methods for problem instance A when deploying from the exterior base.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Sound Searching				
Benchmark	3.330	3.457	4.043	7.020
Single task	1.630	3.330	4.134	24.642
Path scan	1.629	4.110	4.116	24.478
Primary area	1.000	4.102	6.450	53.640
Visual searching				
Benchmark	1.650	5.883	6.528	48.002
Single task	3.330	4.122	4.130	7.875
Path scan	3.509	6.532	6.552	29.949
Primary area	4.020	4.090	13.816	96.412

evident, with the only exceptions returning ratios that are approximately equal under both search methods. When the best performance ratios are considered this trend holds for all but the benchmark method.

The single task method performs consistently well when the best, median and worst performance measures are taken into account, and it has the least variation in subject detection times of the methods tested. The primary search area method returns the greatest variation in subject detection times across both base locations, and sound and visual searching strategies. It is also the only method to fail in detecting the subject. The remaining resource allocation methods perform similarly to one another in solving this particular problem instance.

12.6 Problem Instance B

The second problem instance incorporates a subject moving via a path which is in line with the POA values of the TIN and displaying confusion. The path of the subject is generated by the parameters:

- $PATHTYPE$ = *poa*;
- $CONFUSION$ = TRUE;
- $SPATHSTART$ = 27;
- $SPATHLENGTH$ = 4.

Referring again to Figure 12.3, the path begins from the exterior of the TIN at vertex 27 and moves through seven regions, visiting one region twice. The path is solely comprised of triangular regions of varying density classification. Specifically, the path is described by:

vertex path:	27 - 17 - 35 - 17 - 35 - 24 - 23 - 10
region path:	116 - 167 - 168 - 167 - 130 - 137 - 132
initial POA value:	0.008 - 0.014 - 0.012 - 0.014 - 0.006 - 0.010 - 0.011

where the region labels identify: triangle (27,32,17), triangle (17,23,35), triangle (35,24,17), triangle (24,35,23), triangle (24,15,23) and triangle (23,14,10).

The region in which the subject is located in, triangular region 132, has a density of level 3 and an area of 0.106 km^2. The region can be searched to 50% POD in ideal search conditions, selecting the quickest direction, in 53 minutes. Ten percent of the clues laid along the subject's path are inorganic. These clues are located within regions 116, 168 and 167. The search operation is conducted under the weather conditions described in weather file one. The subject is initially responsive and uninjured.

In this search scenario the intended route of the subject is known ($KNOW$=TRUE) and represented by:

vertex path:	27 - 17 - 13 - 20
region path:	5 - 6 - 17

where the region labels identify: track edges (27,17), (17,13) and (13,20), and form a route to the hut located at vertex 20. The initial POA values given for the regions along the subject's path are calculated after consideration of the subject's intended route.

The search base selected under the external base criterion is located at vertex 27, the origin vertex of the subject's intended route. The base location interior to the convex hull is also positioned at a vertex on the intended path, vertex 13.

12.6.1 General Results for Problem Instance B

Each of the four resource allocation methods are trialled over this problem instance, with the search results detailed in Table 12.20 for the exterior base and in Table 12.21 for the interior base.

It is clear from Table 12.20 that the strategy of searching throughout the night dominates a strategy of no night searching. When searching is continuous in this respect, successful detection can occur during the night, as is seen for the benchmark allocation method with index order resource criterion. Unlike problem instance A, the *lost_region* has a terrain density classification of 3 and can be searched by a single team to the target POD level, during the night.

When allocating search paths by the benchmark method, the most consistent results obtained were from the exterior base, while both the slowest and fastest subject detection times were achieved when using the interior base. The fastest time achieved by this method was eleven times faster than the slowest time, indicating the great variability among the results obtained by the different resource criteria. This detection time was the fastest over all four methods and occurred when the *lost_region* was allocated as an alternative region (via Algorithm 8.5), adjacent to the searcher's position, when no regions within the current primary search area could be allocated within the time constraints. As the *lost_region* was not within the defined primary search area at this stage of the operation, the advantage of such a strategy is evident.

Of all the methods tested over this problem instance, the single task allocation method performed the best across region criteria and search base location. The same ranking of resource criteria, independent of base location, was observed for both this method and for the path scan method. An interesting result is observed when allocating search paths via the path scan method with POS criteria from the exterior base: this method produced a very slow subject detection time, even though the *lost_region* was actually allocated for searching at an earlier time than both the PSR and search priority criteria which detected the subject much earlier. Upon examination of the simulation output this was found to occur because the end of the search period occurred whilst the *lost_region* was being searched. At this time the resource searching the region was redeployed to the base as their remaining useful search hours were less than $MINLIMIT$. The *lost_region* was then subsequently allocated for searching a further three times before detection resulted. Detection did not result in earlier searches due to the POD level of the search and the pseudo random number stream governing subject detection. In this respect, the POS criterion does not perform as badly as indicated by subject detection time alone.

The primary search area method was inferior to all other methods over both base

Table 12.20: Resource allocation methods for problem instance B from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at completion	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark, no NS									
least hrs	-	0.029	6	13.483	-	-	133.7	56	ND
grtst hrs	-	0.031	6	13.549	-	-	145.5	64	ND
index	-	2.850	6	13.549	-	-	145.38	60	ND
Benchmark, NS									
least hrs	58.874	0.023	5	13.483	57.776	44.293	0.00	36	W
index	76.800	0.021	6	13.549	75.679	62.130	0.00	48	W
grtst hrs	80.809	0.029	5	13.549	78.589	65.040	0.00	52	W
Single task									
search priority	14.557	0.025	0	-	13.864	-	0.00	8	W
PSR	17.030	0.025	0	-	15.800	-	0.00	9	W
POS	39.418	0.018	5	13.483	38.079	24.596	0.00	24	W
Path scan									
PSR	15.449	0.024	0	-	12.892	-	0.00	8	W
search priority	15.885	0.010	3	14.419	12.797	(1.622)	0.00	8	W
POS ∗(4)	130.940	0.008	6	13.483	12.533	(0.950)	8.46	80	S
Primary area									
POS	74.543	0.025	5	8.238	72.870	64.632	0.00	48	W
search priority	76.831	0.025	5	7.599	75.710	68.111	0.00	48	W
smlst search & access hr	112.351	0.028	6	8.669	108.601	99.932	0.00	74	H
smlst search hrs	128.014	0.026	6	7.341	126.113	118.772	8.46	78	S
PSR	152.029	0.029	7	7.341	150.176	142.835	20.46	89	S
hgst elevation	155.734	0.035	5	26.882	151.650	124.768	20.46	86	S
smlst access hrs	-	0.024	8	9.374	-	-	81.99	93	ND
hgst degree exit vertex	-	0.022	7	22.290	-	-	82.24	93	ND

Table 12.21: Resource allocation methods for problem instance B from the interior base located at vertex 13.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
grtst hrs	9.157	0.020	0	-	8.609	-	0.00	4	W
least hrs	76.844	0.023	5	13.924	75.663	61.739	0.00	48	W
index	104.241	0.033	5	18.305	102.437	84.132	0.00	70	H
Single task									
search priority	14.620	0.024	0	-	13.665	-	0.00	8	W
PSR	21.266	0.024	0	-	18.331	-	0.00	13	W
POS	41.169	0.017	5	13.924	38.987	25.063	0.00	24	W
Path scan									
PSR	13.913	0.022	0	-	12.331	-	0.00	8	W
search priority	16.084	0.008	4	14.017	12.331	(1.686)	0.00	8	W
POS	35.875	0.010	5	13.924	30.887	16.963	0.00	20	W
Primary area									
search priority	70.427	0.019	5	9.410	66.856	57.446	0.00	45	W
POS	86.615	0.028	5	8.828	84.331	75.503	0.00	56	W
PSR	126.668	0.035	5	9.676	114.707	105.031	8.46	78	S
smlst search & access hrs	129.482	0.031	6	8.738	127.974	119.236	8.46	79	S
hgst degree exit vertex	131.008	0.024	6	9.061	129.396	120.335	8.46	81	S
smlst access hrs	131.722	0.027	6	21.775	128.703	106.928	8.46	82	S
smlst search hrs	150.308	0.034	5	9.676	132.000	122.324	20.46	83	S
hgst elevation	153.792	0.032	5	22.307	151.280	128.973	20.46	86	S

locations. This was particularly true for the exterior base where two region criteria - highest degree exit vertex and smallest access hours - failed to detect the subject, and in fact failed to allocate the *lost_region* for searching at any stage of the simulated SAR operation. In the instance where the smallest access hours criterion was used, this failure resulted from the *lost_region* never being an element of the primary search area and hence, never being a candidate for allocation. When the highest degree exit vertex criterion was used, the *lost_region* was a component in the primary search area once, near the conclusion of the search, but was not selected for allocation based on this criterion.

When clues are detected, a greater time elapses between initial clue detection and the allocation of the *lost_region* for searching, for the two methods which allocate regions for searching from a primary search area: the benchmark and primary search area methods. This is explained by the *lost_region* not featuring as a component of the primary search area until just prior to allocation, at which time its POA value has increased to a level which warrants its inclusion.

No clear advantage resulted from selecting either the interior or exterior base for all methods. However, for the majority of methods, the quickest detection time over each of the criteria tested occurred when allocating search paths emanating from the interior base. The exception was the single task method, and in this case the best detection times achieved from each base were very similar, with detection occurring at 14.557 when routing from the exterior base and at 14.620 when routing from the interior base.

When hybrid variants of the path scan method are used to generate the search paths, the search results obtained are detailed in Table 12.23 for deployment from the exterior base, and in Table 12.24 for deployment from the interior base. When analysing the search results arising from the exterior base deployment, we see that using search priority and POS as the initial generating criterion resulted in quicker detection times and fewer clues being detected than the non-hybrid method, for all secondary criteria. This was true for only some of the paths generated from the PSR criterion. However, when the time of initial allocation of the *lost_region* is considered as the performance measure, rather than the subject detection time, the paths generated by the POS criterion actually allocated the *lost_region* at a later time than the non-hybrid method.

When resources are deployed from the interior base the hybrid path generation method is also superior to the non-hybrid method for all secondary criteria under the search priority criterion and for all, but one, criteria under the POS criterion. As in the exterior base results, the PSR initial generating criterion only produced improvements over the non-hybrid method for approximately half of the secondary criteria. Those criteria that did not show improvement in the detection time of the subject did, however, allocate the *lost_region* for searching at an identical time to the non-hybrid path method.

Table 12.22: Ranking of secondary selection criteria for the hybrid path scan method for problem instance B.

Criterion	Base location		Overall Rank
	Interior base	Exterior base	
Search priority	2.0	2.3	2.2
PSR	1.3	3.0	2.2
Smallest search hours	3.0	4.0	3.5
POS	4.0	3.3	3.7
Highest degree exit vertex	3.7	3.7	3.7
Smallest access hours	5.0	2.3	3.7
Highest elevation	3.7	4.3	4.0
Smallest search and access hours	5.3	5.0	5.2

Comparison of the hybrid variants between search base locations shows a close similarity in subject detection times, with no one location detecting the subject faster over all secondary criterion.

The rankings of the secondary selection criteria for both base locations and three initial criteria are detailed in Table 12.22. The secondary criteria that result in faster subject detection are those with a priority component, which mirrors the results obtained for problem instance A.

12.6.2 Path Scan Heuristic

Further computational experiments were conducted for the path scan heuristic from the exterior base restricting the candidate region set to the primary search area ($PRIMARY\_AREA$), relaxing the constraint on the re-searching of a region within the same period ($MULTIPLE$), and varying the criteria for resource selection ($RESOURCE\_CRITERION$). The results of these experiments are detailed in Table 12.25, Table 12.26 and Table 12.27 respectively.

From Table 12.25 it is evident that the subject is detected faster when regions to be allocated for searching are not constrained to only those which comprise the primary search area. In stating this conclusively we disregard the result produced by the POS criterion without primary search area restriction due to the failure to detect the subject on the first search of the *lost_region* which was interrupted at time 18.000 when the search resource was redeployed to the search base. All paths generated from regions comprising the primary search area detected the subject on their first search of the *lost_region*, at times greatly exceeding 18.000.

When re-searching a region is permitted within the same search period, the search

Table 12.23: Hybrid resource allocation methods for problem instance B from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
PSR/									
smlst access hrs	13.210	0.022	0	-	12.255	-	0.00	8	W
smlst search hrs	13.973	0.021	0	-	13.018	-	0.00	8	W
POS	14.378	0.023	0	-	12.081	-	0.00	8	W
smlst search & access hrs	16.609	0.024	0	-	12.965	-	0.00	8	W
search priority	17.390	0.013	2	16.701	12.309	(4.392)	0.00	8	W
hgst degree exit vertex	22.601	0.025	0	-	20.084	-	0.00	12	W
hgst elevation	28.363	0.026	0	-	25.673	-	0.00	16	W
Search priority/									
hgst elevation	13.566	0.023	0	-	12.611	-	0.00	8	W
smlst access hrs	13.616	0.023	0	-	12.660	-	0.00	8	W
smlst search hrs	13.714	0.022	0	-	12.759	-	0.00	8	W
hgst degree exit vertex	13.751	0.022	0	-	12.796	-	0.00	8	W
smlst search & access hrs	13.754	0.022	0	-	12.799	-	0.00	8	W
PSR	13.760	0.023	0	-	12.805	-	0.00	8	W
POS	15.848	0.025	0	-	12.000	-	0.00	8	W
POS/									
hgst degree exit vertex	14.038	0.012	2	13.483	13.083	(0.400)	0.00	8	W
search priority	22.528	0.012	4	13.483	20.011	6.528	0.00	12	W
PSR	22.645	0.011	4	13.483	20.128	6.645	0.00	12	W
smlst access hrs	28.009	0.011	4	13.483	25.438	11.955	0.00	16	W
hgst elevation	45.530	0.019	5	13.862	42.720	28.858	0.00	27	W
smlst search & access hrs	45.597	0.022	5	13.710	36.499	22.789	0.00	28	W
smlst search hrs	45.609	0.022	5	13.483	42.644	29.161	0.00	27	W

Table 12.24: Hybrid resource allocation methods for problem instance B from the interior base located at vertex 13.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
PSR/									
search priority	13.449	0.023	0	-	12.331	-	0.00	8	W
smlst access hrs	13.694	0.022	0	-	12.576	-	0.00	8	W
hgst elevation	13.913	0.022	0	-	12.331	-	0.00	8	W
hgst degree exit vertex	13.913	0.021	0	-	12.331	-	0.00	8	W
smlst search hrs	14.886	0.022	0	-	12.331	-	0.00	8	W
POS	15.104	0.023	0	-	12.331	-	0.00	8	W
smlst search & access hrs	22.297	0.025	0	-	12.331	-	0.00	12	W
Search priority/									
hgst elevation	13.449	0.022	0	-	12.331	-	0.00	8	W
smlst search hrs	13.567	0.022	0	-	12.449	-	0.00	8	W
PSR	14.239	0.023	0	-	12.331	-	0.00	8	W
hgst degree exit vertex	15.612	0.025	0	-	12.331	-	0.00	8	W
smlst search & access hrs	15.621	0.025	0	-	12.331	-	0.00	8	W
POS	16.084	0.025	0	-	12.331	-	0.00	8	W
smlst access hrs	16.084	0.025	0	-	12.331	-	0.00	8	W
POS/									
PSR	22.712	0.012	4	13.924	20.208	6.284	0.00	12	W
smlst search hrs	27.852	0.013	4	13.924	24.917	10.993	0.00	16	W
hgst degree exit vertex	28.771	0.013	4	13.924	25.892	11.968	0.00	16	W
smlst search & access hrs	28.783	0.013	4	13.924	25.904	11.980	0.00	16	W
search priority	32.158	0.016	4	13.924	30.734	16.810	0.00	20	W
smlst access hrs	32.355	0.014	4	13.924	30.931	17.007	0.00	20	W
hgst elevation	41.400	0.017	5	13.924	36.674	22.750	0.00	24	W

results detailed in Table 12.26 are inconclusive in showing dominance of this strategy over that of no re-searching. In particular, the first time that the *lost_region* is allocated for searching had a maximum deviation of half an hour across all search strategies and all criteria tested.

Table 12.27 displays the search outcomes that arise when generating the search paths via the three different resource criteria. Once again the *lost_region* was first allocated at a similar time across all of the paths, with a maximum deviation of approximately 55 minutes. In this respect no criterion stands out as being clearly superior to the others, with the search results being further obfuscated by the number of multiple *lost_region* allocations necessary to achieve subject detection in half of the experiments. This was necessary for all of the experiments using the smallest hours criterion, where the *lost_region* was allocated as the last region in the search path and was often not searched before the resource was redeployed to the base at the end of the search period.

12.6.3 Sound Sweep Method

The sound sweep method of searching was employed in the first two periods of searching for a selection of resource allocation methods. The methods selected were those that detected the subject fastest under each of the four path generation methods for visual searching alone when resources were deployed from the interior base. This set of allocation methods consisted of: benchmark with greatest number of available search hours criterion, single task with search priority criterion, path scan with PSR criterion, and primary search area with search priority criterion. The search results arising from the use of sound sweep searching in these initial periods were compared with those when visual searching alone was used, over both weather scenarios. The results of these experiments are recorded in Table 12.28 with W1 representing weather scenario one and W2 representing weather scenario two.

The subject was responsive throughout the first two periods and, when the sound sweep method was used, was detected during these periods via sound detection in all but one case. The instance where sound detection did not occur was under the adverse weather scenario for the single task method; in this case visual detection of the subject resulted soon after weather conditions improved and searching re-commenced.

The use of sound sweeping in the first two search periods detects the subject faster than visual searching alone for all but one instance; the benchmark method under weather scenario one. In this respect the experimental results are similar to those obtained for problem instance A. The greatest improvement in subject detection times when utilizing sound detection was seen for the primary search area method. In this instance the subject

Table 12.25: Primary search area restrictions for the path scan allocation method, for problem instance B from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
No Primary Area									
PSR	15.449	0.024	0	-	12.892	-	0.00	8	W
search priority	15.885	0.010	3	14.419	12.797	(1.622)	0.00	8	W
POS *(4)	130.940	0.008	6	13.483	12.533	(0.950)	8.46	80	S
Primary Area									
PSR	74.977	0.026	5	8.271	73.615	65.344	0.00	48	W
search priority	74.543	0.025	5	8.699	73.007	64.308	0.00	48	W
POS	74.543	0.026	5	8.186	73.187	65.001	0.00	48	W

Table 12.26: Re-searching options for the path scan allocation method, for problem instance B from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
No multiple searching									
PSR	15.449	0.024	0	-	12.892	-	0.00	8	W
search priority	15.885	0.010	3	14.419	12.797	(1.622)	0.00	8	W
POS *(4)	130.940	0.008	6	13.483	12.533	(0.950)	8.46	80	S
Multiple searching									
PSR	14.747	0.023	0	-	12.898	-	0.00	8	W
search priority	17.145	0.026	0	-	12.415	-	0.00	8	W
POS *(3)	76.870	0.011	4	13.483	12.533	(0.950)	0.00	47	W

Table 12.27: Resource criteria for the path scan allocation method, for problem instance B from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Greatest hours									
PSR	15.449	0.024	0	-	12.892	-	0.00	8	W
search priority	15.885	0.010	3	14.419	12.797	(1.622)	0.00	8	W
POS *(4)	130.940	0.008	6	13.483	12.533	(0.950)	8.46	80	S
Smallest hours									
PSR *(2)	23.710	0.025	0	-	12.611	-	0.00	12	W
search priority *(2)	22.430	0.011	4	14.264	12.000	(2.264)	0.00	12	W
POS *(3)	70.147	0.011	5	13.483	12.606	(0.877)	0.00	43	W
Index									
PSR	15.404	0.024	0	-	12.848	-	0.00	8	W
search priority	17.954	0.011	4	14.381	12.000	(2.381)	0.00	8	W
POS *(4)	130.805	0.009	6	13.483	12.561	(0.922)	8.46	80	S

was detected ten times quicker under adverse search conditions by sound detection, more than five days before detection occurred via visual searching alone. For all methods visual searching alone under adverse search conditions detected the subject more slowly than under more favourable conditions.

An unusual result obtained by these experiments was the faster detection times that resulted from sound sweep searching under adverse conditions over those resulting from more favourable conditions. The *lost_region* was allocated via an alternative region strategy (using Algorithm 8.5) in the first search period, when using the benchmark method under adverse conditions, but it was not allocated until the beginning of the second search period under the more favourable conditions. The alternative region allocation resulted from all edges within the primary search area having been already searched, and the selection of the *lost_region* was determined from the search resource's position and the predicted POS values of the regions adjacent to that position. Hence, the selection of the *lost_region* for searching under this scenario was a product of the set of regions comprising the primary search area, the time needed to search each of these, the relative positions of the resources and the simulation rules developed to govern resource allocation. The result highlights the advantage of diversifying the search to regions outside of the primary search area, particularly when adverse weather conditions are predicted.

When allocating search tasks to resources under the path scan and primary search area methods the *lost_region* was allocated at the same time under both search conditions and the later detection time of one over the other was a result of the composition of the primary search area, for the primary search area method, and the resource to which the *lost_region* was allocated to and their total search allocation, for the path scan method. Thus the results, while unexpected, are a result of the interaction between differing search duration times under the different search conditions and the resource allocation rules built into the simulation model.

12.6.4 Intended Path Knowledge

When the intended route of the subject is known, as in this problem instance, the POA values of the search regions are adjusted to reflect this information by the procedure detailed in Section 5.3.5. In this problem instance the first four regions comprising the subject's actual path have vertices in common with the intended path and consequently have their historical POA values (those values originally attached to each region in the TIN) increased. The subject then moves away from the intended route, resulting in a lower POA value for the *lost_region* than if the historical POA values had been used without taking into account knowledge of the subject's intentions. Hence, it would be

Table 12.28: Comparison of selected resource allocation methods for problem instance B from the interior base located at vertex 13, under visual and sound sweep searching for both weather scenarios.

Method	Time of detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark,grtst hr									
visual sweep, W1	9.157	0.020	0	-	8.609	-	0.00	4	W
sound sweep, W2	9.722	0.020	0	-	9.401	-	0.00	4	W
visual search, W2	9.756	0.019	0	-	8.940	-	0.00	4	W
sound sweep, W1	15.397	0.027	0	-	12.331	-	0.00	8	W
Single task, search priority									
sound sweep, W1	9.182	0.022	0	-	9.007	-	0.00	4	W
visual search, W1	14.620	0.024	0	-	13.665	-	0.00	8	W
sound sweep, W2	55.925	0.025	0	-	54.500	-	40.54	12	S
visual search, W2	58.232	0.025	0	-	57.232	-	39.79	13	S
Path scan, PSR									
sound sweep, W2	10.672	0.021	0	-	6.331	-	0.00	4	W
sound sweep, W1	11.053	0.021	0	-	6.331	-	0.00	4	W
visual search, W1	13.913	0.022	0	-	12.331	-	0.00	8	W
visual search, W2	59.404	0.025	0	-	54.500	-	38.73	12	S
Primary area, search priority									
sound sweep, W2	13.269	0.022	0	-	12.331	-	0.00	8	W
sound sweep, W1	17.409	0.023	1	8.867	12.331	3.464	0.00	8	W
visual search, W1	126.668	0.035	5	9.676	114.707	105.031	8.46	78	S
visual search, W2	143.847	0.029	2	8.784	140.409	131.625	94.33	41	H

anticipated, that if the searches were simulated again with the subject's intended route not being taken into account, the subject would in fact be detected faster. The search results from the exterior base (vertex 27) under these conditions can be found in Table 12.20. It can be seen from comparing these results with those of Table 12.20 that for all path generation methods but one (benchmark with least hours), the subject is, indeed, detected faster when the subject's intended route information is not taken into consideration.

The single task method again performs the best of the four methods and the primary search area method, which is highly dependent on the POA distribution in the formation of the primary search area, performs markedly better for all but the POS region criterion. The paths generated via the primary search area method with smallest search hours criterion are in fact almost ten times quicker than when intended knowledge is incorporated. The subject is detected by all methods within three days compared with the six days needed by the slowest detection method when planning the search based on known intention. In general, fewer clues are detected without route knowledge, which is also to be expected.

When the intentions of the subject are not considered, the interior base location shifts from vertex 13 to vertex 0. The results from these simulated searches are detailed in Table 12.30 and again show improvements over those searches planned with respect to the intended path, for all but the benchmark method with greatest hours. Marked improvements in subject detection time are again seen for the primary search area method, with the greatest improvement being a factor of 12.5 for the smallest search hour criterion. The subject is detected within two days by all methods, compared with the six days needed to detect the subject, in the worse case, by those methods incorporating intended route knowledge. The search results are also superior to those achieved under the same planning information from the exterior base.

As the interior base location is shifted when the POA values are not adjusted to account for the subject's intended path, experiments were also conducted fixing the base at the original location, vertex 13. The results of these trials are recorded in Table 12.31. When these results are compared with the results from base 13 that take into account the subject's intentions, it can be seen that the subject is detected faster by the majority of path generation heuristics. The fastest time, over all allocation methods, is still found under knowledge of the intended route, for the benchmark method with greatest hours. However, this is an anomaly in the context of all results, with the fastest detection time for all other methods deploying from base 13 being found without intended route knowledge. The worst case detection times under these two search planning scenarios are almost equal at approximately 153 hours. Again, fewer clues are detected under no route knowledge.

Comparisons can also be drawn between the two interior base locations when the subject's intentions are not considered. Such comparisons reveal an overall superiority of the base location at vertex 0, where the subject was detected within two days by all methods. However, the benchmark method performs slightly better for paths generated from vertex 13. The most significant differences in detection times were observed for the primary search area method, with quicker detection times resulting when positioning the base at vertex 0, including the fastest subject detection time of 8.983 for the search priority criterion. The faster detection times that resulted from paths generated from vertex 0 validates the base positioning algorithm (detailed in Section 11.2.1) which selected this location over vertex 13, based on the POA distribution and shortest path structure of the TIN.

12.6.5 Method Comparison

We now compare the relative performance of each of the four resource allocation methods and the hybrid variants of the path scan method. Relative performance ratios are calculated as for problem instance A. The best solution for this problem instance is the detection of the subject at 9.157 by paths generated by the benchmark method with the greatest remaining resource hours criterion. This result was achieved when deploying resources from the interior base. When resources are deployed from the exterior base, the best solution obtained is the detection of the subject at 13.210 by paths generated by the hybrid path scan method. This method creates the first portion of the search paths via the PSR criterion and the second portion of the search paths via the smallest access hours criterion. These two results are used as the value of the best solution in calculating the performance ratios for resource deployment from the interior and exterior search base locations, respectively. The performance ratios of each resource allocation method and base location are presented in Table 12.32 and Table 12.33.

The hybrid path scan method with search priority criterion was the best performer for this problem instance, with low best, median and worst performance ratios for both base locations as well as having the least variation in detection times. The hybrid path scan method with PSR criterion also performed well, while the search priority region selection criterion achieved the best results over all methods.

The primary search area method was the poorest performer of the methods tested having the highest best, median and worst performance ratios irrespective of base location, and the highest variation in detection times, including the only subject non-detection, of all methods. The worst performance of the method when successful detection did occur

Table 12.29: Resource allocation methods for problem instance B from the exterior base located at vertex 27, when no intended route knowledge is known.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
index	14.977	0.016	2	13.822	12.000	(1.822)	0.00	8	W
grtst hrs	15.712	0.016	2	14.468	12.000	(2.468)	0.00	8	W
least hrs	74.543	0.028	6	9.059	72.642	63.583	0.00	48	W
Single task									
search priority	12.955	0.033	0	-	12.000	-	0.00	8	W
POS	12.955	0.029	0	-	12.000	-	0.00	8	W
PSR	15.892	0.034	0	-	15.282	-	0.00	8	W
Path scan									
PSR	14.069	0.032	0	-	12.034	-	0.00	8	W
search priority	14.579	0.034	0	-	12.000	-	0.00	8	W
POS	17.211	0.014	4	13.483	12.000	(1.483)	0.00	8	W
Primary area									
smlst search hrs	12.955	0.015	2	12.744	12.000	(0.744)	0.00	8	W
smlst search & access hrs	13.257	0.031	0	-	12.301	-	0.00	8	W
search priority	14.380	0.016	2	13.483	12.000	(1.483)	0.00	8	W
hgst degree exit vertex	26.758	0.030	0	-	25.339	-	0.00	16	W
hgst elevation	28.781	0.030	0	-	27.001	-	0.00	16	W
smlst access hrs	52.176	0.024	5	8.232	48.840	40.608	0.00	32	W
POS	69.872	0.024	6	8.236	66.497	58.261	0.00	44	W
PSR	74.543	0.026	6	7.341	72.321	64.980	0.00	48	W

Table 12.30: Resource allocation methods for problem instance B from the interior base located at vertex 0, when no intended route knowledge is known.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
grtst hrs	16.052	0.031	0	-	12.556	-	0.00	9	W
least hrs	16.972	0.014	4	14.150	12.556	(1.594)	0.00	9	W
index	16.972	0.016	2	15.645	12.556	(3.089)	0.00	10	W
Single task									
POS	10.399	0.028	0	-	9.350	-	0.00	4	W
search priority	11.491	0.033	0	-	9.982	-	0.00	5	W
PSR	16.044	0.033	0	-	14.833	-	0.00	10	W
Path scan									
POS	10.399	0.028	0	-	6.556	-	0.00	4	W
search priority	11.572	0.032	0	-	6.556	-	0.00	4	W
PSR	13.822	0.032	0	-	12.972	-	0.00	8	W
Primary area									
search priority	8.983	0.023	0	-	6.556	-	0.00	4	W
PSR	10.996	0.013	2	10.445	6.556	(3.889)	0.00	4	W
POS	11.360	0.013	2	10.552	6.556	(3.996)	0.00	4	W
smlst search & access hrs	11.821	0.028	0	-	6.556	-	0.00	4	W
smlst search hrs	12.004	0.029	0	-	6.556	-	0.00	4	W
smlst access hrs	12.290	0.009	4	8.375	6.556	(1.819)	0.00	4	W
hgst elevation	28.227	0.031	0	-	25.246	-	0.00	16	W
hgst degree exit vertex	46.135	0.036	0	-	43.490	-	0.00	28	W

Table 12.31: Resource allocation methods for problem instance B from the interior base fixed at vertex 13, when no intended route knowledge is known.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
least hrs	14.698	0.016	2	13.924	12.335	(1.589)	0.00	8	W
index	14.733	0.016	2	13.924	12.331	(1.593)	0.00	8	W
grtst hrs	15.831	0.032	0	-	12.331	-	0.00	8	W
Single task									
POS	10.319	0.029	0	-	9.614	-	0.00	4	W
search priority	13.918	0.032	0	-	13.225	-	0.00	8	W
PSR	16.316	0.033	0	-	15.467	-	0.00	8	W
Path scan									
PSR	13.449	0.031	0	-	12.331	-	0.00	8	W
search priority	14.698	0.034	0	-	12.331	-	0.00	8	W
POS	46.239	0.015	5	13.924	12.331	(1.593)	0.00	28	W
Primary area									
search priority	13.918	0.032	0	-	12.331	-	0.00	8	W
smlst search & access hrs	14.333	0.032	0	-	12.331	-	0.00	8	W
smlst search hrs	15.956	0.017	2	14.151	12.331	(1.820)	0.00	8	W
PSR	15.988	0.031	0	-	12.331	-	0.00	8	W
hgst degree exit vertex	22.223	0.030	0	-	18.873	-	0.00	12	W
POS	70.414	0.030	5	8.002	66.839	58.837	0.00	44	W
hgst elevation	130.857	0.042	6	22.105	129.756	107.651	8.46	80	S
smlst access hrs	153.473	0.042	6	9.218	150.502	141.284	20.46	88	S

Table 12.32: Performance ratios for visual search resource allocation methods for problem instance B when deploying from the exterior base.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Benchmark	4.457	5.814	6.117	21.935
Single task	1.102	1.289	2.984	24.861
Path scan	1.169	1.202	9.912	115.491
Primary area	5.643	10.600	-	-
Hybrid - PSR	1.000	1.257	2.147	15.153
Hybrid - search priority	1.027	1.041	1.200	2.282
Hybrid - POS	1.063	2.120	3.453	31.571

Table 12.33: Performance ratios for visual search resource allocation methods for problem instance B when deploying from the interior base.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Benchmark	1.000	8.392	11.384	95.084
Single task	1.597	2.322	4.496	26.549
Path scan	1.519	1.756	3.918	21.962
Primary area	7.691	14.224	16.795	83.365
Hybrid - PSR	1.469	1.519	2.435	8.848
Hybrid - search priority	1.469	1.705	1.756	2.635
Hybrid - POS	2.480	3.143	4.521	18.688

was the detection of the subject at a time sixteen times slower than the best solution. The benchmark method also performs poorly overall, with relatively high median and worst performance ratios from both base locations. The obvious exception to this general performance is the method's detection of the subject in the fastest time from the interior base. This resulted from the search strategy of allocating an adjacent region to a resource for searching when no candidates were available for selection from the primary search area. In this respect the quality of the solution achieved is a result of this strategy rather than resource allocation by the benchmark method. The methods that restricted the set of regions which could be allocated for searching to a primary search area performed poorly in comparison to those which did not.

12.7 Problem Instance C

The third problem instance incorporates a subject moving by a random path, without confusion. The path of the subject is generated by the parameters:

- $PATHTYPE = random$;

- $CONFUSION$ = FALSE;
- $SPATHSTART = 4$;
- $SPATHLENGTH = 4$.

The path begins from the exterior of the TIN at vertex 4 and moves through six regions; two edges representing streams and four triangles of density classification 4. Specifically, the path is described by:

vertex path:	4 - 6 - 19 - 6 - 4 - 6 - 8
region path:	55 - 148 - 169 - 113 - 147 - 47
initial POA value:	0.000 - 0.016 - 0.015 - 0.000 - 0.006 - 0.000

where the region labels identify: stream edge (4,6), triangle (6,22,19), triangle (19,36,6), triangle (4,6,8), triangle (4,6,29) and stream edge (6,8).

The region in which the subject is located in, edge region 47, is a predominantly shingle stream bed with a density classification of 2 and a length of 712m. The average time needed to search the edge under ideal search conditions is 39 minutes. Inorganic clues are positioned within regions 148 and 113, and comprise 4% of the total number of simulated clue placements. The search operation is conducted under both the weather conditions described in weather file one and those conditions described in weather file two. The subject is initially unresponsive. The intended route of the subject is unknown, hence, the search base selected under the external base criterion is located at vertex 27 and the base location interior to the convex hull is located at vertex 0.

12.7.1 General Results for Problem Instance C

The search results for each resource allocation method under weather scenario one are detailed in Table 12.34 and Table 12.36 for an externally and internally positioned search base, respectively. When the search is conducted under less favourable weather conditions (weather scenario two) the resulting performance measures are detailed in Table 12.35 and Table 12.37 for an externally and internally positioned search base, respectively. The subject was detected by all methods except for the benchmark method under the greatest and least remaining search hours resource selection criteria, in adverse conditions from the interior base, and the primary search area method with highest elevation selection criterion, from the exterior base under weather scenario one. As the subject was unconscious the retrieval method was via helicopter when the subject was detected in favourable conditions, and via stretcher-carry otherwise.

It can be seen that no clues at all are detected under the adverse weather scenario. Under these search conditions the organic clues degrade, reducing their detectability, and the attainable POD level of the subject, and consequently the clues, is also reduced. Few clues are also detected under the more favourable weather scenario, and when this did eventuate, it was often after the *lost_region* was scheduled for searching, with the clue being detected within the *lost_region* prior to subject detection. The primary search area method was the only resource allocation method to detect clues outside the region in which the subject was located, with the *lost_region* being scheduled for searching within 2 hours of such a detection.

Conducting the search operation from the exterior base resulted in the best overall subject detection time for each resource allocation method, under both weather scenarios. Over all of the resource allocation methods, the primary search area method, despite detecting clues outside of the *lost_region*, was consistently the worst performer. However, it was also the method that exhibited the most consistent detection results over the individual region selection criteria tested within each method.

The index ordering resource selection criterion resulted in the best detection times for the benchmark method and performed equivalently to the greatest remaining search hours criterion for all experiments but deployment from the interior base under adverse search conditions. The selection of the region with greatest PSR value was the region criterion that resulted in the fastest detection of the subject over the single task and path scan allocation methods.

An analysis of Table 12.34 through to Table 12.37 results in the ranking of region criteria for the primary search area method displayed in Table 12.38. The access and search duration criteria, on the whole, performed better than the priority based criteria. This observation is similar to that obtained for problem instance A, with the exception of the search priority criterion, but dissimilar to problem instance B where the priority based criteria detected the subject more quickly.

Utilizing a night searching strategy results in no search down time under weather scenario one. Under weather scenario two, however, down time occurs for all cases where the subject was not detected prior to 16.543, the time that weather conditions deteriorate to level 4. Two hours prior to the onset of these conditions search resources are recalled to the base and remain inactive until conditions improve. Visibility is also hampered under the difficult conditions simulated and searching is not executed during periods of limited visibility, such as the time between nightfall and dawn.

When hybrid variants of the path scan method are used to allocate search tasks from the exterior base under both weather scenarios, the results are displayed in Table 12.39 and Table 12.40. The ranking of the secondary region selection criteria is the same for the POS initial path criterion, across the two weather scenarios, and very similar for the search priority criterion. The PSR initial criterion returns identical results for all

Table 12.34: Comparison of resource allocation methods for problem instance C under weather scenario one, from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at completion	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
index	11.964	0.203	1	11.788	11.589	(0.199)	0.00	6	H
grtst hrs	12.332	0.203	1	12.156	11.958	(0.198)	0.00	6	H
least hrs	60.331	0.000	0	-	59.883	-	0.00	35	H
Single task									
PSR	7.838	0.000	0	-	7.174	-	0.00	4	H
POS	28.564	0.000	0	-	27.494	-	0.00	16	H
search priority	60.557	0.000	0	-	60.067	-	0.00	36	H
Path scan									
PSR	7.838	0.000	0	-	6.000	-	0.00	4	H
POS	11.771	0.205	1	11.595	11.462	(0.133)	0.00	4	H
search priority	16.427	0.000	0	-	12.000	-	0.00	8	H
Primary area									
smlst search hrs	36.622	0.200	1	36.446	35.653	(0.793)	0.00	20	H
smlst access hrs	37.712	0.000	0	-	36.554	-	0.00	24	H
PSR	37.984	0.000	0	-	36.509	-	0.00	24	H
POS	42.985	0.205	1	42.659	41.992	(0.667)	0.00	24	H
smlst search & access hr	43.265	0.202	1	43.023	41.864	(1.159)	0.00	24	H
hgst degree exit vertex	60.584	0.000	0	-	59.963	-	0.00	38	H
search priority	66.447	0.010	2	64.812	66.033	1.221	0.00	40	H
hgst elevation	-	0.000	0	-	-	-	81.56	92	ND

Table 12.35: Comparison of resource allocation methods for problem instance C under weather scenario two, from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
grtst hrs	12.239	0.000	0	-	12.033	-	0.00	6	S
index	12.239	0.000	0	-	12.033	-	0.00	6	S
least hrs	133.136	0.000	0	-	132.232	-	91.20	30	H
Single task									
PSR	7.839	0.000	0	-	7.174	-	0.00	4	S
search priority	129.430	0.000	0	-	129.227	-	92.27	29	H
POS	135.226	0.000	0	-	134.512	-	92.62	35	H
Path scan									
PSR	7.839	0.000	0	-	6.000	-	0.00	4	S
POS	11.722	0.000	0	-	11.479	-	0.00	6	S
search priority	60.278	0.000	0	-	54.500	-	38.83	12	S
Primary area									
smlst search hrs	61.136	0.000	0	-	60.232	-	38.63	12	S
search priority	84.577	0.000	0	-	84.174	-	50.96	21	S
hgst elevation	85.042	0.000	0	-	84.045	-	51.12	20	S
smlst search & access hrs	133.136	0.000	0	-	132.232	-	91.40	30	H
POS	133.136	0.000	0	-	132.232	-	91.70	30	H
PSR	133.136	0.000	0	-	132.232	-	92.46	30	H
smlst access hrs	135.804	0.000	0	-	133.187	-	91.74	35	H
hgst degree exit vertex	158.066	0.000	0	-	157.552	-	91.83	51	H

Table 12.36: Comparison of resource allocation methods for problem instance C under weather scenario one, from the interior base located at vertex 0.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
index	12.419	0.197	1	12.243	12.045	(0.198)	0.00	9	H
grtst hrs	12.420	0.203	1	12.244	12.152	(0.092)	0.00	8	H
least hrs	43.773	0.201	1	43.531	42.656	(0.875)	0.00	26	H
Single task									
PSR	8.174	0.000	0	-	7.361	-	0.00	4	H
search priority	10.250	0.204	1	10.075	9.982	(0.093)	0.00	4	H
POS	44.830	0.000	0	-	43.907	-	0.00	28	H
Path scan									
PSR	8.174	0.000	0	-	6.556	-	0.00	4	H
POS	12.521	0.000	0	-	6.556	-	0.00	4	H
search priority	23.998	0.000	0	-	12.816	-	0.00	12	H
Primary area									
hgst degree exit vertex	25.278	0.120	1	25.036	24.589	(0.447)	0.00	14	H
smlst search hrs	43.181	0.224	3	40.847	42.556	1.709	0.00	25	H
smlst search & access hrs	43.379	0.201	2	41.418	42.623	1.205	0.00	26	H
POS	54.678	0.000	0	-	54.338	-	0.00	33	H
smlst access hrs	54.790	0.000	0	-	54.479	-	0.00	35	H
search priority	55.485	0.000	0	-	54.866	-	0.00	36	H
PSR	55.806	0.000	0	-	55.187	-	0.00	36	H
hgst elevation	73.141	0.000	0	-	72.589	-	0.00	47	H

Table 12.37: Comparison of resource allocation methods for problem instance C under weather scenario two, from the interior base located at vertex 0.

Method	Time of visual detection	POA_{lost_region} at completion	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Benchmark with NS									
index	12.229	0.000	0	-	11.836	-	0.00	6	S
grtst hrs	-	0.000	0	-	-	-	65.54	23	ND
least hrs	-	0.000	0	-	-	-	65.54	22	ND
Single task									
PSR	8.175	0.000	0	-	7.361	-	0.00	4	S
search priority	127.120	0.000	0	-	126.500	-	95.47	30	S
POS	132.842	0.000	0	-	132.533	-	96.18	32	S
Path scan									
PSR	8.175	0.000	0	-	6.556	-	0.00	4	S
search priority	55.656	0.000	0	-	54.500	-	40.79	11	S
POS	84.502	0.000	0	-	84.101	-	53.43	21	S
Primary area									
smlst search & access hr	132.704	0.000	0	-	132.165	-	94.59	31	S
smlst access hrs	133.129	0.000	0	-	132.506	-	94.61	33	S
POS	133.435	0.000	0	-	132.815	-	95.11	34	S
search priority	133.718	0.000	0	-	132.843	-	94.11	32	S
PSR	140.710	0.000	0	-	139.543	-	95.60	34	H
hgst elevation	152.055	0.000	0	-	151.543	-	95.48	52	H
hgst degree exit vertex	152.055	0.000	0	-	151.543	-	96.14	52	H
smlst search hrs	157.737	0.000	0	-	151.543	-	96.36	52	H

Table 12.38: Ranking of region criteria under the primary search area method for problem instance C.

Criterion	Rank for weather scenario one	Rank for weather scenario two	Overall rank
Smallest search hours	1.5	4.0	2.8
Smallest search and access hours	4.0	2.5	3.3
Smallest access hours	3.5	4.5	4.0
POS	4.0	4.0	4.0
Search priority	6.5	3.0	4.8
Highest degree exit vertex	3.5	7.0	5.3
PSR	5.0	5.5	5.3
Highest elevation	8.0	4.5	6.3

secondary criteria, with these search results being identical to the non-hybrid variant. Again, no clues were detected under weather scenario two and the only clues detected under weather scenario one were found within the lost region itself, just prior to subject detection.

The criteria used to select the second portion of the search path were ranked in order of subject detection time. These ranks are recorded in Table 12.41. In every case, the fastest hybrid variant of each of the path construction methods detected the subject either quicker than, or in identical time (PSR criterion) to, the non-hybrid variant. This improvement in detection time is most significant for the search priority criterion under adverse weather conditions, where the detection time of the fastest hybrid path was almost six times faster. The highest elevation and highest degree exit vertex criteria were the poorest performing secondary criteria and often failed to improve upon the non-hybrid path scan method. The majority of secondary criteria did generate search paths that resulted in improved subject detection times, with the smallest search hours criterion performing the best for this problem instance. Consistent with problem instances A and B, the priority-based criteria performed well.

12.7.2 Weather Scenarios

When the search is conducted under the adverse weather conditions described by the second weather scenario, the search results are quite distinct from those when good search conditions exist, and do not merely represent a multiplicative scaling in search duration time.

When the search results for good weather conditions are examined it can be seen that all subject detections occurred under weather level 2 conditions and prior to the

Table 12.39: Comparison of hybrid resource allocation methods for problem instance C under weather scenario one, from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Search priority/									
smlst search & access hrs	9.984	0.203	1	9.809	6.000	(3.809)	0.00	4	H
smlst search hrs	11.034	0.000	0	-	6.000	-	0.00	4	H
POS	14.726	0.000	0	-	12.266	-	0.00	8	H
smlst access hrs	15.667	0.000	0	-	12.178	-	0.00	8	H
PSR	23.132	0.200	1	22.890	19.187	(3.703)	0.00	12	H
hgst elevation	32.698	0.201	1	32.439	32.306	(0.133)	0.00	18	H
hgst degree exit vertex	44.576	0.000	0	-	12.000	-	0.00	28	H
PSR/									
POS	7.838	0.000	0	-	6.000	-	0.00	4	H
search priority	7.838	0.000	0	-	6.000	-	0.00	4	H
smlst search & access hrs	7.838	0.000	0	-	6.000	-	0.00	4	H
smlst access hrs	7.838	0.000	0	-	6.000	-	0.00	4	H
smlst search hrs	7.838	0.000	0	-	6.000	-	0.00	4	H
hgst elevation	7.838	0.000	0	-	6.000	-	0.00	4	H
hgst degree exit vertex	7.838	0.000	0	-	6.000	-	0.00	4	H
POS/									
smlst search hrs	9.743	0.000	0	-	6.000	-	0.00	4	H
PSR	10.163	0.000	0	-	6.000	-	0.00	4	H
smlst search & access hrs	10.571	0.000	0	-	6.000	-	0.00	4	H
search priority	10.802	0.000	0	-	6.000	-	0.00	4	H
smlst access hrs	22.258	0.000	0	-	19.316	-	0.00	12	H
hgst degree exit vertex	30.494	0.205	1	30.289	30.110	(0.179)	0.00	17	H
hgst elevation	32.244	0.205	1	32.068	31.852	(0.216)	0.00	18	H

Table 12.40: Comparison of hybrid resource allocation methods for problem instance C under weather scenario two, from the exterior base located at vertex 27.

Method	Time of visual detection	POA_{lost_region} at detection	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
Search priority/									
smlst search & access hrs	10.062	0.000	0	-	6.000	-	0.00	4	S
smlst search hrs	10.089	0.000	0	-	6.000	-	0.00	4	S
PSR	14.485	0.000	0	-	12.000	-	0.00	8	S
POS	57.068	0.000	0	-	54.500	-	38.83	12	S
smlst access hrs	57.068	0.000	0	-	54.500	-	38.72	12	S
hgst degree exit vertex	57.383	0.000	0	-	54.500	-	39.00	12	S
hgst elevation	57.408	0.000	0	-	54.500	-	38.88	12	S
PSR/									
POS	7.839	0.000	0	-	6.000	-	0.00	4	S
search priority	7.839	0.000	0	-	6.000	-	0.00	4	S
smlst search & access hrs	7.839	0.000	0	-	6.000	-	0.00	4	S
smlst access hrs	7.839	0.000	0	-	6.000	-	0.00	4	S
smlst search hrs	7.839	0.000	0	-	6.000	-	0.00	4	S
hgst elevation	7.839	0.000	0	-	6.000	-	0.00	4	S
hgst degree exit vertex	7.839	0.000	0	-	6.000	-	0.00	4	S
POS/									
smlst search hrs	9.744	0.000	0	-	6.000	-	0.00	4	S
PSR	10.163	0.000	0	-	6.000	-	0.00	4	S
smlst search & access hrs	10.572	0.000	0	-	6.000	-	0.00	4	S
search priority	10.914	0.000	0	-	6.000	-	0.00	4	S
smlst access hrs	64.053	0.000	0	-	61.440	-	38.86	16	S
hgst degree exit vertex	78.944	0.000	0	-	78.000	-	49.57	16	S
hgst elevation	134.779	0.000	0	-	133.619	-	90.93	32	H

Table 12.41: Rankings of secondary selection criteria for the hybrid path scan method from the exterior base located at vertex 27.

Criterion	Weather Scenario		Overall Rank
	One	Two	
Smallest search hours	1.3	1.3	1.3
POS	1.3	1.7	1.5
Search priority	1.7	1.7	1.7
Smallest search and access hours	1.7	1.7	1.7
PSR	2.3	1.7	2.0
Smallest access hours	3.3	3.3	3.3
Highest degree exit vertex	4.7	4.3	4.5
Highest elevation	4.7	5.0	4.8

simulated weather deterioration at 74.54 hours. When searching is conducted under the more changeable and extreme weather conditions described by the second weather scenario, the subject was detected under weather level 3 conditions in the majority of cases. In the remainder of cases the subject was detected between 139.54 and 166.54 hours, under weather level 2 conditions. The subject remained undetected during the periods of extreme weather (level 4) as searching was suspended until search conditions improved.

It is interesting to note that, even under the scenario of adverse weather conditions, some heuristic methods detected the subject slightly earlier than under the more favourable weather scenario, *i.e.*, detection occurred earlier in level 3 weather conditions than in level 2 weather conditions. As the heuristic governing the selection and allocation of search regions for searching is identical for both scenarios and the traversal times are a factor of one another, this result may appear unusual. In general, if the allocation method used detected the subject prior to weather conditions deteriorating to level 4 at 16.54, under the second weather scenario, the time to detection was quicker or practically identical to the detection time under better weather conditions. This observation was valid for either base location. In those instances where detection occurred at an earlier time and under more difficult search conditions, this behaviour is explained by a combination of two factors - the allocation of the *lost_region* as a component in a redeployment path and the detection of a clue under favourable weather conditions. In experiments where the *lost_region* was allocated as an edge in a path redeploying the resource to the base at the end of their useful searching hours, this occurs at a similar time under both weather scenarios. When a clue was detected within the *lost_region*, prior to the subject, under favourable search conditions, a delay was encountered whilst this fact was communicated to search management.

A much larger range in subject detection times is seen under adverse weather conditions. This range is equal to 150.2 hours from the exterior base and 149.6 hours from the interior base. These ranges are more than twice the ranges observed under more favourable search conditions (*i.e.,* 58.7 hours from the exterior base and 65.0 hours from the interior base). Such differences are largely accounted for by the amount of search down time required under each scenario. When the weather is favourable the search is continuous, while the search is temporarily suspended when the weather conditions reach level 4 or light conditions are poor (level 3 weather conditions during the night period), under the less favourable weather scenario. The duration of these suspensions are recorded in Table 12.37 and Table 12.35, and can result in the loss of up to 96.36 hours of searching.

When the search down time is subtracted from the time of subject detection the results are recorded in Table 12.42 and Table 12.43 for each base location. Methods that failed to detect the subject under either scenario are not included. These results record the actual search time that elapsed before the subject was detected. When compared with the search time that elapsed before detection under the conditions described by weather scenario one, fewer searching hours were needed in the majority of instances by the second weather scenario. This result is unexpected given that searching under this scenario is executed under less favourable conditions, or exactly the same conditions for searches exceeding time 139.5, and with some regions flooded, limiting access routes.

These results can be partly explained by the fact that searching executed under the extreme conditions (prior to 139.5) does not extend into the night, while searching under the more favourable conditions does. Lower relative POD levels are applied to searching during the night and the predicted POS values are subsequently less. In each instance where the search duration until subject detection was less for weather scenario two and the actual time of that detection was greater than under weather scenario one, night searching was utilized. However, the magnitude of some differences in the search duration time between the two weather scenarios and the failure to detect the subject under one set of search conditions whilst detection occurs under the other set, is not explained solely by the ability to search at night. In such instances it would appear that subject detection is instead dependent upon whether or not a search resource requires deployment from a position on the TIN that results in the redeployment of the resource through the *lost_region.*

The PSR region criterion is the criterion that detects the subject the fastest under adverse weather conditions for both the single task and path scan methods. The superiority of this criterion is particularly evident for the single task method where the subject is detected in two to three hours from the commencement of the operation, compared to

Table 12.42: Comparison of subject detection times between weather scenarios when deployment is from the exterior base located at vertex 27.

Search Method	Search Duration Until Subject Detection		Difference Factor
	Scenario One	Scenario Two	
benchmark, NS, index	11.964	12.239	1.023
benchmark, NS, grtst hrs	12.332	12.239	0.992
benchmark, least hrs	60.331	41.937	0.695
single task, PSR	7.838	7.839	1.000
single task, POS	28.564	42.606	1.492
single task, search priority	60.557	37.160	0.614
path scan, PSR	7.838	7.839	1.000
path scan, POS	11.771	11.722	0.996
path scan, search priority	16.427	21.448	1.306
primary area, smlst search hrs	36.622	22.506	0.615
primary area, smlst access hrs	37.712	44.064	1.168
primary area, PSR	37.984	40.676	1.071
primary area, POS	42.985	41.436	0.964
primary area, smlst search & access hrs	43.265	41.736	0.965
primary area, hgst degree exit vertex	60.584	66.236	1.093
primary area, search priority	66.447	33.617	0.506

Table 12.43: Comparison of subject detection times between weather scenarios when deployment is from the interior base located at vertex 0.

Search Method	Search Duration Until Subject Detection		Difference Factor
	Scenario One	Scenario Two	
benchmark, NS, index	12.419	12.229	0.985
single task, PSR	8.174	8.175	1.000
single task, search priority	10.250	31.650	3.088
single task, POS	44.830	36.662	0.818
path scan, PSR	8.174	8.175	1.000
path scan, POS	12.521	31.072	2.482
path scan, search priority	23.998	14.866	0.619
primary area, hgst degree exit vertex	25.278	55.915	2.212
primary area, smlst search hrs	43.181	61.377	1.421
primary area, smlst search & access hrs	43.379	38.114	0.879
primary area, POS	54.678	38.325	0.701
primary area, smlst access hrs	54.790	38.519	0.703
primary area, search priority	55.485	39.608	0.714
primary area, PSR	55.806	45.110	0.808
primary area, highest elevation	73.141	56.575	0.773

the greater than five days of searching required by the other criteria. The PSR criterion does not perform as well when a primary search area is defined. In this instance no criterion performs significantly better over both base locations. The utilization of the index resource criterion is clearly superior for the benchmark allocation method.

12.7.3 Method Comparison

The relative performance of each of the four resource allocation methods is now compared for each base location and weather scenario. The performance of the hybrid path methods is included in these comparisons when resources are deployed from the exterior base. Best, median, and worst performance ratios are calculated, as in the previous two problem instances.

The quickest time to subject detection under both weather scenarios was found when resource allocation was performed by three methods - the single task method with PSR criterion, the path scan method with PSR criterion, and the hybrid path scan method with PSR as the initial construction criterion. Under weather scenario one the fastest detection time was 7.838 from the exterior base and 8.174 from the interior base. Under weather scenario two the best solution was at time 7.839 from the exterior base and 8.175 from the interior base.

The performance ratios for each base location under weather scenario one are presented in Table 12.44 and Table 12.45. Table 12.46 and Table 12.47 detail the performance ratios for each base location under weather scenario two. From these tables it is evident that the best worst case performance occurs for the method with the fastest best solution, while the resource allocation method that has the worst best solution generally also performs the worst in the worst case. As would be expected, the worst case performance ratios are greater under adverse weather conditions. When the four basic path methods are ranked in order of best solution, the same ranking occurs independent of base location and weather scenario. When ranked, the single task and path scan methods perform equally well, followed by the benchmark method and then the primary search area method (which has the worst best solution in all instances).

Although the single task method and the path scan method achieve the same best solution, the path scan method performs better overall, with both lower median and worst performance ratios, and less total variation in detection time. The hybrid method with PSR initial criterion is the best performer across all search scenarios, having a best, median and worst performance ratio all equal to 1.000. The primary search area method performs the most poorly when considering all performance ratios. The hybrid path scan method with initial POS selection criterion also performs well when considering the best

Table 12.44: Performance ratios for visual search resource allocation methods for problem instance C, when deploying from the exterior base under weather scenario one.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Benchmark	1.526	1.573	7.697	48.367
Single task	1.000	3.644	7.726	52.719
Path scan	1.000	1.502	2.096	8.589
Primary area	4.672	5.502	-	-
Hybrid - search priority	1.274	1.999	5.687	34.592
Hybrid - POS	1.243	1.378	4.114	22.501
Hybrid - PSR	1.000	1.000	1.000	0.000

Table 12.45: Performance ratios for visual search resource allocation methods for problem instance C, when deploying from the interior base under weather scenario one.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Benchmark	1.519	1.519	5.355	31.354
Single task	1.000	1.254	5.484	36.656
Path scan	1.000	1.532	2.936	15.824
Primary area	3.092	6.696	8.948	47.863

Table 12.46: Performance ratios for visual search resource allocation methods for problem instance C, when deploying from the exterior base under weather scenario two.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Benchmark	1.561	1.561	16.984	120.897
Single task	1.000	16.511	17.250	127.387
Path scan	1.000	1.495	7.690	52.439
Primary area	7.799	16.984	20.164	96.930
Hybrid - search priority	1.284	7.280	7.323	47.346
Hybrid - POS	1.243	1.392	17.193	125.035
Hybrid - PSR	1.000	1.000	1.000	0.000

Table 12.47: Performance ratios for visual search resource allocation methods for problem instance C, when deploying from the interior base under weather scenario two.

Method	Best Ratio	Median Ratio	Worst Ratio	Range
Benchmark	1.496	-	-	-
Single task	1.000	15.550	16.250	124.667
Path scan	1.000	6.808	10.337	76.327
Primary area	16.233	16.785	19.295	25.033

Table 12.48: Clue placement and detectability for problem instance A.

Region	Subject exit vertex	Number of clues	Number detectable
0	2	25	6
141	29	13	3
171	19	12	0

and median performance ratios rather than the worst case performance of the method. Under weather scenario two the best solution of each method, except the primary search area method, detected the subject prior to weather deterioration and search suspension.

In summary, the hybrid path scan method with PSR criterion produces the most superior results for this problem instance, with the PSR region selection criterion performing well for all allocation methods not restricting region selection to a primary search area. The path scan method is the next best performer over all scenarios. Both methods utilizing a primary search area performed relatively poorly in comparison, with the primary search area method being the worst of all methods tested for this problem instance. In the worst case the method's performance was twenty times worse than the best solution found.

12.8 Clue Analysis

As outlined in Section 5.3.1, clues are placed within the regions that the subject moves through, in proportion to the time that he spends in each. Clue detectability is then determined in consideration to the POD level of the search, the variable $pod\_fraction$, the type of clue, the past weather conditions, the elapsed time since the clue was formed and a pseudo random number input stream, as detailed in Section 5.3.3. When triangular regions are searched to a 50% POD level and edge regions are searched to a 100% POD level, past weather conditions are good and $pod\_fraction = 0.2$, the number of clues which can be detected by searchers within each region along the subject's path are detailed in Tables 12.48 - 12.50, for each problem instance. Under these searching conditions 18% of clues are detectable for problem instance A, 12% for problem instance B and 14% for problem instance C.

Upon initial observation, it would seem unusual that those resource allocation methods resulting in a greater number of clues being detected, detected the subject at a later time (if at all) than those methods which detected few or no clues. These results appear to be even more unusual when the initial clues found were detected relatively early in the

Table 12.49: Clue placement and detectability for problem instance B.

Region	Subject exit vertex	Number of clues	Number detectable
116	17	15	2
167	35	18	4
168	17	5	0
130	24	3	0
137	23	4	0
132	10	5	0

Table 12.50: Clue placement and detectability for problem instance C.

Region	Subject exit vertex	Number of clues	Number detectable
55	6	0	0
148	19	18	2
169	6	20	4
113	4	5	0
147	6	3	0
47	8	4	1

search and a large time difference is observed between the detection of the first clue and the time at which the method first allocated the *lost_region* to a resource for searching. These observations are particularly pertinent for problem instance B. Under this problem instance, the primary search area method with PSR criterion and deployment from the exterior base is an extreme example of such behaviour, with 142 hours elapsing between the initial clue detection and *lost_region* allocation.

We now analyse the relationship between clue detection and subject detection over each of the three problem instances, beginning with problem instance B.

12.8.1 Problem Instance B

Some of the observations of earlier subject detection occurring via methods detecting few or no clues can be explained by these methods selecting the *lost_region* for searching at an early phase of the operation, prior to any clue detection having occurred, *i.e.*, the *lost_region* was searched prior to those regions which contained the clues that were able to be detected, given the POD level of the region search and the pseudo random number streams dictating each clue's detection likelihood. In the majority of cases, however, an analysis of the clue detections and subsequent update of the POA value of the *lost_region*

Table 12.51: Clue detection under the benchmark method from the exterior base, under night searching and least hours resource criterion.

Time clue detected	Region clue located in	Vertex subject exited by	Previous POA of *lost_region*	Updated POA of *lost_region*
13.483	167	35	0.022	0.015
13.757	167	35	0.015	0.011
14.748	167	35	0.011	0.009
15.442	167	35	0.009	0.007
40.520	130	24	0.019	0.013

reveals that the reason for these search outcomes is in fact the nature of the regions in which the clues are detected. As outlined in Section 5.3.5, the set of regions that have their POA value increased upon clue detection consist of the region in which the clue was detected, and the regions adjacent to it in the direction of subject movement. Subject movement is defined by the vertex that the subject exited the region by, and hence, the adjacent regions are defined as those that have this exit vertex in common.

When we examine the five clues detected by the benchmark method from the exterior base, searching through the night and utilizing the least search hours resource criterion, it can be seen from Table 12.51 that, after each clue detection, the *lost_region* is less favoured. This decrease in the POA value of the *lost_region* is due to that region not being adjacent to the region in which the clue was detected, in the direction that the subject was moving in when the clue was placed. Hence, whilst regions adjacent to these exit vertices have their POA value increased, the net effect of then normalizing the POA regions is that the POA value of the *lost_region* is decreased. Figure 12.12 illustrates the relationship between the *lost_region*, and the regions 167 and 130 in which clues were detected in this instance.

As the *lost_region* is defined by vertices 10, 14 and 23, it will only have its POA value increased as a result of clue detection[1] if the clue detected indicates that the subject exited an adjacent region via vertex 10, 14 or 23, or a clue is detected within the *lost_region* itself. Upon inspection of the vertex path taken by the subject: 27 - 17 - 35 - 17 - 35 - 24 - 23 - 10, it can be seen that this first condition will only arise if a clue is detected in the region just preceding the *lost_region* in the path, triangular region 137 (24, 15, 23). Four organic clues are located within region 137, but even though this region is searched, these clues are not detected throughout the simulated search operation for this particular search method and set of input parameters. When the pseudo random number stream

[1]The POA value of the *lost_region* will also be increased due to a search without detection of another region of the search region graph.

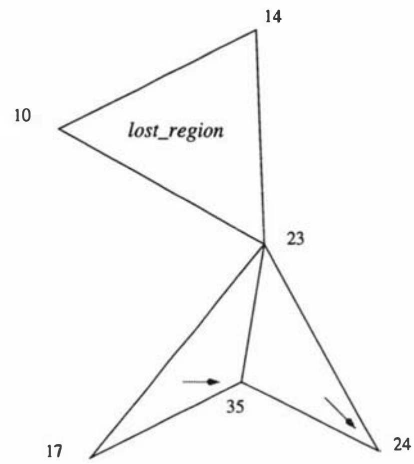

Figure 12.12: Adjacency of the *lost_region* to regions in which clues are detected.

that governs clue detection is altered to ensure the detection of two of the four clues in region 137, the search results for three levels of the ROC_0 parameter are detailed in Table 12.53.

The time at which the *lost_region* is first allocated for searching occurs earlier in each of these instances than when no clues are detected within region 137. In particular, the *lost_region* is first allocated for searching after the clues within region 137 have been detected and prior to any further clues being detected. A ROC_0 value of 0.6 results in a vast improvement in detection time and exhibits one of the highest POA values for the *lost_region* upon detection. The failure to detect the subject for a ROC_0 value of 0.2 and the greater detection time resulting for a ROC_0 value of 0.4, are due to the pseudo random number stream governing subject detection rather than a failure to allocate the *lost_region* for searching.

The same three levels of the ROC_0 parameter are also trailed for this resource allocation method when no clues are detected within region 137; these results are displayed in Table 12.52. As all the clues detected within these simulated searches indicated that the subject was moving in a direction away from the *lost_region*, the net effect of these discoveries was a decrease in POA value for the *lost_region*. When the ROC_0 value is set at a lower level, *e.g.*, 0.2, this decrease in POA value is to a smaller extent and, hence, it would be expected that an earlier subject detection time would result than when ROC_0 is set at a higher level. From Table 12.52 it can be seen that this is true when comparing the detection times of an ROC_0 value of 0.2 to 0.6 (where no detection occurred for a ROC_0 value of 0.6) but that the pattern is not followed when the ROC_0 parameter is set equal to 0.4. In this instance the *lost_region* is allocated at the earliest time, with

Table 12.52: Benchmark allocation method with least hours resource criterion, for problem instance B from the exterior base located at vertex 27, under night searching when no clues are detected in region 137.

ROC_0 value	Time of visual detection	POA_{lost_region} at completion	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
0.2	69.572	0.026	5	13.483	66.762	53.279	0.00	44	W
0.4	58.874	0.023	5	13.483	57.776	44.293	0.00	36	W
0.6	-	0.025	7	13.483	-	-	81.77	96	ND

Table 12.53: Benchmark allocation method with least hours resource criterion, for problem instance B from the exterior base located at vertex 27, under night searching when clue detection in region 137 is assured.

ROC_0 value	Time of visual detection	POA_{lost_region} at completion	Number clues detected	Time 1st clue detected	Time 1st allocate *lost_region*	Time between 1st clue detect & region allocate	Down time	Number resources	Retrieval means
0.2 *(2)	-	0.031	9	7.749	12.000	4.251	81.89	95	ND
0.4 *(3)	99.545	0.021	7	7.749	12.000	4.251	0.00	66	H
0.6	9.909	0.038	2	7.749	9.361	1.612	0.00	4	W

subject detection following soon after. The reverse pattern is expected when clues are detected within region 137, such that a higher ROC_0 value would result in a greater POA value of the *lost_region* when these clues are detected, resulting in faster detection of the subject, and indeed, this is what is observed in Table 12.53.

Similar clue and subject detection patterns can be seen in problem instance A, where if clues are detected within a region that indicates that the subject moved towards the *lost_region*, the subject is generally detected more quickly than if the clues detected indicate that the subject moved in another direction.

12.8.2 Problem Instance A

In problem instance A, the number of clues falling within each region along the subject's path is greater than in the other two problem instances, as only three regions are moved through yet the same total number of clues are placed. A non-organic clue is also located within each of the regions, except the *lost_region* itself. Hence, the likelihood of a clue detection occurring within the region directly preceding the *lost_region* and increasing the POA value of the *lost_region* is greater than for other problem instances.

When considering the search results of problem instance A, it can be seen that more clues were detected when deploying resources from the exterior search base located at vertex 27, than from the interior search base located at vertex 0. Upon inspection of the regions comprising the subject path, this can be explained by the first edge in the path, edge (27,2), being incident to the exterior search base and being utilized as a means of redeploying resources back to the search base. The region is often selected as an unsearched edge under the redeployment path algorithm of Algorithm 11.8, despite its POA value of 0.0. This region contains the greatest number of clues in the subject's path and is also searched to a POD level of 1.0, as it is an edge. Thus, more clues are detected when this region is searched than when the other two regions along the subject's path are searched.

The clues detected within region 0 (edge (27,2)) indicate subject movement away from the *lost_region* and, hence, the POA value of the *lost_region* is decreased upon their discovery. Region 141 however, directly precedes the *lost_region*. When clues are detected solely within region 141, the time between the first clue detection and the time that the *lost_region* is allocated for searching is less than when clues are first detected within region 0. This can be observed when examining the single task allocation method from the interior base: this method first detected clues in region 141 when allocations were made via the PSR and search priority criteria, and first detected clues in region 0 when allocating via the POS criterion. Subject detection followed clue detection more rapidly under the first instance than the second.

The single task method is designed to respond more quickly to changing information as reflected by the POA values. Hence, it would be expected that this allocation method would respond to clue detection quickly, with this being reflected in the subsequent search path assignments. When the single task and path scan methods are compared across the region criteria, it can be seen that the time between clue detection in region 141 and the allocation of the *lost_region* for searching, is less for the single task method for each criteria from the exterior base, and for the search priority and POS criteria from the interior base. However, when the PSR criterion is employed from the interior base, the path scan method allocates the *lost_region* more quickly. In this instance the *lost_region* is allocated between clue detections in region 141 to a search resource which becomes available for further search allocations. In this respect the method can be viewed, on this occasion, as being similar to the single task method, since new search allocations are made soon after clue detection.

12.8.3 Problem Instance C

In the random subject path of problem instance C, the subject repeatedly moves through vertices 4 and 6. As vertex 6 is a vertex of the *lost_region*, this means that its POA value will be increased when clues are discovered in regions 55, 169 and 147, where the subject exits through vertex 6, or in the *lost_region* itself, *i.e.,* in 67% of the regions on the subject's path. The primary search area method with smallest search hour criterion, from the interior base under weather scenario one, illustrates this behaviour with two clues being detected within region 169 prior to the *lost_region* being allocated, with subject detection following soon afterwards.

An interesting feature of this subject path is that every region on the path has vertex 6 as a vertex and, as such, has its POA value increased, along with the *lost_region*, when a clue is discovered indicating subject movement through vertex 6. The consequence of this is that regions comprising the subject's path are more likely to be allocated for searching upon clue detection occurring within these regions, leading to further clue detection taking place.

Even though the percentage of the regions in which clues are located within which, if detected, would increase the POA values of the *lost_region*, is equal to that of problem instance A, fewer clues are detected on the whole when simulating a search over problem instance C. This phenomenon can be explained by the high concentration of clues laid along region 0 (ridge edge (2,27)) which have a greater probability of detection given that the edge is searched to a 100% POD level. Hence, when this region is searched it returns a higher number of clues than any other single region over all the subject paths considered. The percentage of total clues which are detectable is also less in problem instance C compared to problem instance A.

The schedule of clues detected by those allocation methods to detect clues in regions 148 or 169 are depicted in Table 12.54, together with the resulting POA value change for the *lost_region*. From Table 12.36 and Table 12.34 it can be seen that the *lost_region* was allocated for searching after the first clue was detected in each of these simulated searches and that it was allocated for searching more quickly when clues were first detected in region 148 rather than region 169. In these instances the knowledge obtained from the clue detection did not increase the POA value of the *lost_region*. While this result is unexpected it is explained by analysis of the search allocations which followed clue detection. When clues were detected within region 148 the POA value of the *lost_region* remained at 0.000 but the *lost_region* was then allocated to a search resource as an edge of a redeployment path back to the search base. The *lost_region* was never a search region within the primary search area. When clues were first detected in region 169 the *lost_region* falls within the primary search area and is allocated for searching on that basis. Such results highlight the fact that, within this problem instance, the *lost_region*, which has an initial POA value of 0.000 and often a final POA value of 0.000, rarely falls within the primary search area, when this is defined. The region is only included in this area when its POA value is greater than 0.000 or the region is completely enclosed by regions which comprise the primary search area. Hence, when employing a resource allocation method that allocates search resources to only those regions within the primary search area, the *lost_region* is searched only when it features as an edge in a redeployment path.

The *lost_region* is selected both in shortest redeployment paths and in composite redeployment paths as an unsearched edge. This exemplifies the contribution that such composite paths have in diversifying the search strategy. The *lost_region* is a component in the shortest edge path to the base for 10% of origin vertices; vertices 4, 6, 22 and 36.

12.9 Analysis of Resource Allocation Methods

We now analyse the performance of each resource allocation method over the three problem instances.

12.9.1 Benchmark Method

The construction of search paths via the benchmark method, varying the resource selection criteria, reveals no particular dominance of one criterion over the others. The ranking

Table 12.54: Clue detection and subsequent POA update for problem instance C when clues were detected outside of the *lost_region* by the primary search area method.

Region criterion	Time of clue detection	Region clue detected in	Subject exit vertex	Previous POA *lost_region*	Updated POA *lost_region*
Interior base					
Smallest search hrs	40.847	169	6	0.000	0.020
	41.132	169	6	0.020	0.023
Smallest search & access hrs	41.418	148	19	0.000	0.000
Exterior base					
Search priority	64.812	148	19	0.000	0.000
	64.871	169	6	0.000	0.010

Table 12.55: Ranking of resource criteria for the benchmark method by problem instance.

Criterion	Rank			Overall Rank
	Problem Instance A	Problem Instance B	Problem Instance C	
Index	1.5	2.5	1.0	1.7
Least remaining search hours	2.0	1.5	2.5	2.0
Greatest remaining search hours	2.5	2.0	1.8	2.1

of the resource selection criteria for each problem instance is detailed in Table 12.55. As discussed in Section 12.5.1, the general allocation of search paths to individual search resources throughout the simulation, rather than a combined allocation to all resources, means that the use of this criterion is mainly limited to the commencement of the operation and, as such, meaningful analysis of this method parameter is not possible.

12.9.2 Single Task Method

The three region selection criteria used to generate search allocations via the single task method are ranked for each problem instance in order of increasing subject detection time. These are then averaged over all the problem instances to achieve the results detailed in Table 12.56. No one criterion performs better than all others for each problem instance and their average performance rank is very similar.

In order to distinguish how much variation there is in the subject detection times obtained by each criterion, the ratio is calculated to the best solution found by the single task method for each criterion over problem instance and base location. These ratios

Table 12.56: Ranking of region criteria for the single task method by problem instance.

Criterion	Rank			Overall Rank
	Problem Instance A	Problem Instance B	Problem Instance C	
PSR	2.5	2.0	1.0	1.8
Search priority	2.5	1.0	2.3	1.9
POS	1.0	3.0	2.8	2.3

Table 12.57: Best solution ratios for each region selection criterion for the single task method by problem instance.

Criterion	Problem Instance A		Problem Instance B		Problem Instance C	
	Exterior Base	Interior Base	Exterior Base	Interior Base	Exterior Base	Interior Base
PSR	1.240 *	1.180	1.170	1.455	1.000	1.000
Search priority	1.238	1.187	1.000	1.000	7.726 *	1.254
POS	1.000	1.000	2.708	2.816 *	3.644	5.484

are presented in Table 12.57, with the worst case performance for each problem instance identified by an '*'. These results show that for problem instance A there is little variation in the results produced by the individual region criteria while significant variation in detection times is observed for problem instance C. In this problem instance there is a factor of approximately eight between the best and worst solution. In conclusion, the average ranks of each criterion actually obscure the robustness of the PSR criterion, which performs the best over all problem instances.

12.9.3 Primary Search Area Method

Each region selection criterion under the primary search area method is ranked in order of resulting subject detection time for each problem instance and then assigned an average rank for the three problem instances. This data is recorded in Table 12.58. From this table it can be seen that the smallest search and access hour criterion performed consistently well over all problem instances and achieved the top average rank overall together with the other criterion that considered both the search hour duration and the time required to access a search task, *i.e.*, search priority. The consistently poor performance of the highest elevation criterion may be due to the small range in elevations generated for the TIN used in these experiments.

The ratio of the solution values obtained by each of the eight region criteria, to the

Table 12.58: Ranking of region criteria for the primary search area method by problem instance.

Criterion	Rank			Overall Rank
	Problem Instance A	Problem Instance B	Problem Instance C	
Smallest search & access hours	2.5	3.5	3.3	3.1
Search priority	3.0	1.5	4.8	3.1
POS	5.5	1.5	4.0	3.7
Smallest search hours	3.5	5.5	2.8	3.9
Smallest access hours	4.0	6.5	4.0	4.8
PSR	6.0	4.0	5.3	5.1
Highest degree exit vertex	3.5	6.5	5.3	5.1
Highest elevation	8.0	7.0	6.3	7.1

best solution values obtained for the method, are calculated and presented in Table 12.59 for each problem instance. In the cases where the criterion used failed to detect the subject, this is indicated by a '-'. Such non-detections occurred for each problem instance and as the result of the use of three different criteria. The variability of the detection times over the three problem instances, when these non-detections are excluded from consideration, are within a factor of four. No one problem instance reveals significantly greater variability than any other, and the highest elevation criterion is the only criterion to consistently display significantly worse behaviour over all problem instances and base location.

12.9.4 Path Scan Method

The same three region criteria used to generate resource allocations for the single task method are also used to generate resource allocations for the path scan method. When these are ranked in increasing order of subject detection times for each problem instance and then averaged, the results are depicted in Table 12.60. The results are very similar to those obtained for the single task method and the same ordering of criteria occurs. The PSR criterion in this case was the best performer for all problem instances, performing equally to the search priority criterion for problem instance A.

As for the single task method, we also calculate the ratio with respect to the best solution value for each region selection criterion in order to determine the variability of subject detection times between the criteria. Table 12.61 displays these ratios. As for the single task method, the PSR criterion is the most robust and dominates both other

Table 12.59: Best solution ratios for each region selection criterion for the primary search area method by problem instance.

Criterion	Problem Instance A		Problem Instance B		Problem Instance C	
	Exterior Base	Interior Base	Exterior Base	Interior Base	Exterior Base	Interior Base
Smlst search & access hours	1.000	2.371	1.507	1.839	1.182	1.716
Search priority	1.011	2.362	1.031	1.000	1.817	2.195
POS	1.017	3.820	1.000	1.230	1.174	2.163
Smallest search hours	1.018	1.990	1.717	2.134	1.000	1.708
Smallest access hours	1.010	3.407	- *	1.870	1.030	2.167
PSR	2.051	2.382	2.039	1.799	1.037	2.208
Highest degree exit vertex	1.032	1.000	- *	1.860	1.654	1.000
Highest elevation	3.437	- *	2.089	2.184	- *	2.893

Table 12.60: Ranking of region criteria for the path scan method by problem instance.

Criterion	Rank			Overall Rank
	Problem Instance A	Problem Instance B	Problem Instance C	
PSR	1.5	1.0	1.0	1.2
Search priority	1.5	2.0	2.8	2.1
POS	3.0	3.0	2.3	2.8

criteria across the three problem instances. If the anomalous result for resource deployment from the exterior base under problem instance B is excluded from consideration, the worst case performance ratio again occurs under problem instance C, the problem initially predicted to be the most difficult to solve. Over all three problem instances and region criteria there is less variation in subject detection times than for the single task method.

12.9.5 Hybrid Path Scan Method

When the secondary region selection criteria are ranked with respect to the initial selection criteria, the results are displayed in Table 12.62. These results show different patterns depending on the initial selection criteria. For both the PSR and the POS initial selection criteria the remaining priority based criteria perform well in generating the second portion of the path while they do not perform so well, relative to the other criteria, when the initial portion of the path is generated via the search priority criterion. The highest degree exit vertex criterion performs the least well of all secondary criteria,

Table 12.61: Best solution ratios for each region selection criterion for the path scan method by problem instance.

Criterion	Problem Instance A		Problem Instance B		Problem Instance C	
	Exterior Base	Interior Base	Exterior Base	Interior Base	Exterior Base	Interior Base
PSR	1.862	1.000	1.000	1.000	1.000	1.000
Search priority	1.000	1.012	1.028	1.156	2.096	2.936 *
POS	1.867 *	1.721	8.476 *	2.579	1.502	1.532

irrespective of the initial path criterion.

When the best solution ratios are calculated for each of the secondary selection criterion the results are depicted in Table 12.63. The PSR initial criterion resulted in the least variation over all secondary criteria and problem instance, and the use of PSR as the secondary criterion in the hybrid path generation also showed relatively small variation across the problem instances and weather scenarios. The greatest variations in solutions obtained using the secondary criterion occurred under adverse search conditions, as would be expected.

12.9.6 CPU times

While computing code was not written with the intent of achieving great efficiencies, the CPU times recorded for each resource allocation method provide a measure of the computational effort required by each method to generate the search paths for the initial allocation of resources at the commencement of the search operation. The average CPU time required by each method over each of the three problem instances is recorded in Table 12.64. It can be seen from this data that the method requiring the least computational effort is the benchmark method. This result is somewhat misleading as it occurs because only edge regions are considered as candidates for searching at operation commencement. The computational effort to determine the best search orientation for an edge is significantly less than that required to determine the best search orientation for a triangular region. In subsequent periods the computational effort required by the method is greater.

The path scan method is the method requiring the greatest computational effort. This is expected as the method allocates a resource's entire search allocation for the period and the search for the components of this allocation is not restricted to a specified primary search area. The single task method, on the other hand, only allocates a single task to each resource and subsequently requires a relatively small amount of computational

Table 12.62: Ranking of secondary region criteria for the hybrid path scan method by initial criterion and problem instance.

Criterion	Rank			Overall Rank
	Problem Instance A	Problem Instance B	Problem Instance C	
PSR/				
Search priority	1.0	3.0	1.0	1.7
Smallest access hrs	2.5	1.5	1.0	1.7
POS	3.5	4.0	1.0	2.8
Smallest search and access hrs	3.5	5.0	1.0	3.2
Smallest search hrs	6.5	3.0	1.0	3.5
Highest elevation	4.5	5.0	1.0	3.5
Highest degree exit vertex	6.5	4.5	1.0	4.0
Search priority/				
Highest elevation	1.5	1.0	6.0	2.8
Smallest search hrs	5.0	2.5	2.0	3.2
Smallest access hrs	3.0	4.0	4.0	3.7
POS	1.5	6.5	3.5	3.8
PSR	4.5	4.5	4.0	4.3
Smallest search and access hrs	7.0	5.0	1.0	4.3
Highest degree exit vertex	5.5	4.0	6.0	5.2
POS/				
PSR	4.0	2.0	2.0	2.7
Smallest search hrs	4.0	4.5	1.0	3.2
Search priority	3.5	3.5	4.0	3.7
Smallest search and access hrs	5.0	5.0	3.0	4.3
Smallest access hrs	4.0	5.0	5.0	4.7
Highest elevation	1.0	6.0	7.0	4.7
Highest degree exit vertex	6.5	2.0	6.0	4.8

Table 12.63: Best solution ratios of secondary region criteria for the hybrid path scan method by initial criterion and problem instance.

Criterion	Problem Instance A		Problem Instance B		Problem Instance C	
	Exterior Base	Interior Base	Exterior Base	Interior Base	Exterior Base,W1	Exterior Base,W2
PSR/						
search priority	1.000	1.000	1.316	1.000	1.000	1.000
smlst access hrs	1.006	1.008	1.000	1.018	1.000	1.000
POS	1.008	1.014	1.088	1.123	1.000	1.000
smlst search and access hrs	1.443	1.008	1.257	1.658	1.000	1.000
smlst search hrs	1.542	1.187	1.058	1.107	1.000	1.000
hgst elevation	1.009	1.019	2.147	1.035	1.000	1.000
hgst degree exit vertex	1.446	1.708	1.711	1.035	1.000	1.000
Search priority/						
hgst elevation	1.000	1.001	1.000	1.000	3.275	5.705
smlst search hrs	3.508	1.177	1.011	1.009	1.105	1.003
smlst access hrs	2.049	1.003	1.004	1.196	1.569	5.672
POS	2.035	1.000	1.168	1.196	1.475	5.672
PSR	2.458	1.186	1.014	1.059	2.317	1.440
smlst search and access hrs	3.518	1.192	1.014	1.161	1.000	1.000
hgst degree exit vertex	3.506	1.187	1.014	1.161	4.465	5.703
POS/						
PSR	1.646	3.615	1.613	1.000	1.043	1.043
smlst search hrs	1.795	2.506	3.249	1.226	1.000	1.000
search priority	1.165	3.615	1.605	1.416	1.109	1.120
smlst search and access hrs	1.169	3.996	3.248	1.267	1.085	1.085
smlst access hrs	1.646	2.532	1.995	1.425	2.285	6.574
hgst elevation	1.000	1.000	3.243	1.823	3.309	13.832
hgst degree exit vertex	1.863	3.623	1.000	1.267	3.130	8.102

Table 12.64: Average CPU time (in seconds) for each resource allocation method by problem instance.

Method	Problem Instance A		Problem Instance B		Problem Instance C	
	Exterior Base	Interior Base	Exterior Base	Interior Base	Exterior Base	Interior Base
Benchmark	0.63	0.63	0.25	0.30	0.63	0.64
Single task	0.97	1.02	0.99	0.99	0.99	1.17
Primary search area	3.54	2.44	0.66	0.83	3.72	2.41
Path scan	6.99	5.83	5.72	6.01	6.91	5.75

effort. Both the benchmark and primary search area methods require relatively less computation time for problem instance B than for the other two problem instances due to the lesser number of search regions which comprise the primary search area.

12.10 Comparison of Problem Instances

Problem instance A, where the subject was located in a region of high density that required greater search effort to search to a given POD level than the *lost_region* in the other two problem instances, resulted in the slowest best subject detection time of the three problems. This problem instance was in fact the most difficult to solve of the three, under weather scenario one, a fact contributed to by the search effort required to search the *lost_region*. This was particularly evident when the level of required search effort discounted the region from selection during times of reduced visibility. This identified a deficiency in the current model to adequately address the search of such a region to a minimum POD level within a constrained time frame. This could be overcome by permitting further segmentation of the region or modelling multiple resource deployment to the region.

Problem instance B proved to be the middle case of the three problem instances under weather scenario one, resulting in slower overall detection times than problem instance C, but faster detection times than problem instance A. The worst best solution time of any method was, however, worse than that of problem instance A. In general, a greater number of clues was detected for problem instance B than for any other problem instance. This was due to both the confused movement of the subject that resulted in an additional number of clues being laid in the region that was re-visited, and the route intentions of the subject being known at the commencement of the search operation.

Problem instance C, where the subject was located on an edge with a POA value of

0.0, resulted in the quickest best subject detection times. When the search was simulated under adverse weather conditions the worst performance of the allocation methods was greater, as was to be expected. This problem instance illustrated the drawback of selecting regions solely from a primary search area when the subject was located within a region that was not identified as being a likely region for him to be positioned within. The problem also illustrated the benefits associated with the use of search diversification strategies to solve problems where the POA of the *lost_region* is low, and where subject intentions are non-existent and few clues are detected. These strategies, which included the use of redeployment paths over unsearched edges, were particularly effective in solving this problem instance where the subject was located on an edge region.

12.11 Conclusion

Simulation is a descriptive form of experimentation that permits a number of factors to be varied in order to gain understanding of the behaviour of the resource allocation methods under differing search scenarios. While we have only conducted preliminary analysis over three problem instances, some trends have been identified. These trends may apply to most search scenarios in general, while others may be restricted to those scenarios that exhibit similar characteristics to those tested. In most instances further experimental work would be required to establish this.

General trends in search strategy that have been identified by this experimental work include the dominance of a night searching strategy and the use of the sound sweep method in the initial search periods, when the subject is responsive. The detection by sound was shown to be particularly beneficial when adverse weather conditions were predicted, with subject detection occurring five days earlier than when visual searching alone was used, in one instance. The benefit of the strategy of redeploying search resources over unsearched edges was also displayed, with such a strategy diversifying the search, and subsequently resulting in clue or subject detection.

The allocation of a search region adjacent to a search resource's position that returned the highest POS of the neighbouring regions when no regions remaining in the primary search area were candidates for allocation to that resource, (via Algorithm 8.5), also proved to be beneficial and was another form of search diversification. This strategy was most successful when the *lost_region* was not included in the primary search area. Further experimental work is required to establish whether the strategy of permitting a search region to be searched more than once in a search period is beneficial. This strategy is dependent on both the target POD level, the amount of search effort required to search a particular region, and the type of search scenario faced.

A preference for not restricting the set of regions for search allocation to a primary search area is shown in both the experiments conducted for the path scan method in problem instance B, and in the dominance of path methods that do not restrict in this way, for problem instances B and C. In particular, all instances where the subject remained undetected and the search was suspended resulted from the application of either the benchmark or primary search area method, both of which select regions for searching from a primary search area. The primary search area method performs the least well of all methods tested, with the worst best subject detection time for each problem instance. Of the remaining resource allocation methods tested - the single task method, and the path scan method together with hybrid variants - no one method was shown to be superior for all search scenarios. As was to be expected the single task method was the quickest of the methods to respond to changing information, while the path scan method required the greatest computational effort of all the methods. The PSR region selection criterion was the most robust over the three problem instances, for the path scan and single task methods.

In general, more clues were detected when the subject's intended route was known, as in problem instance B. There was also no evidence from the experiments conducted that the benchmark method detected a greater number of clues in its reconnaissance phase (first search period) than other methods. It was also shown from experiments conducted for problem instance B that it was preferable not to incorporate intended route knowledge when the subject moves away from this path and follows the historical POA values of the search area. However, in reality it is not possible to know this without the benefit of hindsight. Search planning that takes into account multiple scenarios of subject movement would address this problem.

Chapter 13

Conclusions And Avenues For Further Research

We conclude this study by revisiting the initial objectives of the thesis detailed in Chapter 1, examining the degree to which these have been met, and exploring avenues for further research.

13.1 The Physical Terrain Model

After considering the DTMs currently used to model terrain we selected the TIN model to model a search terrain, constructing it via a constrained Delaunay triangulation. This model was selected for its ability to: represent the physical search area by continuously connected triangular faces that model terrain variability and provide explicit proximity information; provide good visualization of the search terrain; and include existing natural features such as regions of vegetation change, streams, tracks and ridge lines. Modelling such features as edges in the TIN, via the constrained Delaunay triangulation, lends itself to the problem of defining the boundaries of individual search regions in a practical way that ensures the similarity of terrain falling within each region.

A technique for allocating realistic searching and non-searching speeds over the TIN model was then developed. This technique calculated traversal speeds between any two points on the TIN depending on the type of vegetation and gradient modelled, thus distinguishing between 'up-hill' and 'down-hill' traversal. Correction factors were then included to adjust these speeds for different weather and light conditions. Such a costing model could be manually adjusted to more accurately represent the actual speeds possible over terrain which falls under the jurisdiction of a given SAR team, including any special 'short-cuts' that may exist.

The model allows for fast calculation of the time needed to move between locations so that this can be accurately incorporated into search planning. By incorporating these into a computer model and ensuring that features which provide means of fast access are modelled as edges of the TIN, shortest time paths between any two points on the TIN can be calculated over the edges of the TIN by a shortest path algorithm such as that developed by Dijkstra [160]. This provides an advanced definition of access paths.

Heuristic methods were developed to address the problem of search region definition by utilizing the edges of the TIN as the boundaries of the search regions; hence boundaries represent natural terrain phenomena in terms of the constrained triangulation approach that is used. The objective of the methods is to form single search regions in a way that avoids natural barriers occurring within a region and maintains similar vegetation throughout the region, hence reducing search difficulty. The heuristic based on Colwell's Trail-based POA method [27] was developed to amalgamate regions of the TIN such that an edge representing a track was used as the formation edge of each search region, where possible, in order to provide quick access to search regions.

The methods preprocess a given terrain prior to any search operation, which has both advantages and disadvantages. The advantages of preprocessing are that it: reduces planning time when an operation is commenced; enables accurate search duration times to be allocated to each region; enables familiarity of search regions for resources who are regularly called upon to search over the terrain; and enables the matching of historical lost person behaviour statistics to each region (this requires search regions to incorporate areas of similar POA). The disadvantages of the preprocessing approach are: unless regions comprise areas of similar POA values, ineffective search allocations will result; and as the area which can be effectively searched within a given time constraint will alter, both with search conditions and the POD level of the search required by search management, the size of search regions alters, a factor not accounted for in the static preprocessing approach. The development of real-time search region definition, which addresses these disadvantages, is an area for further research, as is the programming of region definition methods.

To our knowledge no other physical model for land SAR exists which incorporates all of these aspects. We would recommend modelling the search terrain in this way prior to a search operation, defining search regions, and assigning each a POA value based on historical POA information. A statistical database, such as the SARDAT database developed by Downs and Dolan [43], could be linked to the model to enable historical subject movement data, categorized by subject category, to be used to determine historical POA values for each defined search region. This would need to be updated with each search, and would provide an invaluable management tool over time. When a search operation is initiated, these POA values can be adjusted in light of information known about the subject and his movements. Prior definition of search regions linked to historical POA data would speed up search planning.

The base model could be further enhanced by the inclusion of hazards that exist in the search area and the identification of any regions where radio communication could not be received. A graphical interface could also be developed to provide real-time visual information to search management and, if incorporated with the use of GPS units, would enable the progress of search resources to be visually traced.

13.2 Visibility, Detection and Clue Modelling

A visibility model was developed over the TIN, defined in terms of Perkins' 'visibility measure' [137]. This model was defined in relation to the density of vegetation covering a region and correction factors were proposed to adjust the distance a search resource could see under varying environmental conditions, including searching at night. This model was extended to also model the distance a search resource could hear under differing environmental conditions and was defined in consideration to Colwell's experimental work in trialling the sound sweep search method [20, 25]. While this visibility model did not model true 3D visibility over the TIN, this was justified by the close search work usually required for searching for clues and the search subject, where the resource's concentration is focussed on their immediate vicinity. An avenue for further research would be to develop a 3D visibility model over the TIN such as that developed by Goodchild and Lee [76]. Such a model would be particularly useful when observation points were established to cover given regions of the search area.

Research into detection models over land, and particularly bush-covered terrain, is greatly lacking and is an area which needs to be urgently addressed. Field experiments need to be conducted under all types of search terrain and conditions which may be experienced so that accurate POD data is available on which to effectively plan searches. Given the current lack of such field data, especially for New Zealand conditions, we adapted the detection model proposed by Perkins [137] for his critical separation model to generate POD data for a human subject. We then extended this detection model to model the detection of a subject by sound. While these models are only approximate, they were shown to be reasonable with respect to the field data that is currently available. The responsiveness of a subject was also modelled over time to realistically account for the subject's deteriorating condition as the period of time spent in the open lengthened. Further levels of responsiveness, such as a normal or quiet voice response, could be incorporated.

The study then developed two methods by which to generate subject motion, including the modelling of confused behaviour such as that which may result from disorientation. The placement of clues was simulated along the subject's path and a model of clue detection was developed based upon the visible detection model for a subject. The model also took into account the deterioration of organic clues that results over time from exposure to the environment. To our knowledge, this is the first such attempt to model clues and their detection.

An algorithmic approach to the update of the subject's location probability distribution upon the discovery of a clue was also developed. Unlike currently used methods of POA update, this approach developed an update method which did not rely on human input as to the relevancy of the detected clue. The method took into account the cumulatory nature of a number of clues all detected within one search region which provided similar information, rather than considering the information provided by each clue as being equally important. An area for further research would be to model the presence of confounding clues, not laid by the subject, and the effect these have on information gathered and its utilization. Further testing on the appropriate level at which to set the simulation parameter ROC_0 would also be valuable.

13.3 Resource Allocation Methods and Search Strategies

The problem of resource allocation in a SAR operation was shown to be both similar to, and unique from, other problems in the literature. In particular, the problem is characterized by its dynamism, changing windy traversal costs, spatial and linear traversal of regions, multiple resource paths, and multiple conflicting objectives. The problem was shown to be NP-complete, requiring heuristic solutions to the problem to be developed.

The development of heuristic methods of resource allocation comprised two parts; those designed to specifically address the reconnaissance phase of the search operation and those designed to address the general phase of the search operation. We restricted the search of regions in the reconnaissance phase to the edges of the TIN as the regions traditionally targeted in this phase, such as tracks, ridge lines and streams, are all represented by edges of the TIN, and huts are placed at vertices of the TIN. No region restrictions existed when allocating search tasks to resources in the general search phase.

The allocation of search tasks to resources in the reconnaissance phase has the traditional objective of collecting information as quickly as possible in order to narrow down the search area. We developed resource allocation methods which formalized the current search techniques used to meet this objective, namely: the search of the subject's intended path; the search of any huts and hazard features; a search in the vicinity of the

PLS; a perimeter search; and a form of binary search which cuts the subject's intended path. The shortest time access paths defined over the edges of the TIN were utilized in these methods to ensure fast access to search regions in order to meet the 'hastiness' objective of this search phase.

Further heuristic methods of resource allocation were developed from edge routing literature that considered different objectives from those currently used in land SAR. In particular, we proposed approaches to allocate a search resource to follow a median search path and a hedging search path, where a median searching path allocated regions for searching which were 'central' to a primary search area, and a hedging search path allocated regions for searching which were 'most different' in nature from those already allocated. A hedging search path effectively hedges the current search scenario and strategy being followed by search management against incorrect subject location assumptions, and additionally provides a minimal response time to regions that are not being searched by other resources.

Additional heuristics were proposed to allocate multiple search paths to resources that adapted solution approaches to routing problems in the literature and that were based on the allocation of a required set of edges. A special case of the problem was also examined where a visibility cover of the search area could be achieved from a vertex cover. A minimum weighted matching approach was developed to determine such a cover for k search resources.

The resource allocation methods developed assumed that all resources were initially located at a central search base and that access to search regions was via foot. Allocations were also static in nature. Further research could include the extension to a dynamic response during this search phase and the extension to bilevel routing, where resources are initially transferred to the beginning of their search assignment via helicopter or 4-wheel-drive vehicle.

The problem of resource allocation in the general search phase was addressed by first considering some possible ways of approximating the search of a triangular search region. The width strip traversal method was selected for the purpose of our model for its ease of navigation and its ability to model parallel sweep motion. Limiting searching to individual triangular regions, rather than defining search regions of other shapes, has several disadvantages. One disadvantage is the unequal workload required by each member of the search team when the traversal method results in the length of ground covered by a resource varying with their position in the team. This occurs with the width strip traversal method, and is the most pronounced when the triangular region is of a size that can be searched in one strip. When the width of the search corridor of a search team is greater than the width of the edge that the sweep begins from, closer spacing of resources is required in order to restrict coverage to just the allocated search region; hence searching the region at a higher POD level than the target POD level. In real applications it is unlikely that, if search teams comprise only three or four resources, a triangular region would be of such a small area that this would occur. Nevertheless, regions which are more rectangular in shape would be more desirable, and further research into ways in which these can be obtained and ways in which natural movement, such as those discussed in Section 8.2.5, can be modelled, would be advantageous.

The problem of allocating search resources to search both edge and triangular regions was then addressed in Chapter 8. Heuristic methods were developed which were short-term to medium-term in focus and designed to generate search paths over a planning period of fixed time horizon. The allocation heuristics were designed to be general enough to cater for any shape of search region and any means of search traversal — all that is required is that the time needed to search a region in a particular direction and to a given POD level be calculable.

Two methods for allocating POD levels to regions were considered, in order to determine the search duration time of each task prior to the selection and sequencing of regions for searching. However, a better understanding of detection capabilities is required before extensive research is conducted on these and other methods of allocating POD levels to search tasks. Both single and multiple resource allocation were addressed, with a generic path construction heuristic developed for each. Each heuristic ranks a selected candidate set of regions for searching based on a given criterion. Regions are then chosen for searching based on this ranking, and are inserted into the path of a resource. The heuristics are particularly flexible, being governed by a set of parameters that can be varied to generate quite distinct paths. Four particular types of paths were generated by given sets of parameter values, and were programmed in C; these were then used in the preliminary computational experiments. More extensive experimental work is required over different parameter values and search scenarios to determine the specific type of resource allocation method best suited to specific instances of the SAR problem. An interactive computer package could also be designed which could present several choices of resource assignments to search management for their consideration.

The study then considered the specific attributes of the SAR problem which define it as a dynamic, real-time decision problem with stochastic elements. Some solution approaches from the literature for such problems were examined in concept form and a selection of real-time, dynamic strategies were presented for solving the SAR problem under changing knowledge and changing weather conditions. The majority of these strategies were not explicitly tested within the preliminary experimental work, and provide considerable scope for further development and testing, given the particularly uncertain and changeable nature of a SAR operation.

13.4 Simulation Model

A DES model was developed to simulate the planning and execution of a land SAR operation, thus providing a flexible environment within which to examine the performance of resource allocation methods and search strategies under different search scenarios. This model incorporated the terrain, visibility and detection model developed, and a selection of the heuristics developed earlier in the study as sub-routines. Further rules and heuristics were also developed to address the decisions required by search management at each stage of the search operation. In particular, the DES simulates: the positioning of the search base; the allocation of search resources; communication between search resources and the base; clue detection and information update; subject detection and retrieval; resource movement, including rest breaks; the redeployment and recall of search resources, and the scaling-up and scaling-down of the operation; flooding of regions under adverse conditions, limiting access paths; and the suspension of an unsuccessful operation. Resource redeployment was executed in the field via radio communication in order to improve the efficiency of the search operation. To our knowledge this is the most advanced simulation model developed to date to simulate a land SAR operation.

Precursory computational experiments were conducted over this model using an artificial computer-generated TIN, a restricted set of parameter values, four resource allocation methods, and three search scenarios. The experimental component of this study was preliminary only, due to the emphasis on the modelling contribution. The experiments that were conducted were designed to investigate some of the main behavioral aspects of the resource allocation heuristics under distinct scenarios, rather than exhaustive testing of different simulation and method parameter levels. In particular, we examined the relative performance of the different heuristics under scenarios described by different: weather conditions; subject movement, including knowledge of their intentions; characteristics of the region in which the subject was finally located; levels of subject responsiveness; and levels of clue detection.

While the experiments were too limited to reveal conclusive trends, the initial findings identified tentative trends. These included the dominant strategies of sound sweep searching when the subject was responsive and searching at night, both of which were anticipated. The use of sound detection was shown to be especially beneficial in those cases where adverse weather conditions were predicted as it provided a much faster means of detection, ensuring that the subject was detected prior to the onset of these conditions.

The experiments also showed the advantages of diversifying the search to previously unsearched edge regions when redeploying resources, with such a strategy resulting in both clue and subject detection in some of the problem instances that were examined. Further diversification strategies that allowed for searching outside of a predetermined primary search area were also shown to be advantageous, with the two resource allocation methods that did not confine allocations to regions within a primary search area — the path scan and single task methods — generally performing better than those methods which did. This was particularly evident when the subject was positioned in a region with a relatively low POA value or when he had moved away from his intended path.

Further research is warranted in order to determine at what phase in a search operation, and under what types of search scenarios, diversification and intensification search strategies are best implemented. Research into problem characterization would be particularly beneficial, with the goal of classifying which types of search strategies and allocation methods are best applied to different problem types. Specific avenues for further experimental research that utilize the simulation model include:

- examination of the initialization of the target POD level in each search planning period and particularly, whether a constant POD level should be utilized over each search period with multiple region searching permitted, or whether regions should be searched only once in a period but with the target POD level increasing with subsequent periods;
- the applicability of specific resource allocation methods to solving the SAR problem for different underlying POA distributions;
- the performance of sound versus visual searching during the night;
- experiments conducted over larger networks and models of real terrain; and
- the use of different types of allocation methods at different phases of the search operation, such as when the urgency level of the search increases or when new information is gathered.

The simulation model could also be extended to include such aspects as: the utilization of air resources for searching; regions where communication is not possible or where obstacles to traversal exist; increasing resource levels as an operation develops, as commonly occurs; explicit modelling of resource fatigue; and the use of sub-bases together with permitting the siting of bases at locations other than the vertices of the TIN.

We consider that the greatest areas for further research would be to extend the simulation model and resource allocation methods to the search for a moving subject and to the search for a subject who wishes to avoid detection.

13.5 Conclusion

In conclusion, this study has developed a physical model for a land SAR operation which addresses a number of the limitations of the data on which searches are currently planned, including: resource traversal times that are calculated with respect to vegetation, gradient and search conditions; determination of shortest time access paths; resource visibility restrictions that are determined with respect to vegetation and search conditions; plausible models of subject and clue detection; ease of visualization of the search terrain; and the clear definition of vegetation breaks and the explicit modelling of linear terrain features to facilitate effective and logical definition of search regions.

Search strategies that respond to the stochastic and dynamic elements of the problem have been developed together with search phase dependent, resource allocation methods that can be executed in real-time. These strategies fit well within the description of the Dynamic Era of search theory identified by Stone [154]. A simulation model was then developed which incorporated these components, in order to examine the progress and outcome of a SAR operation under different search scenarios. Precursory experiments were then conducted over a restricted set of resource allocation methods and simulation parameters.

While the overall objective of this study has been to address a number of the limitations of current decision-making involved in planning a SAR operation there remains considerable scope for further research, especially further computational experimentation examining the performance of various resource allocation methods and search strategies over differing search scenarios, SAR problem characterization, the extension of the model to incorporate a moving subject, comprehensive field experiments to determine accurate sensor detection functions for different search conditions, and the incorporation of subject behavioural information into the terrain model.

This concludes our investigation into aspects of the SAR problem. Ultimately it is our hope that the research presented here will seed further research into land SAR procedures which, in time, will result in lives being saved.

Bibliography

[1] Aaronson-Hutson, V., and ReVelle, C. Indirect covering tree problems on spanning tree networks. *European Journal of Operational Research 65* (1993), 20–32.

[2] Allen, R., Ed. *The Concise Oxford Dictionary*, 8th ed. Oxford University Press, Oxford, England, 1990.

[3] Anon. New on the market. *OR/MS Today* (April 1996), 70.

[4] Arkin, E., Halldórsson, M., and Hassin, R. Approximating the tree and tour covers of a graph. *Information Processing Letters 47* (1993), 275–282.

[5] Arkin, E., and Hassin, R. Approximation algorithms for the geometric covering salesman problem. *Discrete Applied Mathematics 55* (1994), 197–218.

[6] Assad, A., Pearn, W., and Golden, B. The capacitated chinese postman problem: lower bounds and solvable cases. *American Journal of Mathematical and Management Sciences 7* (1987), 63–88.

[7] Aurenhammer, F. Voronoi diagrams: A survey of a fundamental geometric data structure. *ACM Computing Surveys 23*, 3 (1991), 345–405.

[8] Barr, R., Golden, B., Kelly, J., Resende, M., and Stewart Jr, W. Designing and reporting on computational experiments with heuristic methods. *Journal of Heuristics 1* (1995), 9–32.

[9] Beasley, J., and Nascimento, E. The vehicle routing-allocation problem: a unifying framework. Tech. rep., The Management School, Imperial College, London, England, June 1995.

[10] Beltrami, E., and Bodin, L. Networks and vehicle routing for municipal waste collection. *Networks 4* (1974), 65–94.

[11] Benavent, E., Campos, V., Corberan, A., and Mota, E. The capacitated arc routing problem: lower bounds. *Networks 22* (1992), 669–690.

[12] Bodin, L., and Levy, L. The arc partitioning problem. *European Journal of Operational Research 53* (1991), 393–401.

[13] Butt, S., and Cavalier, T. A heuristic for the multiple tour maximum collection problem. *Computers and Operations Research 21*, 1 (1994), 101–111.

[14] Carnes, J., and Cooke, J. Pod sheet v2.0 instructions. ftp://ftp.ecst.csuchico.edu/pub/ems/sar/programs/podsht20.zip.

[15] Chapleau, L., Ferland, J., Lapalme, G., and Rousseau, J.-M. A parallel insert method for the capacitated arc routing problem. *Operations Research Letters 3* (1984), 95–99.

[16] Chew, L. Constrained delaunay triangulations. *Algorithmica 4* (1989), 97–108.

[17] Chin, W., and Ntafos, S. Optimum watchman routes. *Information Processing Letters 28* (1988), 39–44.

[18] Christofides, N. The optimum traversal of a graph. *OMEGA 1*, 6 (1973), 719–732.

[19] Clark, J. Derivation of simplification. http://www.sarinfo.bc.ca/Library/Planning/POAderiv.pln, August 1995.

[20] Colwell, M. Sound sweep: A new tool for search teams. *SAR Technical Publications Series 1*, 2 (1992).

[21] Colwell, M. Planning the grid search. http://www.sarinfo.bc.ca/Library/Planning/GridPlan.doc, August 1995.

[22] Colwell, M. Accurate PODs and efficient SAR team deployments. http://www.sarinfo.bc.ca/Library/Planning/PODEffic.pln, April 1996.

[23] Colwell, M. Prioritizing multiple SAR resource deployments to the same search area. http://www.sarinfo.bc.ca/Library/Planning/SrchPrsq.pln, May 1996.

[24] Colwell, M. Search efficiency: bringing science to SAR. http://www.sarinfo.bc.ca/Sartechs.htm, August 1996.

[25] COLWELL, M. The sound sweep - an efficient new ground search technique. http://www.sarinfo.bc.ca/Sartechs.htm, August 1996.

[26] COLWELL, M. Search priority - an integrated approach to search planning. http://www.sarinfo.bc.ca/Library/Planning/srchpri.doc, May 1997.

[27] COLWELL, M. Trail-based probability of area: a terrain-based approach to POA estimation. http://www.sarinfo.bc.ca/Library/Planning/trailpoa.doc, January 1997.

[28] COLWELL, M. Unifying geographical search assignments: coordinating and prioritizing both area and trail searches. http://www.sarinfo.bc.ca/Library/Planning/TrailPriority.doc, October 1998.

[29] COLWELL, M. POD calibration curves. Personal communication via email, June 1999.

[30] COOPER, D. Re:Reassessing POA? Email to the Search and Rescue Research and Development List, 20 January 1998.

[31] CORNELL, E. Modified Mattson needs revision. Email to the Search and Rescue Research and Development List, 11 December 1997.

[32] CORNELL, E., AND HETH, C. Distance travelled during urban and suburban walks led by 3 to 12 year olds: tables for search managers. http://www.sarinfo.bc.ca/Sartechs.htm, May 1997.

[33] CURRENT, J., PIRKUL, H., AND ROLLAND, E. Efficient algorithms for solving the shortest covering path problem. *Transportation Science 28*, 4 (1994), 317–327.

[34] CURRENT, J., REVELLE, C., AND COHON, J. The shortest covering path problem: an application of locational constraints to network design. *Journal of Regional Science 24*, 2 (1984), 161–183.

[35] CURRENT, J., REVELLE, C., AND COHON, J. The maximum covering/shortest path problem: A multiobjective network design and routing formulation. *European Journal of Operational Research 21* (1985), 189–199.

[36] CURRENT, J., AND SCHILLING, D. The covering salesman problem. *Transportation Science 23*, 3 (1989), 208–213.

[37] CURRENT, J., AND SCHILLING, D. The median tour and maximal covering tour problems: formulations and heuristics. *European Journal of Operational Research 73* (1994), 114–126.

[38] DERIGS, U. Solving non-bipartite matching problems via shortest path techniques. *Annals of Operations Research 13* (1988), 225–261.

[39] DILLENCOURT, M. A non-Hamiltonian, nondegenerate Delaunay triangulation. *Information Processing Letters 25* (1987), 149–151.

[40] DILLENCOURT, M. Travelling salesman cycles are not always subgraphs of Delaunay triangulations or of minimum weight triangulations. *Information Processing Letters 24* (1987), 339–342.

[41] DINGLE, G. *Outdoor Pursuits in New Zealand.* Reed Books, Octopus Publishing Group, 1991.

[42] DITTMER, D. Search and Rescue Advisor, Levin, New Zealand. Personal communication, May 1996.

[43] DOWNS, M., AND DOLAN, M. Sardat: A search database. *Response* (winter 1995). http://www.nasar.org/prod/members/library/response_1995_winter_downs.html.

[44] DROR, M., AND POWELL, W. Stochastic and dynamic models in transportation. *Operations Research 41*, 1 (1993), 11–14.

[45] DROR, M., STERN, H., AND TRUDEAU, P. Postman tour on a graph with precedence relation on arcs. *Networks 17* (1987), 283–294.

[46] EDMONDS, J., AND JOHNSON, E. Matching, Euler tours and the Chinese postman. *Mathematical Programming 5* (1973), 88–124.

[47] EILON, S., AND CHRISTOFIDES, N. The loading problem. *Management Science 17*, 5 (1971), 259–268.

[48] EISELT, H., GENDREAU, M., AND LAPORTE, G. Arc routing problems, part 2: The Rural Postman Problem. *Operations Research 43*, 2 (1995), 399–414.

[49] ENGEL, D., AND WEISINGER, J. Estimating visual detection performance at sea. *Operations Research 36*, 5 (1988), 651–659.

[50] FEENEY, B. *Land Search and Rescue*, 4th ed. New Zealand Police, 1992.

[51] FIALA TIMLIN, M., AND PULLEYBLANK, W. Precedence constrained routing and helicopter scheduling: Heuristic design. *Interfaces 22* (1992), 100–111.

[52] FISCHBECK, P. GIS: More than a map. *OR/MS Today* (August 1994), 42–45.

[53] Foulds, L. The heuristic problem-solving approach. *Journal of the Operational Research Society 34*, 10 (1983), 927–934.

[54] Frederickson, G., Hecht, M., and Kim, C. Approximation algorithms for some routing problems. *SIAM Journal of Computing 7*, 2 (1978), 178–193.

[55] Freund, J. *Modern Elementary Statistics*, 7th ed. Prentice-Hall, New Jersey, 1988.

[56] Frost, J. Clue detection probabilities. Email to the Search and Rescue Research and Development List, 2 June 1998.

[57] Frost, J. Clues and information theory. Email to the Search and Rescue Research and Development List, 28 September 1998.

[58] Frost, J. Re: potential NFPA search committee. Email to the Search and Rescue Research and Development List, 2 November 1998.

[59] Frost, J. Re:clue detection probabilities. Email to the Search and Rescue Research and Development List, 6 June 1998.

[60] Frost, J. Re:clue detection probabilities. Email to the Search and Rescue Research and Development List, 10 June 1998.

[61] Frost, J. Re:clue detection probabilities. Email to the Search and Rescue Research and Development List, 3 June 1998.

[62] Frost, J. Principles of search theory part four: optimal effort allocation. http://www.sarinfo.bc.ca/Library/Planning/SrchThy4.doc, April 1999.

[63] Frost, J. Principles of search theory part one: detection. http://www.sarinfo.bc.ca/Library/Planning/SrchThy1.doc, March 1999.

[64] Frost, J. Principles of search theory part three: probability density distributions. http://www.sarinfo.bc.ca/Library/Planning/SrchThy3.doc, March 1999.

[65] Frost, J. Principles of search theory part two: effort, coverage and POD. http://www.sarinfo.bc.ca/Library/Planning/SrchThy2.doc, March 1999.

[66] Gallas, F. *Land search and rescue*, 2nd ed. Federated Mountain Clubs of New Zealand, 1981.

[67] Gatrell, A. Concepts of space and geographic data. In *Geographical Information Systems: Principles and Applications*, D. Maguire, M. Goodchild, and D. Rhind, Eds., vol. 1. Longman Scientific and Technical, Essex, England, 1991, pp. 119–134.

[68] GENDREAU, M., LAPORTE, G., AND SEMET, F. The covering tour problem. Tech. Rep. CRT-95-08, Centre de Recherche sur les Transports, 1995.

[69] GENDREAU, M., LAPORTE, G., AND SEMET, F. The covering tour problem. *Operations research 45*, 4 (1997), 568–576.

[70] GOEMANS, M., AND WILLIAMSON, D. A general approximation technique for constrained forest problems. Working Paper, 1994.

[71] GOLD, C., AND CORMACK, S. Spatially ordered networks and topographic reconstructions. *International Journal of Geographical Information Systems 1*, 2 (1987), 137–148.

[72] GOLDEN, B., DEARMON, J., AND BAKER, E. Computational experiments for a class of routing problems. *Computers and Operations Research 10* (1983), 47–59.

[73] GOLDEN, B., LEVY, L., AND DAHL, R. Two generalizations of the travelling salesman problem. *OMEGA 9*, 4 (1981), 439–441.

[74] GOLDEN, B., AND WONG, R. Capacitated arc routing problems. *Networks 11* (1981), 305–315.

[75] GOLDRING, R., AND MULLINS, W., Eds. *Bushcraft - Outdoor Skills for the New Zealand Bush*, 3rd ed. New Zealand Mountain Safety Council Inc., 1995.

[76] GOODCHILD, M., AND LEE, J. Coverage problems and visibility regions on topographic surfaces. *Annals of Operations Research 18* (1989), 175–186.

[77] GOODMAN, R. The real truth about POA/POD: an opinion. In *Probability of Detection (POD) research and other concepts for search management.* Emergency Response Institute, 1989, pp. 59–65.

[78] GOODMAN, R., AND COWAN, B. The "POC" factor. Working Paper, 1996.

[79] GORDON, R. Night searching - a component of managing search operations courses. *Emergency Services Review 2* (1995), 11–12.

[80] GORDON, R. Managing Director, Emergency Management Limited, New Zealand. Personal communication, August 1996.

[81] GORDON, R. Managing Director, Emergency Management Limited, New Zealand. Personal communication, May 1996.

[82] GUAN, M. On the windy postman problem. *Discrete Applied Mathematics 9* (1984), 41–46.

[83] GUAN, M., AND PULLEYBLANK, W. Eulerian orientations and circulations. *SIAM Journal of Algorithms and Discrete Mathematics 6*, 4 (1985), 657–664.

[84] HALL, N., AND HOCHBAUM, D. The multicovering problem. *European Journal of Operational Research 62* (1992), 323–339.

[85] HALL, R., AND PARTYKA, J. On the road to efficiency. *OR/MS Today* (June 1997), 38–47.

[86] HANDEL, T. Relative urgency response factors. Internet document ftp.ecst.csuchico.edu/pub/ems/sar/programs/rurf.zip, May 1995.

[87] HARVEY, D. Personal communication, January 1997.

[88] HILL, K. An introduction to ground search management for the police authority. http://www.sarbc.org/k-hill.html, 1994.

[89] HILL, K. How many clues do you need? Applying information theory to land search. *Response* (fall 1995). http://www.nasar.org/prod/members/library/response_1995_fall_hill.html.

[90] HILL, K. Scenario analysis in land search planning. http://www.sarbc.org/k-hill.html, 1995.

[91] HILL, K., AND BOWER, D. GPS navigation in land search. http://www.sarbc.org/k-hill.html, August 1996.

[92] JENNEX, P. POD-aircraft. http://www.sarinfo.bc.ca/Library/Planning/Aircfpod.pln, April 1996.

[93] JONES, A. Leadership in search and rescue. In *Probability of Detection (POD) research and other concepts for search management.* Emergency Response Institute, 1989, pp. 68–71.

[94] JONES, C., KIDNER, D., AND WARE, J. The implicit triangulated network and multiscale spatial databases. *The Computer Journal 37*, 1 (1994), 43–57.

[95] KEANE, P. Aircraft search PODs. http://www.sarinfo.bc.ca/Library/Planning/Podair.doc, March 1996.

[96] KEANE, P. Information, misinformation, or disinformation; the SAR intelligence quandry. http://www.sarinfo.bc.ca/Sartechs.htm, August 1996.

[97] KELLER, C., AND GOODCHILD, M. The multiobjective vending problem: a generalization of the travelling salesman problem. *Environment and Planning B: Planning and Design 15* (1988), 447–460.

[98] KETCHAM, K. Digital imaging provides clear view for search and rescue efforts. *Hazard Technology 15*, 1 (1995), 16.

[99] KNOX, R. Search and rescue. *New Zealand's Heritage: the making of a nation 7*, 103 (1971).

[100] KOOPMAN, B. *Search and Screening - General Principles with Historical Applications.* Pergamon Press, New York, 1980.

[101] LAPALME, G., ROUSSEAU, J., CHAPLEAU, S., CORMIER, M., COSSETTE, P., AND ROY, S. GEOROUTE: A geographic information system for transportation applications. *Communications of the ACM 35*, 1 (1992), 80–88.

[102] LAURINI, R., AND THOMPSON, D. *Fundamentals of spatial information systems.* Academic Press, London, 1992.

[103] LAVALLA, P., AND STOFFEL, R. *Search Is An Emergency*, 2nd ed. Emergency Response Institute, 1987.

[104] LEE, D., AND SCHACHTER, B. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer and Information Sciences 9*, 3 (1980), 219–242.

[105] LEE, J. Analyses of visibility sites on topographic surfaces. *International Journal of Geographical Information Systems 5*, 4 (1991), 413–429.

[106] LEE, J. Comparison of existing methods for building triangular irregular network models of terrain from grid digital elevation models. *International Journal of Geographical Information Systems 5*, 3 (1991), 267–285.

[107] LENSTRA, J., AND RINNOOY KAN, A. On general routing problems. *Networks 6* (1976), 273–280.

[108] LEVY, L., AND BODIN, L. The arc oriented location routing problem. *INFOR 27*, 1 (1989), 74–94.

[109] LOKIN, F. Procedures for TSP with additional constraints. *European Journal of Operational Research 3* (1978), 135–141.

[110] MACEDONIO, G., AND PARESCHI, M. An algorithm for the triangulation of arbitrarily distributed points: Applications to volume estimate and terrain fitting. *Computers and Geosciences 17*, 7 (1991), 859–874.

[111] MAGUIRE, D. An overview and definition of GIS. In *Geographical Information Systems: Principles and Applications*, D. Maguire, M. Goodchild, and D. Rhind, Eds., vol. 1. Longman Scientific and Technical, Essex, England, 1991, pp. 9–20.

[112] MALANDRAKI, C., AND DASKIN, M. The maximum benefit chinese postman problem and the maximum benefit traveling salesman problem. *European Journal of Operational Research 65* (1993), 218–234.

[113] MANDELBAUM, M., AND BUZACOTT, J. Flexibility and decision making. *European Journal of Operational Research 44* (1990), 17–27.

[114] MARIN COUTY SHERIFF'S DEPARTMENT SEARCH AND RESCUE TEAM. Marin county incident pre plan. http://www.marin.org/npo/search/preplan.htm, December 1994.

[115] MARTIN, Y. Cost of getting lost. *Sunday Star-Times* (12 January 1997).

[116] MCCONAUGHY, S. Decreasing search segment size and changing boundaries to increase POD (coverage) of the search area. http://www.sarinfo.bc.ca/Library/Planning/SegResize.pln, August 1997.

[117] MCDONALD, A. Preliminary design of a ground search and rescue incident database. http://www.sarinfo.bc.ca/Library/Planning/database.pln, May 1997.

[118] MINIEKA, E. The Chinese postman problem for mixed networks. *Management Science 25*, 7 (1979), 643–648.

[119] MITCHELL, J. An algorithmic approach to some problems in terrain navigation. *Artificial Intelligence 37* (1988), 171–201.

[120] NEBIKER, R. Five suggestions to save OR. *OR/MS Today* (August 1995), 10–11.

[121] NEW ZEALAND POLICE. Search and rescue guidelines. Policy report.

[122] NEW ZEALAND POLICE. Search and rescue statistics. Annual.

[123] NEW ZEALAND POLICE. Radio communication. New Zealand Police Search and Rescue Course, April 1996.

[124] ORLOFF, C. A fundamental problem in vehicle routing. *Networks 4* (1974), 35–64.

[125] ORLOFF, C. Routing a fleet of M vehicles to/from a central facility. *Networks 4* (1974), 147–162.

[126] ORLOFF, C. On general routing problems: comments. *Networks 6* (1976), 281–284.

[127] O'ROURKE, J. *Art Gallery Theorems and Algorithms.* Oxford University Press, 1987.

[128] O'ROURKE, J. *Computational geometry in C.* Cambridge University Press, 1994.

[129] PAIRMAN, D. Programme Leader: Information Engineering for Remote Sensing, Landcare Research, New Zealand. Personal communication via email, October 1999.

[130] PAPADIMITRIOU, C., AND STEIGLITZ, K. *Combinatorial Optimization: Algorithms and Complexity.* Prentice-Hall, New Jersey, 1982.

[131] PAPIDIMITRIOU, C. On the complexity of edge traversing. *Journal of the Association for Computing Machinery 23*, 3 (1976), 544–554.

[132] PEARN, W. New lower bounds for the capacitated arc routing problem. *Networks 18* (1988), 181–191.

[133] PEARN, W. Solvable cases of the k-person Chinese postman problem. *Operations Research Letters 16* (1994), 241–244.

[134] PERKINS, D. A consensus method for search sector ranking which leads to POAs. In *Probability of Detection (POD) research and other concepts for search management.* Emergency Response Institute, 1989, pp. 44–49.

[135] PERKINS, D. Motorcycle maintenance and the art of searching. In *Probability of Detection (POD) research and other concepts for search management.* Emergency Response Institute, 1989, pp. 40–43.

[136] PERKINS, D. A simple approach to POS optimisation. In *Probability of Detection (POD) research and other concepts for search management.* Emergency Response Institute, 1989, pp. 50–58.

[137] PERKINS, D., AND ROBERTS, P. Critical separation. In *Probability of Detection (POD) research and other concepts for search management.* Emergency Response Institute, 1989, pp. 1–39.

[138] PERKINS, D., AND ROBERTS, P. Who is in charge of your search plan? *Response* (Fall 1995). http://www.nasar.org/prod/members/library/response_1995_fall_perkins.html.

[139] POTVIN, J., LAPALME, G., AND ROUSSEAU, J. ALTO: a computer system for the design of vehicle routing algorithms. *Computers and Operations Research 16*, 5 (1989), 451–470.

[140] PREPARATA, F., AND SHAMOS, M. *Computational Geometry - An Introduction.* Springer-Verlag, New York, 1985.

[141] PSARAFTIS, H. Dynamic vehicle routing problems. In *Vehicle Routing: Methods and Studies*, B. Golden and A. Assad, Eds., vol. 16 of *Studies in Management Science and Systems.* Elsevier Science Publishers, North-Holland, Amsterdam, 1988, pp. 223–248.

[142] REEVES, C., Ed. *Modern heuristic techniques for combinatorial problems.* John Wiley & Sons, Inc., New York, 1993.

[143] RICHARDSON, H., AND DISCENZA, J. The United States Coast Guard Computer-assisted Search Planning System (CASP). *Naval Research Logistics Quarterly 27* (1980), 659–680.

[144] ROSING, K. An optimal method for solving the (generalized) multi-Weber problem. *European Journal of Operational Research 58* (1992), 414–426.

[145] ROY, S., AND ROUSSEAU, J. The capacitated Canadian postman problem. *INFOR 27*, 1 (1989), 58–73.

[146] SÉGUIN, R., POTVIN, J.-Y., GENDREAU, M., CRAINIC, T., AND MARCOTTE, P. Real-time decision problems: an operational research perspective. *Journal of the Operational Research Society 48* (1997), 162–174.

[147] SIBSON, R. Locally equiangular triangulations. *The Computer Journal 21* (1978), 243–245.

[148] SILVER, E., VIDAL, R., AND DE WERRA, D. A tutorial on heuristic methods. *European Journal of Operational Research 5* (1980), 153–162.

[149] SLATER, P. Locating central paths in a graph. *Transportation Science 16*, 1 (1982), 1–18.

[150] SOZA & COMPANY, LTD, AND OFFICE OF SEARCH AND RESCUE, U.S. COAST GUARD. The theory of search: a simplified explanation. http://www.soza-intl.com/sci-mcs.htm, October 1998.

[151] STERN, H., AND DROR, M. Routing electric meter readers. *Computers and Operations Research 6* (1979), 209–223.

[152] STONE, L, D. *Theory of Optimal Search.* Academic Press, New York, 1975.

[153] STONE, L. The process of search planning: current approaches and continuing problems. *Operations Research 31*, 2 (1983), 207–233.

[154] STONE, L. What's happened in search theory since the 1975 Lanchester prize? *Operations Research 37*, 3 (1989), 501–506.

[155] SWEERE, K. Optimization techniques in the search for a lost person. http://www.sarinfo.bc.ca/Library/Planning/optimise.pln, March 1995.

[156] SWEERE, K. Superior Search and Rescue's training manual. http://www.civil.mtu.edu/~hssantef/sar/bookEd4/contents.html, May 1997. 4th edition.

[157] SWOMBOW, C. The changing subject profile. In *Probability of Detection (POD) research and other concepts for search management.* Emergency Response Institute, 1989, pp. 66–67.

[158] SWOMBOW, C. The searching science. *Police Science and Technology Review* (May 1996), 26–32.

[159] SYROTUCK, W. An introduction to land search probabilities. Arner Publications, 1975.

[160] SYSLO, M., DEO, N., AND KOWALIK, J. *Discrete optimization algorithms with Pascal programs.* Prentice-Hall, 1983.

[161] TATE, D. Modified Mattson needs revision. Email to the Search and Rescue Research and Development List, 11 December 1997.

[162] TELCIK, T. Vehicle navigation using a triangulated irregular network: solving the cross-country problem. Honours Dissertation, Curtin University of Technology, Australia, November 1992.

[163] TELEVISION NEW ZEALAND. Inside New Zealand: Knights of the air. Television documentary, May 1996.

[164] THOMAS, L., AND HULME, P. Searching for targets who want to be found. *Journal of the Operational Research Society 48* (1997), 44–50.

[165] TRISTRAM, J. New Zealand Land Search and Rescue Coordinator. Personal communication, February 1996.

[166] TRUMMEL, K., AND WEISINGER, J. The complexity of the optimal searcher path problem. *Operations Research 34*, 2 (1986), 324–327.

[167] TSILIGIRIDES, T. Heuristic methods applied to orienteering. *Journal of the Operational Research Society 35*, 9 (1984), 797–809.

[168] TVERSKY, A., AND KAHNEMAN, D. Judgement under uncertainty: heuristics and biases. In *Judgement under uncertainty: heuristics and biases*, D. Kahneman, P. Slovic, and A. Tversky, Eds. Cambridge University Press, New York, 1982, pp. 3–20.

[169] WEDDLE, C. Search techniques. http://www.sarinfo.bc.ca/Sartechs.htm, August 1996.

[170] WEDDLE, C. Team membering. http://www.sarinfo.bc.ca/Sartechs.htm, August 1996.

[171] WEIBEL, R., AND HELLER, M. Digital terrain modelling. In *Geographical Information Systems: Principles and Applications*, D. Maguire, M. Goodchild, and D. Rhind, Eds., vol. 1. Longman Scientific and Technical, Essex, England, 1991, pp. 269–297.

[172] WIN, Z. On the windy postman problem on Eulerian graphs. *Mathematical Programming 44* (1989), 97–112.

[173] ZANAKIS, S., AND EVANS, J. Heuristic 'optimization': why, when, and how to use it. *Interfaces 11*, 5 (1981), 84–91.