Modeling RNA Evolution: In-Silico and In-Vivo

Part 1: RNA Evolution in the RNA World.

Part 2: The Evolution of a RNA Virus: RS Virus in New Zealand

A thesis presented

for the degree of

Master of Science

in

BioMathematics

at Massey University, Palmerston North

New Zealand

James William Matheson

2004

Abstract

We look at two aspects of the evolution of RNA.

First we look at RNA replication dynamics in an early RNA world context. Experimental evidence (Spiegelman *et al.* 1965, Biebricher *et al.* 1981) shows that under some conditions RNA evolves towards small quickly replicating molecules. We investigate what conditions are sufficient for a population of RNA molecules to evolve towards a balanced population of molecules. This is a population not completely dominated by a single length of molecule. We consider two models: A linear model in which indel rate is inversely proportional to length and a game theory model in which reproductive efficiency depends on the distribution of molecule lengths within a population (this is linked to catalytic efficiency). Models are investigated using analytic, numerical and simulation methods. The linear model is not sufficient to support a population with balanced length distribution. Simulation methods show that the game theory model may support such a population.

We next look at RNA evolution in the context of RNA virus evolution. Using virus samples taken over a thirty year period we investigate the evolution of Respiratory Syncytial Virus (RSV) in New Zealand. RSV most strongly affects infants and the elderly, causing cold like symptoms in mild cases and bronchiolitis or occasionally death in severe cases. New Zealand has a higher incidence of RSV bronchiolitis per head of population than many other developed countries. We compare New Zealand strains of the virus to those isolated overseas to investigate if New Zealand may have significantly different strains. We look at the evolution of the virus within New Zealand looking for evidence of antigenic drift, as well as analysing substitution rates and selection at individual codon sites. No evidence is found to suggest that New Zealand has significantly different strains of RSV from other countries. We conclude the higher rate of severe RSV in New Zealand must be caused by factors other than virus strain. The portion of the virus analysed shows strong evidence of being under positive selective pressure. This and other analyses suggest that RSV may be undergoing antigenic drift.

Acknowledgements

I would like to say a special thank you to my supervisors, Mike Hendy, David Penny and Barbara Holland for their support, and for the many drafts of this document they have read and corrected. All mistakes that remain are my own.

I would like to thank the Allan Wilson Centre and IFS for their financial support. Thank you to my friends and colleagues there: Klaus, Michael, Tim, Bhalchandra and Barbara for keeping me sane. Thank you to Tim for putting up with so many computer questions and last but not least to Susan and Joy for helping with all the day to day things.

Thank you to Joanna Kirman, Fenella Rich and Catherine Cohet at the Malaghan Institute, without whom I would not have been able to do the RSV section of this thesis. Thank you for putting up with my ignorance of immunology and for sequencing all those snot samples.

Thank you also to Dr. Keith Grimwood of Wellington Hospital and Sue Huang of ESR for your part in the RSV project.

Thanks go to:

- Wim Hordijk for help with the initial simulations.
- Igor Boglaev and Andreas Dress for your help with the differential equations description of the game theory.
- Allen Rodrigo for taking the time to tell me about your work.
- Brett Ryland for suggesting a bilinear interpolation.

- James Chai, for putting up with my many enquiries about Helix.

Finally thanks go to my family for being so supportive, especially in my transition from PhD to Masters.

James W. Matheson

5th of December 2004.

Preface

This project initially started as a PhD thesis and was transmuted into a Masters thesis after one year for personal reasons. In its initial stages the project focused on RNA replication dynamics in an RNA world situation. Simulations for this study were done on the Helix supercomputing cluster.
I also wanted to do some more practical work so when data became available at the Malaghan Institute on Respiratory Syncytial Virus (RSV) I agreed to analyse it as part of a project investigating RSV in New Zealand lead by Dr. Joanna Kirman of the Malaghan Institute.

The majority of the analyses in both parts of this project were done using R (R Development Core Team 2004). High speed simulation code was written in ANSI C. Parallel code used the MPI Parallel library.

Motivation: RNA, with its ability to encode both genotype (sequence) and phenotype (folding) in the same molecule is thought to have preceded DNA and protein as a carrier of genetic information by some scientists. An RNA world, in which RNA is the primary information carrying and catalytic molecule, is postulated to have been the first stage of evolution. The plausibility and structure of such a world rests on how RNA behaves. This motivates the study of early RNA evolution.

The study of RSV is motivated primarily by its medical significance. RSV has it largest effect in infants where it causes cold like symptoms in mild cases and bronchiolitis and occasionally death in severe cases. RSV data over a long time period (30 years) is not often available, and there is little data available on New Zealand RSV, so the discovery of New Zealand RSV samples from 1967 to 1997 in an ESR freezer offers a useful opportunity to study this virus.

Contents

List of Figures

List of Tables

Glossary of terms and abbreviations.

Asymmetric Indel rate 6

The rate at which insertions occur does not equal the rate at which deletions occur.

Asymmetry Parameter 7

Determines the relative probability of insertions and deletions.

Bilinear Interpolation 35

Estimation of internal points of a surface by linear interpolation from two boundaries.

Cluster Measure 67

Measures the degree of homogeneity of a sequence.

Copy Error 5

The rate at which copy errors occur during replication given as a probability per replication of an error occurring.

Cytodomain 49

The part of a protein inside the viral capsid.

Degree of Congruence 65

Extent to which split partitions coincide with partitions defined by identical property values.

Degree of Homogeneity 66

Measure of the degree to which terms of a sequence are identical.

DNA 2

Deoxyribonucleic Acid. Molecule that encodes genetic information of all cellular life forms.

1 Introduction

This thesis investigates several aspects of the evolution of biological molecules. RNA is the molecule we concentrate on. We take a brief look at RNA in section 1.1.

In Section 2 we look at how RNA might have been involved in the early development of life. Eigen (1971) considers constant length self replicating biomolecules (specifically RNA). Eigen's model goes from early chemical self organization in evolution to the formation of hypercycles, which are self-replicating autocatalytic cycles. There is some argument over the veracity of his claims for hypercycles as the path to more complex biomolecules (Boerlijst and Hogeweg 1991, Zintzaras *et al.* 2002), however our area of interest lies before the formation of hypercycles in the area of RNA evolution. We look at variable length self replicating biomolecules and ask the question: Can we find a set of conditions sufficient to create a stable population of molecules with a balanced length distribution? This question arises from experimental observations (Spiegelman *et al.* 1965, Biebricher *et al.* 1981) showing that, under some conditions, RNA will evolve towards a highly biased (short) length distribution. This unlike what is seen in nature today. We attempt to answer this question by using mathematical models and simulation methods to investigate RNA replication dynamics in different model systems.

In section 3 we move from the theoretical to the practical implications of RNA evolution. We look at the evolution of Respiratory Syncytial Virus (RSV) in New Zealand. RSV is common in infants. Mild infection causes symptoms similar to a common cold, severe infection can cause bronchiolitis and death. New Zealand has a higher incidence of hospital admissions from RSV bronchiolitis than many other developed countries (Vogel *et al.* 2003). We investigate if this is due to New Zealand having different strains of RSV to other countries, as well as looking at the characteristics of the virus's evolution in New Zealand.

1.1 What is RNA?

RNA stands for **R**ibo**N**ucleic **A**cid which is a single stranded cousin to DNA. RNA like DNA is constructed from a linear chain of nucleotides attached to a sugar phosphate backbone. Unlike DNA, RNA is usually single stranded and the nucleotides used are Adenine, Cytosine, Guanine and Uracil (not Thymine) often abbreviated to A, C, G and U.

Due to their single stranded nature RNA molecules are free to fold back on themselves. This means that nucleotides that fold to be adjacent can form hydrogen bonds (base-pair) creating 2D structures such as loops and hairpins. Different nucleotides form different strengths of bond. The strongest bond is $C \equiv G$ followed by $A = U$. There is also weak binding affinity in the bond $C - A$. Bases with strong base pairing affinities are said to complement each other, C and G are complementary base pairs as are A and U. The 2D structures formed by RNA can in turn fold in 3D space to form complex structures. Some of these 3D structures will provide the chemical binding sites which allow RNA to catalyse chemical reactions.

Modern theories of the origin of life assume an RNA-world stage (Yarus 1999). This is a stage of evolution that is dominated by RNA. In these theories founding populations of RNA molecules are produced by natural RNA synthesis from nucleotides on ancient earth. RNA replication is aided by catalytic RNA called ribozymes. The founding population gradually evolves, by mechanisms such as that discussed in Eigen (1971), towards the production of protein and eventually DNA. The situation in the RNA-world differs from the modern situation; in the RNA-world RNA was both catalyst and information carrying molecule. In the modern situation the functions of catalyst and information carrier are separated between protein and DNA respectively. RNA catalysts for RNA processing (ribozymes) are essential to the RNA-world hypothesis as they form the basis for theories of self-replicating systems of RNA molecules. Though the existence of efficient ribozymes is yet to be experimentally proven, there is reason to believe it will be, with groups such as that of David Bartel (Lawrence and Bartel 2003) finding molecules that have good RNA polymerase activity but limited processivity (their ability to catalyse other molecules is

not long lived).

In a hypothetical RNA world environment containing free nucleotides and ribozymes some RNA sequences can undergo replication (Spiegelman *et al.* 1965). This process involves a complementary copy of the molecule being created from the original by pairing each base in the original unwound strand with its complement. This complement is complemented in turn to create a replica of the original sequence. There are no known error correcting mechanisms in this process so RNA replication is prone to errors. Errors can take the form of miscoded bases (called 'substitutions') or the addition or deletion of bases from the molecule (these are called insertions and deletions respectively and are collectively referred to as indels).

In this thesis we look at a formal model of aspects of the early RNA world (section 2) as well as how RNA, in the form of RNA viruses, evolves in the world today (section 3).

2 The Evolution of Biological Molecules.

In this section we look at the replication dynamics of RNA in an early RNA world scenario. One of the most well known models of molecular replication dynamics is that of Eigen (1971). In his model Eigen (1971) looks at RNA molecules as molecules of fixed length that can have substitution errors during replication (discussed further in section 2.1). In our models we consider RNA to be a variable length molecule that can have indels during replication; we ignore the effects of substitutions. Here and throughout this thesis the length of a molecule refers to the number of bases in that molecule. Experiments with RNA replication in test tubes by Spiegelman *et al.* (1965) and Biebricher *et al.* (1981) have shown that when catalysed by a fixed protein enzyme and replicated in a test tube (where its length can vary) RNA evolves towards a short faster replicating molecule. This tendency toward the dominance of short molecules is clearly counteracted by some process in nature as we can observe a wide distribution of molecular lengths in nature today. This leads to the question: What set of conditions is sufficient to create a stable population of molecules with a balanced length distribution?

To answer this question we must define what we mean by a balanced distribution. A balanced distribution is a distribution which is not dominated by molecules of a single or only a few lengths and where there is no net tendency for molecules to change in size. Molecules of many different lengths coexist in a balanced distribution. However it should be noted the distribution is not necessarily symmetric or unimodal.

The aim of the first part of this thesis is to find a simple model that produces a stable population of molecules with a balanced length distribution. Using analytic, numerical and simulation methods we investigate several candidate models. This will lead to some insight into the basic mechanisms at work in these situations. Spiegelman *et al.* (1965) used Qβ phage RNA under repeated amplification using Qβ replicase to achieve his results. In Spiegelman's experiment there is no feedback between the catalyst and the molecule being catalysed, they are completely separate. By introducing feedback to the reaction system we hypothesize that a stable mixed length population of molecules could be produced.

The goal of these models is to investigate the mechanisms necessary for the production of a population of molecules with a balanced length distribution. To this end we ignore factors such as genealogy, and secondary and tertiary structure of structure of the molecules. We focus on the length of the molecule. This is a drastic simplification. However it allows us to study length effects that have not been studied in previous models while isolating them from those described in previous models such as Eigen (1971). This simplification results in both a mathematically tractable model and the ability to more clearly see effects that may otherwise be hidden in complexity. Due to this abstraction the model is also relevant to other autocatalytic species.

2.1 Indel Rate Models

A factor that has significant effect on the length of molecules is the copy error rate. This is the rate at which copy errors occur during replication given as a probability per replication of an error occurring. The influence of copy error rate on a population of molecules is discussed in detail by Eigen and Schuster (1977) in their paper on error catastrophe. In Eigen and Schuster (1977) only point mutations (substitutions) are considered. In contrast this thesis looks only at indels. Eigen and Schuster showed that if there is a per base copy error rate e in copying a molecule then the probability of a molecule above a certain length accurately reproducing itself falls quickly to zero in a phase transition called an error catastrophe (or more recently a mutational meltdown), once e is increased past a critical value (Eigen 1971). In this way copy error rate regulates the length of self replicating molecules. This suggests copy errors as a possible mechanism for regulating the distribution of molecular lengths within a population. Here we investigate the effects of a specific type of copy error, the indel. When a molecule is copied in nature its length can be increased or decreased by insertions or deletions respectively. We hypothesise that longer molecules, due to their increased information content and more complex structure, could in principle be more accurate in their catalysis and thus have a lower probability of having an indel per replication than short molecules. In this circumstance shorter molecules will be less stable and will more frequently change to a different length while longer molecules will be more stable and more frequently stay the same length. We investigate if this

scenario could lead to a stable distribution in the rest of this section. Several different types of model are used. Replication rate in these models refers to the time taken for one full replication of a molecule. The meanings of the terms used to describe the models are given in table 2.1.

Table 2.1: **Model Type Definitions.**

Model Type Name	**Definition**
Uniform Replication	Replication rate is uniform over all molecules irrespective of their length. This implies a faster per nucleotide copy rate for larger molecules.
Nonuniform replication	Replication rate is dependent on a molecules length.
Symmetric indel rate	The probability of an insertion occurring for a molecule of given length is the same as the probability of a deletion occurring (except at the boundaries which are a special case).
Asymmetric indel rate	The probability of an insertion occurring for a molecule of given length need not equal the probability of a deletion occurring.

The first model we investigate has a length dependent indel rate with uniform replication, this is the "Uniform-Replication Symetric-Indel-rate" (Ur-Si) model detailed in section 2.1.2. We then look at nonuniform replication to get the "Nonuniform-Replication Symetric-Indel-rate" (Nr-Si) model detailed in section 2.1.3. Both the Ur-Si model and Nr-Si models have equal probability of insertions and deletions occurring. The "Uniform-Replication Asymmetric-Indel-rate" (Ur-Ai) model detailed in section 2.1.4 and the "Nonuniform-

Replication Asymmetric-Indel-rate" (Nr-Ai) model, section 2.1.5, use asymmetric insertion and deletion probabilities. First we will look at the properties common to all four of these models, collectively referred to as the indel rate models. The indel rate models are summarized in table table 2.2. The indel rate models do not contain the feedback mechanism we have hypothesised necessary for the formation of a balanced population. We look at these models as a control case.

Table 2.2: **The four types of indel rate model investigated.** Models can be distinguished by replication rate (uniform and nonuniform) and by indel probabilities (symmetric and asymmetric).

	Uniform Replication Rate	**Nonuniform Replication Rate**
Symmetric Indel Probabilities	Ur-Si model (section 2.1.2)	Nr-Si model (section 2.1.3)
Asymmetric Indel Probabilities	Ur-Ai model (section 2.1.4)	Nr-Ai model (section 2.1.5)

2.1.1 Base Model Specification

The features common to all the indel rate models are briefly described below. The models are discrete time Markov process models. Molecules exist in a population which has a fixed maximum size. The population is described by an integer population vector x which gives the number of molecules in the population of each length. Why $\boldsymbol{x}$ is integer is explained in section 2.1.1.3. Length is the only independent property of the molecules that is tracked. Mutation and replication rates are a function of a molecules length. Table: 2.3 shows the features of the molecules that we are looking at in the model.

Table 2.3: **A summary of the important features of the indel rate models.** The feature column gives the names of parts of the model. The representation column gives a brief description of how each of these features is represented in the model. The names of any mathematical parameters associated with each feature are given. These parameters are used (when relevant) throughout this chapter.

Feature	Representation	Parameter
Population of molecules.	Vector giving number of molecules of each length.	$\boldsymbol{x} = [x_l]$
Sequence	Not Represented in the model.	
Time	One unit of time is one cycle of the model.	t (cycles passed)
Molecule of length l.	Contributes one to the count of molecules of length l in population vector $\boldsymbol{x}$.	Only modeled in aggregate.
Number of distinct lengths molecules can assume	Number of components in the population vector.	$N = l_{\max} - l_{\min} + 1$
Replication rate	Used to determine the number of offspring produced by a molecule of length l in nonuniform replication models. A function of l. See section 2.1.1.2 for more details.	$R(l), l_{\min} \leq l \leq l_{\max}$
Indel rate	The probability of a molecule of length l having an indel. This is a function of l.	$F(l), l_{\min} \leq l \leq l_{\max}$
Asymmetry Parameter	Determines the relative probability of insertions and deletions. Has value $\varepsilon = 0.5$ in symmetric models	$\varepsilon \in [0, 1]$

Let $l_{\min}$ be the lower bound on molecule length and $l_{\max} = l_{\min} + N - 1$ the upper bound. In the simulations and calculations presented here $N = 101$ and $l_{\min} = 10$ therefore

$l_{\max} = 110$. Let $l \in (l_{\min} \ldots l_{\max})$ index the length of molecules in the model. Using the parameterization presented in table 2.3. $\boldsymbol{x} = [x_{l_{\min}}, \ldots, x_{l_{\max}}]$ where x_l is the number of molecules of length l. The process of the model is described below in section 2.1.1.1.

2.1.1.1 The Model Process The population can be regarded as occupying two buckets: An old population bucket, in which all the molecules present at the start of a cycle are located, and a new population bucket, in which any offspring of molecules in the old population bucket are kept. At the start of a cycle the new population bucket is empty and the old population bucket is full, containing approximately 10 000 molecules. Why this number is approximate and not exact is addressed in section 2.1.1.3.
In one cycle every molecule in the old population bucket will undergo the replication process ,described in section 2.1.1.2, in which each molecule produces one or possibly two offspring. The offspring of each molecule will be placed in the new population bucket. By assuming all molecules have at least one offspring we avoid the problems of population extinction and the undesirable effects of scaling a very small population.
Once every molecule in the old population bucket has undergone the replication process the old population bucket is emptied leaving us with only those molecules in the new population bucket. Each molecule in the old bucket produced at least one offspring. Therefore the number of molecules in the new bucket will be greater than or equal to the number in the old bucket at the start of this cycle. To maintain a constant population the number of molecules in the new population bucket is now scaled down to approximately 10 000 molecules. The scaling process is described in section 2.1.1.3. The new population bucket is now emptied into the old population bucket in preparation for a new cycle. This completes the cycle.

2.1.1.2 The Replication Process The replication process will result in one or more copies of a molecule being produced. The exact number of molecules produced is determined by the replication parameter r of the molecule in question. One molecule is produced for certain. A second molecule is produced with a probability $r-1$. For example if we take $r = 1.5$ one molecule will be produced for certain and there is a probability of 0.5 that a further molecule will be produced by the replication process. This notation is

used to allow easy meshing of the simulations and Markov models. Replication parameters used in these models are in the interval $1 \leq r \leq 2$. In uniform replication models $r = 1$ for all molecules. In nonuniform replication models r is a function of molecule length l. Each molecule produced by the replication process is tested independently to see if it has an indel. The probability of an indel is dependent on the molecules length l and is determined by the function $F(l)$ in a model specific way. If an indel occurs it will be of length one so the offspring will be one base shorter or longer than the parent. The relative probabilities of the offspring becoming shorter or longer depend on the symmetry properties of the model. Molecules are constrained to prevent mutating to larger than the maximum length or shorter than the minimum length. Constraining the mutation probabilities in this way can result in boundary effects. At both the lower and upper boundaries distributions of molecules can sometimes 'pile up' as further shortening or lengthening movement of the distribution is curtailed by the boundary conditions. The fact that distributions sometimes pile up in this manner does not alter any conclusions made in this section about the tendency of a distribution (or its parts) to move towards being either longer or shorter. For example a distribution piling up on the upper length boundary is clearly a distribution that favours longer molecules.

2.1.1.3 The Scaling Process At the end of each cycle the population of molecules in the new population bucket is scaled to be approximately 10 000. Let $\boldsymbol{x}_{\text{new bucket}}$ be the population vector for the molecules in the new bucket. Then $|x_{\text{new bucket}}|_1$ is the total number of molecules in the new bucket ($|\boldsymbol{v}|_1$ is the one norm of vector $\boldsymbol{v}$, $|\boldsymbol{v}|_1 = \sum_i |v_i|$. The scaled population vector $\boldsymbol{x}$ is given by equation (2.1).

$$\boldsymbol{x} = \text{floor}\left(10000 \cdot \frac{\boldsymbol{x}_{\text{new bucket}}}{|\boldsymbol{x}_{\text{new bucket}}|_1} \right) \tag{2.1}$$

Applying the floor() function to the new population vector $\boldsymbol{x}$ rounds non integer values down to the nearest integer. A consequence of this rounding is that $|\boldsymbol{x}|_1$, the total number of molecules in $\boldsymbol{x}$, is described by the relation $10000 - N \leq |\boldsymbol{x}|_1 \leq 10000$. Integers with rounding down is used to maintain the possibility of the number of molecules of a given length dropping to zero. Were rounding up used then once a molecule of a given length

has existed in the population there must always be at least one molecule of that length. It is more biologically realistic to allow the number of molecules of a given length to drop to zero (the 'extinction' of molecules of that length). Were real values used in x_i components would become arbitrarily small rather than becoming zero. Molecules are distinct entities and not infinitely divisible so this is again biologically unrealistic.

2.1.2 The Ur-Si model.

Before we look at the effects of different replication rates in the Ur-Si model model it is informative to look at what is happening when we consider only the effects of indel rate in isolation from other factors. This is done in the Ur-Si model which is defined below. Let the probability of a molecule of length l having either an insertion or deletion when it is copied be $F(l)$. At this stage we will assume the chance of an insertion is the same as the chance of a deletion occurring. This gives a Markov transition matrix M as below.

$$M = \begin{bmatrix} 1-2F(l_{\min}) & F(l_{\min}+1) & 0 & \cdots & & & 0 \\ 2F(l_{\min}) & 1-2F(l_{\min}+1) & \ddots & \ddots & & & \\ 0 & F(l_{\min}+1) & \ddots & \ddots & & & \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \\ & & & \ddots & \ddots & F(l_{\max}-1) & 0 \\ & & & \ddots & \ddots & 1-2F(l_{\max}-1) & 2F(l_{\max}) \\ 0 & & & & 0 & F(l_{\max}-1) & 1-2F(l_{\max}) \end{bmatrix}$$

Each column of the matrix represents what happens to molecules of a given length. The transition probabilities for a molecule of length l under the Ur-Si model are summarised in table 2.4

Table 2.4: **Transition probabilities for a molecule of length l under the Ur-Si model.** Mutation probabilities on the boundaries are defined so new molecules remain within the allowed range of lengths.

Event	**Length of molecule**	**Probability**
No Mutation	$l \in \{l_{\min}, \ldots, l_{\max}\}$	$1 - 2F(l)$
Insertion	$l = l_{\min}$	$2F(l)$
$\cdots$	$l \in \{l_{\min} + 1, \ldots, l_{\max} - 1\}$	$F(l)$
$\cdots$	$l = l_{\max}$	0
Deletion	$l = l_{\min}$	0
$\cdots$	$l \in \{l_{\min} + 1, \ldots, l_{\max} - 1\}$	$F(l)$
$\cdots$	$l = l_{\max}$	$2F(l)$

We are interested in the long term behavior of this model which means we want to look at Π, the dominant eigenvector of M (which has eigenvalue 1). This can be found by solving the equation

$$M\Pi = \Pi \tag{2.2}$$

Because M is tridiagonal and the probability of an insertion equals that of a deletion (except at the boundary) we can solve equation (2.2) in terms of $\pi_{l_{\min}}$ by back substitution. We get the following results

$$\pi_l = 2\pi_{l_{\min}} \frac{F(l_{\min})}{F(l)} \tag{2.3}$$

where $l = l_{\min} + 1, \ldots, l_{\max} - 1$.

$$\pi_{l_{\max}} = \pi_{l_{\min}} \frac{F(l_{\min})}{F(l_{\max})} \tag{2.4}$$

We normalize this vector so its components sum to one by dividing by the sum of its components ($|\Pi|_1$). Thus using equations (2.3) and (2.4) we can calculate what the final

distribution of molecules will look like for any given indel rate function F.
Equation (2.5) presents the function F that was used to produce the figures in this section (2.1.2).

$$F(l) = \frac{1}{l} \tag{2.5}$$

This instance of the function F was chosen because it a simple function in which the number of indels is inversely proportional to length. therefore the larger the molecule the fewer indels it will have. The analytically determined steady state distribution in this situation is shown in figure 2.1.

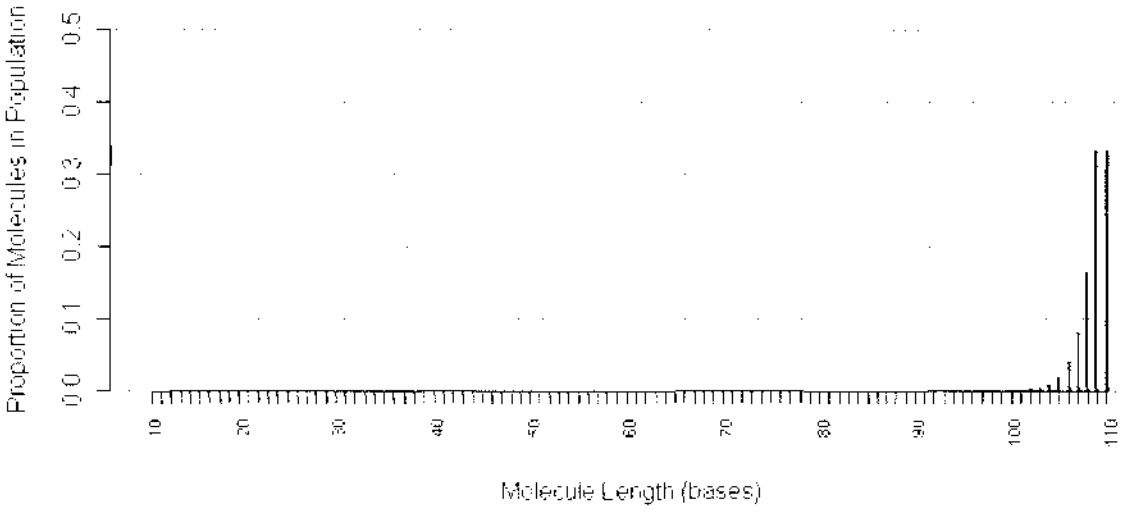


Figure 2.1: **Steady State distribution when long molecules have a smaller chance of indels in the Ur-Si model.** Here we are looking at the proportion of molecules of each length predicted in the long term for a population of molecules where long molecules have a smaller chance of indels than short molecules. Replication rate is uniform across the population.

Figure 2.1 shows that equation (2.5) results in a distribution strongly favoring longer molecules. Thus the effects of indel rate can favor long molecules. Furthermore equation

(2.3) suggests that whichever final distribution we desire can be achieved by appropriate choice of F as the final distribution is entirely determined by F. Before we consider possible justifications for any particular choice of F let us look at what happens when we vary the replication rate.

2.1.3 Nonuniform Replication: The Nr-Si model.

Nonuniform replication is accounted for by multiplying the matrix M by the diagonal replication rate matrix R to obtain the Nr-Si model. The entries in R represent the relative rates of replication of molecules of each length. This definition of the replication rate matrix gives a Markov transition matrix of the form RM for a system with nonuniform replication.

Let λ be the dominant eigenvalue of the system. To find the final state of this new system we must solve the equation

$$RM\Pi = \lambda\Pi \tag{2.6}$$

In equation (2.6) λ is a root of a Nth degree polynomial so we can no longer easily find a general analytical solution to the system. However we can use numerical methods to solve the system for a given R and M.

Let the lth entry of R be given by the function $r(l)$. We will make the effects of replication small and linear and define r as in equation (2.7).

$$r(l) = \frac{l_{\max} - l}{N} + 1 \tag{2.7}$$

For the given values of $l_{\min}$ and $l_{\max}$we find the ratio of replication rate of the longest molecule to that of the shortest molecule is approximately $\frac{1}{2}$, while the ratio of their lengths is approximately 11. If we solve this system we find that the steady state distribution is the one depicted in figure 2.2.

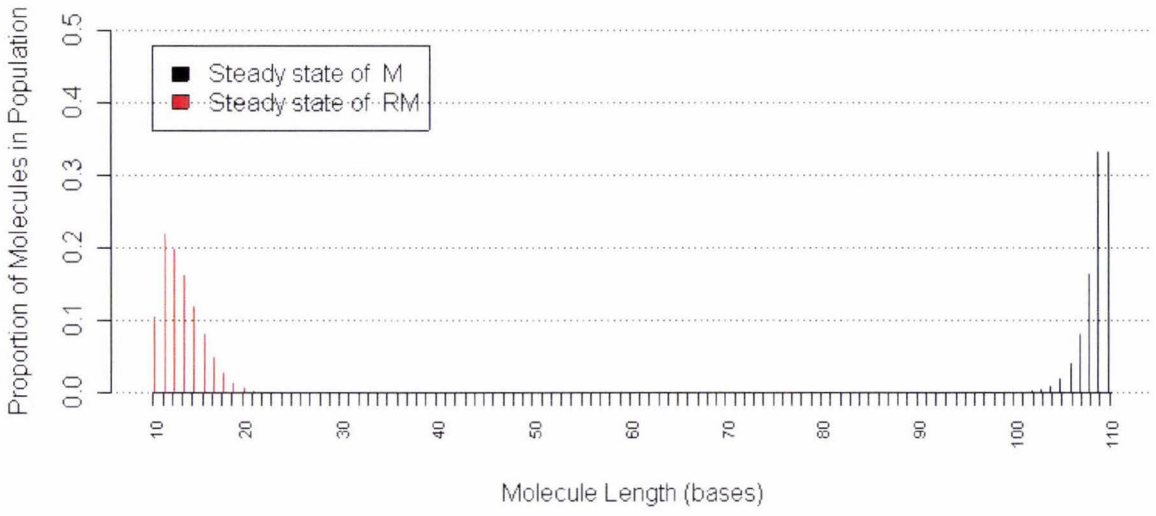


Figure 2.2: **Steady state of the the Nr-Si model with nonuniform replication compared to the Ur-Si model.** We are looking at two distinct populations here. A population with uniform replication rate (**black**) and one in which short molecules replicate faster (**red**) for R as is defined in (2.7). In both populations long molecules have fewer indels than short molecules. The indel rate function F is defined in (2.5)

We can see that the effects of replication are dominating those of indel rate. This effect might arise simply because we are giving a relatively low weighting to the effects of indel rate. Thus let us look at what happens in a model influenced by a factor more dominant than indel rate.

2.1.4 The Ur-Ai model.

A model with a more dominant factor than indel rate is one in which the probabilities of an insertion and a deletion differ. In our previous model for molecules of a length l we had the situation depicted in table 2.5 (a copy of table 2.4).

Table 2.5: **Transition probabilities for a molecule of length l under the Ur-Si model.**

Event	Length of molecule	Probability
No Mutation	$l \in \{l_{\min}, \ldots, l_{\max}\}$	$1 - 2F(l)$
Insertion	$l = l_{\min}$	$2F(l)$
$\cdots$	$l \in \{l_{\min} + 1, \ldots, l_{\max} - 1\}$	$F(l)$
$\cdots$	$l = l_{\max}$	0
Deletion	$l = l_{\min}$	0
$\cdots$	$l \in \{l_{\min} + 1, \ldots, l_{\max} - 1\}$	$F(l)$
$\cdots$	$l = l_{\max}$	$2F(l)$

In the asymmetric models we introduce the asymmetry parameter ε which ranges between 0 and 1 and determines the degree of asymmetry between insertions and deletions. The probabilities in this new model are described in table 2.6.

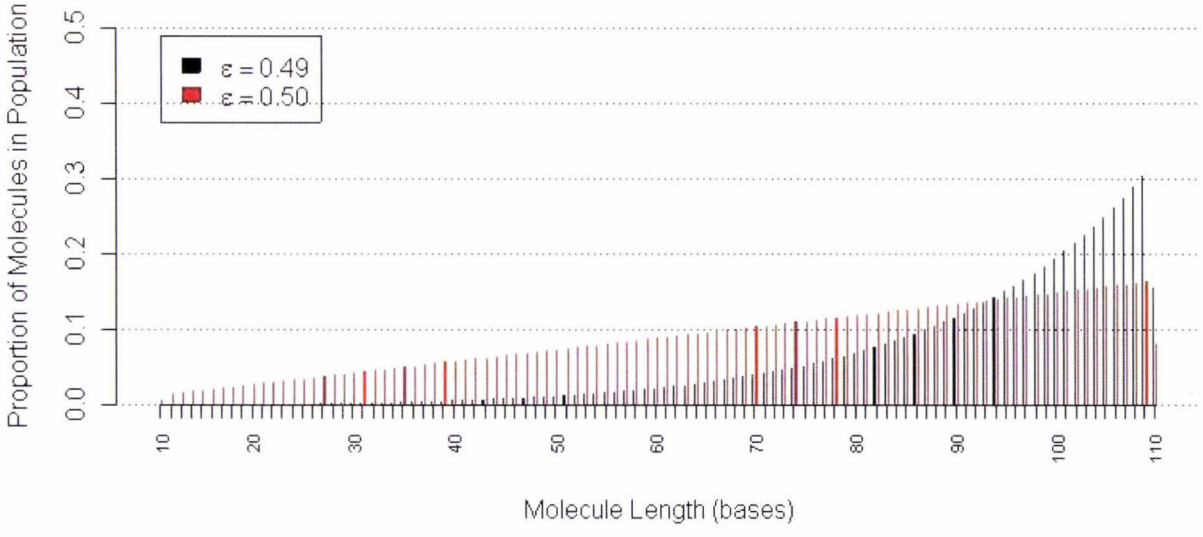
Table 2.6: **Transition probabilities for a molecule of length l under the Ur-Ai model with parameter ε.**

Event	Length of molecule	Probability
No Mutation	$l \in \{l_{\min}, \ldots, l_{\max}\}$	$1 - 2F(l)$
Insertion	$l = l_{\min}$	$2F(l)$
$\cdots$	$l \in \{l_{\min} + 1, \ldots, l_{\max} - 1\}$	$(1 - \varepsilon)2F(l)$
$\cdots$	$l = l_{\max}$	0
Deletion	$l = l_{\min}$	0
$\cdots$	$l \in \{l_{\min} + 1, \ldots, l_{\max} - 1\}$	$\varepsilon 2F(l)$
$\cdots$	$l = l_{\max}$	$2F(l)$

Thus $\varepsilon = 0.5$ gives the Ur-Si model.The Ur-Ai model gives a new Markov transition matrix Q shown below.

$$\begin{bmatrix} 1-2F(l_{\min}) & \varepsilon 2F(l_{\min}+1) & 0 & \cdots & & \cdots & 0 \\ 2F(l_{\min}) & 1-2F(l_{\min}+1) & \ddots & & & & \vdots \\ 0 & (1-\varepsilon)2F(l_{\min}+1) & \ddots & \ddots & & & \\ \vdots & & \ddots & \ddots & \ddots & & \\ & & & \ddots & \ddots & \varepsilon 2F(l_{\max}-1) & 0 \\ \vdots & & & & \ddots & 1-2F(l_{\max}-1) & 2F(l_{\max}) \\ 0 & \cdots & & \cdots & 0 & (1-\varepsilon)\,2F(l_{\max}-1) & 1-2F(l_{\max}) \end{bmatrix} \tag{2.8}$$

Let $\Pi = [\pi_l]_{l=l_{\min},l_{\max}}$ be the dominant eigenvector of Q. Due to the complexity of solving analytically for an expression of Π in terms of π_0, ε and F we take a different approach. To see how Π changes with varying ε we solve the model numerically for multiple values of ε. In figure: 2.3 we see the results for $\varepsilon = 0.49, 0.5, \ldots, 0.54$ using F as defined in equation (2.5).



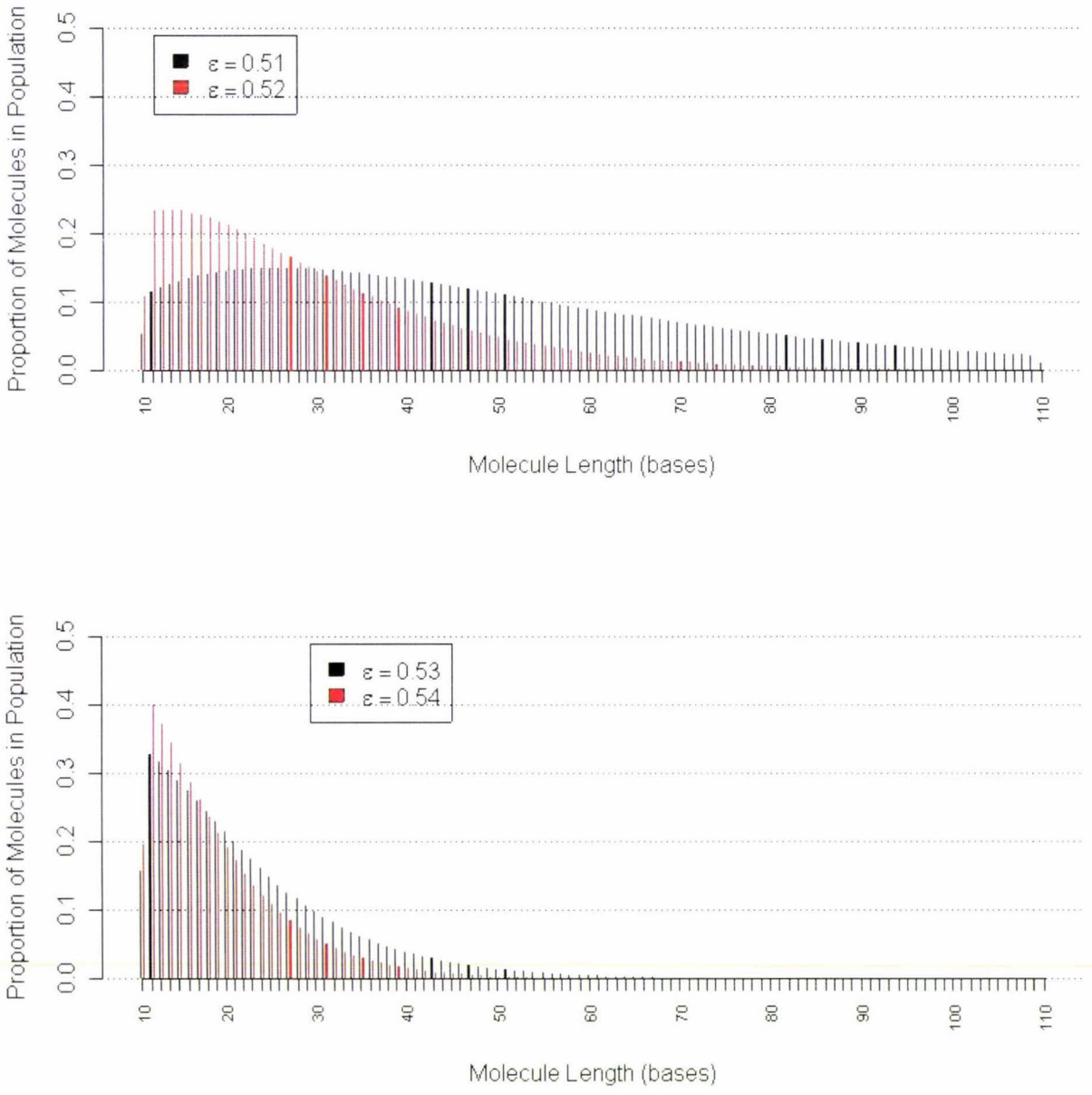


Figure 2.3: **The effect of ε on the steady state.** Each plot shows two different values of ε in a progression from $\varepsilon = 0.49$ to $\varepsilon = 0.54$. Each trace in the plots represents a distinct population. In each of these populations long molecules have fewer indels than short molecules; only the parameter ε is altered.

From these results we can see that asymmetry has a strong effect on the model. A change in the asymmetry parameter ε from 0.50 to 0.51 results in the distribution changing from

being weighted in favour of long molecules to one favouring short molecules. This shows that even small asymmetries in occurrence between insertions and deletions can dominate the effects of indel rate. In our next model we test whether the effects of asymmetry can lead to a balanced distribution of molecules in a model with nonuniform replication.

2.1.5 The Nr-Ai model.

To account for replication effects we must, as before, multiply the transition matrix Q by the replication rate matrix R (where R is defined in section 2.1.3). This gives us the Nr-Ai model. Let λ be the dominant eigenvalue of the system and Π the dominant eigenvector. We must then solve...

$$RQ\Pi = \lambda\Pi \tag{2.9}$$

As λ is a the root of an Nth degree polynomial we cannot easily solve the system using analytical methods. To get an idea of how replication affects the model we look at the extreme case of $z = 0.01$ to see what happens in a worse case scenario (from the perspective of short molecules). Results are displayed in figure: 2.4.

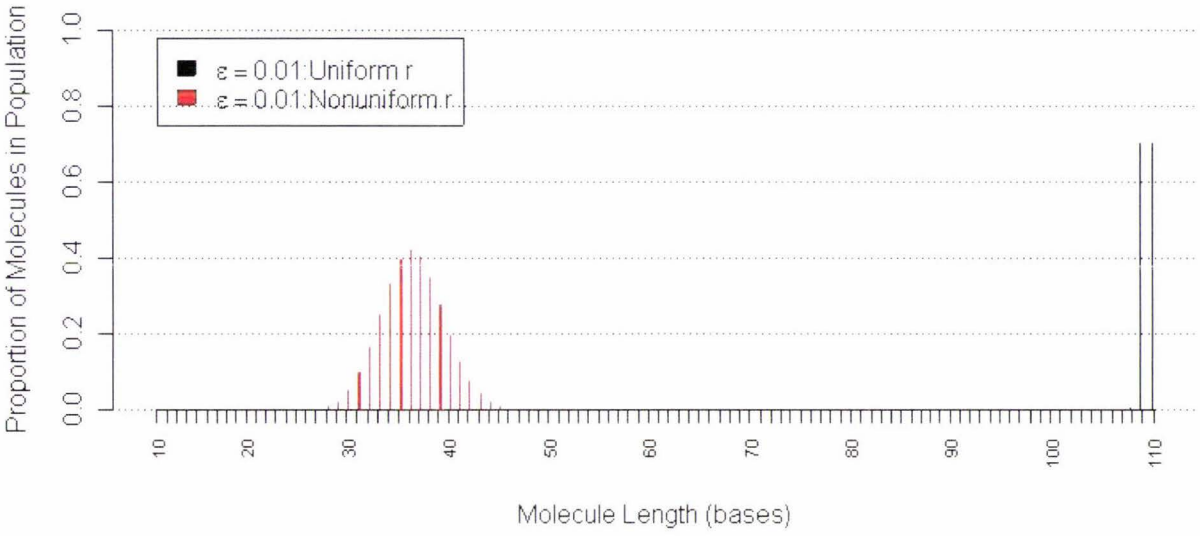


Figure 2.4: **Steady state of a highly asymmetric system under the Nr-Ai model.** Each trace represents a distinct population. Each population is such that long molecules have fewer indels than short and the probability of a molecule having an insertion when it mutates is 98% ($\varepsilon = 0.01$). The **black** bars shows a population with uniform replication rate. The red bars shows a population where shorter molecules replicate more quickly.

In this example 98% of mutations result in a longer molecule ($\varepsilon = 0.01$). Although these extreme parameters produce a distribution of molecule lengths, it is biased towards short molecules and not maintained under more biologically reasonable (though still unlikely) parameter values such as $\varepsilon = 0.2$ (data not shown). We can see the effects of replication strongly dominate those of asymmetry.

2.1.6 Conclusion: Indel Rate Models.

We have seen that regardless of effects of indel rate and indel asymmetry, the fastest replicating (shorter) molecules dominate in these models (section 2.1 and subsections). More extensive studies undertaken as part of this project using both analytic and simulation methods have shown that increasing the allowable length of indels does not qualitatively effect the nature of this result. This leads us to the conclusion that these models are inadequate to provide a balanced distribution of molecules in the face of uneven replication rates.We turn next to game theory.

2.2 Game Theory

We are looking for a simple model with a population of mixed length molecules as a steady state. In the indel rate models we ignored the effects of catalysis on replication by assuming that all molecules were equally well catalysed. In this model we modify that assumption. Now we assume that each molecule is catalysed by one other randomly chosen molecule in the population each cycle. The degree of catalysis depends on the lengths of the two molecules participating in the reaction. To consider these catalytic effects we will turn to an evolutionary game theory model. Evolutionary game theory is often used in situations where the frequency of a genotype affects its fitness. In our case the phenotype we are interested in is length. Before we go into the details of the model we are interested in let us take a general look at evolutionary game theory and what it is.

2.2.1 Evolutionary Game theory.

Evolutionary game theory is an offshoot of classical game theory. In classical game theory we are modeling two or more players competing for advantage. Each player has a predefined set of strategies they can choose from to try and obtain an outcome advantageous to themselves. The game matrix defines what reward (/penalty) each player receives given both their own and their opponent's strategy. A classic example of this is the prisoner's dilemma game (Osborne 1994), an example of which is shown below. Two prisoners are kept in separate cells. The police can convict both on robbery but know one is guilty of aggravated robbery. If a prisoner is willing to testify against his comrade he will get a reduced sentence. However if both testify against each other the sentence for aggravated robbery will be split between them.

Table 2.7: **The Payoff matrix for prisoners P1 and P2 from the point of view of P1 in the prisoners dilemma game.** P1's strategies are to the left. P2's strategies are on the top.The payoffs represent how long P1 would have to spend in jail. P2 has the same payoffs as given for P1 if the roles of P1 and P2 are reversed.

	P2 Testify	P2 Remain Silent
P1 Testify	5 years	1 year
P1 Remain Silent	10 years	2 years

This is a dilemma because the most 'rational' strategy for each player against the other is to testify which ends up getting them more overall jail time then if they had both stayed silent.
In evolutionary game theory we are again modeling players competing for their own best advantage. There are three main differences however.

- Players are randomly drawn from the population being modeled.
- Strategies correspond to the phenotypes available to members of the population.
- The payoff to players represents an increase in Darwinian fitness.

The long term behavior of an evolutionary game is described by the strategy or mixture of strategies that lead to an evolutionary stable state (evolutionary stable strategies or ESS) (Maynard-Smith 1982).

Definition 2.1. *ESS: An ESS is a strategy such that if a population is using it then that population is not vulnerable to invasion by mutants using any other strategy.*

An ESS can be a pure strategy or a mixture of strategies. Mixtures of strategies can be defined in several different ways. We will consider the two main ways presented in

Maynard-Smith (1982) (names of these types are taken from Maynard-Smith for consistency). The examples below are given in the context of a 2 strategy population: However they can be extended to an n strategy population. The two types of mixing are:

1. Individuals within a population use strategy 1 some of the time and strategy 2 the remainder of the time. This is most useful when the strategies represent malleable/continuous phenotypes such as behavioral responses. This is referred to as a mixed strategy.

2. Individuals within a population consistently use one strategy. Some individuals use strategy 1 while the remaining individuals use strategy 2. This is more suitable for non-malleable/discrete phenotypes such as length or colour. This is referred to as a polymorphic strategy.

We will only consider the second type of mixed strategy, polymorphic strategies, as a molecule is only of one length at any given time.

When an evolutionary game is played, individual protagonists are picked at random from the population and fitness benefits are assigned according to the phenotypes (strategies) of the two protagonists. Once each individual has played the game with a random partner, the population has a reproduction phase where individuals have offspring of their own type in proportion to their fitness. A population evolving under the rules of a game will evolve towards an evolutionary stable strategy (Maynard-Smith 1982). It is possible a game will have more than one ESS. In this case the final state will be mostly dependent on the initial conditions. The set of initial conditions that will evolve towards a given ESS can be thought of as a catchment area (Bergstrom and Lachmann 2003) for that ESS. A simple example is given in figure: 2.5.

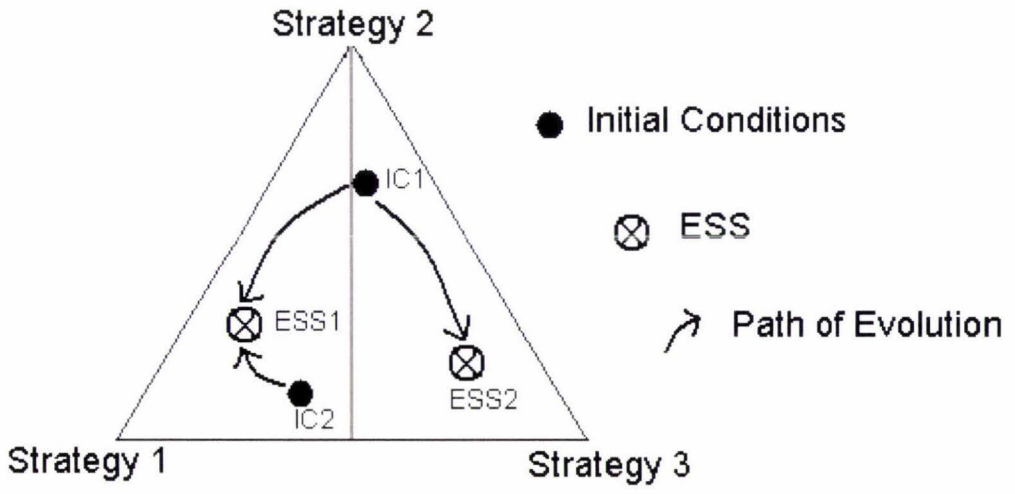


Figure 2.5: **Example of Multiple ESS in a 2 player 3 strategy game.** The mixture of strategies in the population is represented by the position of the population in the simplex. The grey line is the separating line between the catchments of ESS 1 and 2. IC1, the first set of initial conditions, is near the catchment boundary so it could evolve towards either ESS. IC2 is firmly within the catchment of ESS1 so it will evolve towards that ESS.

This still leaves the question of what can we usefully ask about an evolutionary game theory model? In general we can ask what the ESS(s) of the model are. If there is more than one ESS then we can ask which initial conditions lead to which ESS.

2.2.2 The Evolutionary Game Model: The Simple Model.

Now we have an understanding of evolutionary games let us look at how we can use them. First we define our model.

Molecules live in a well mixed medium. In each cycle every molecule is given a chance to replicate. The molecule being replicated in a reaction is called the template. A catalyst for the reaction is randomly picked from the population. The number of copies of the template created is a function of the reaction's rate of replication. A reaction's rate of replication depends on the efficacy of the catalyst which in turn depends on the length of

the template in relation to the catalyst. We will refer to this model as the "full model". We will start with a very simple model which we will call the "simple model". This model is treated in Chapter 2 of Maynard-Smiths book "Evolution and the theory of games" (1982). This model will later be expanded to the full model (as detailed above).
The assumptions we will make for the simple model are as follows:

- Molecules can come in two lengths, long (L) and short (S).
- The population of molecules is infinitely large.
- Reproduction is asexual
- The game is symmetric. This means that the two players of the game should, except in regards to their strategy choice, be indistinguishable from each other when the game begins. Thus all properties of a molecule affecting the game (such as being a catalyst or being a template) must be shared by all molecules in the population.

In assuming the game is symmetric we are assuming that if a molecule catalyses another, then its own replication is catalysed by the same molecule in turn. This is not consistent with there being a random catalyst for each molecule. However symmetry is a useful property for the game to have for an initial analysis so the assumption is used. The payoff matrix for the simple game will be as in table: 2.8

Table 2.8: **Game Payoffs for the template molecule in the simple game.** The entries in this table represent the fitness payoff to the template molecule of being catalysed by the catalyst. For example SL is the payoff to a short molecule with a long catalyst. The game is symmetric so if the roles of the template and catalyst are swapped the new template would have the same payoff matrix.

	Catalyst: Short	Catalyst: Long
Template: Short	SS	SL
Template: Long	LS	LL

The matrix entries represent the replication rate of the template. It is assumed there is a direct correlation between replication rate and fitness. The Nr-Si and Nr-Ai models (as defined in section 2.1.3 and 2.1.5) as well as the results of Spiegelman *et al.* (1965) and Biebricher *et al.* (1981) give us reason to believe this. We are looking for a population of mixed length molecules as an ESS. This means that we are looking for a game with a polymorphic ESS.

Definition 2.2. *Conditions sufficient for a polymorphic strategy*
Maynard-Smith(1982) tells us that there is a polymorphic ESS for the game with matrix defined by the values in table 2.8 provided SS < LS *and* LL < SL.

An explanation for SS < LS is a long molecule is likely to be a better catalyst for a short molecule than another short molecule as it will have more possible catalytic sites. We also need LL < SL. This implies short molecules catalyse long molecules better than other long molecules. A possible explanation for this is steric hindrance (the parts of the long molecules not active in the catalysis interfering with each other).
It should be noted that a system where SS > LS and LL > SL will also have a polymorphic ESS. This ESS will be different to the ESS of system fulfilling Definition 2.2. The conditions in Definition 2.2 were used as they seem easier to explain biologically than the alternative.
Providing the conditions in Definition 2.2 are fulfilled, Maynard-Smith (1982) tells us that the polymorphic ESS of this game will be given by a proportion n of the population being short where n is given in equation (2.10).

$$n = \frac{\mathrm{SL} - \mathrm{LL}}{\mathrm{SL} + \mathrm{LS} - \mathrm{SS} - \mathrm{LL}} \tag{2.10}$$

In Example 2.3 we apply (2.10) to an instance of the simple game.

Example 2.3.

Suppose that we have a similar situation to the indel rate models, where the fastest replicating molecules replicate twice as fast as the slowest replicating molecules. We pick the two intermediate speeds by linearly interpolating between 1 and 2. Values are assigned to

give a mixed strategy solution. This gives rise to a payoff matrix such as the one below.

Table 2.9: **Numerical values of payoff to template molecule in the simple game model.** For example if the template is short and the catalyst long then the payoff to the template is $\frac{5}{3}$.

	Catalyst:Short	Catalyst: Long
Template: Short	$1 = \mathrm{SS}$	$\frac{5}{3} = \mathrm{SL}$
Template: Long	$2 = \mathrm{LS}$	$\frac{1}{3} = \mathrm{LL}$

this gives

$$n = \frac{\frac{5}{3} - \frac{1}{3}}{\frac{5}{3} + 2 - 1 - \frac{1}{3}} = \frac{1}{4}$$

This implies that a evolutionary stable polymorphic population of this type will have $\frac{1}{4}$ *short molecules and* $\frac{3}{4}$ *long molecules. This is the only ESS of the model.*

2.2.3 Expanding the game: The Full Model.

The full model has several important differences from the simple model. Molecules can now be of any (integer) length from $l_{\min}$ to $l_{\max} = l_{\min} + N - 1$. The range of allowable molecule lengths is changed from short and long to $l_{\min} \ldots l_{\max}$ to allow comparison with the indel rate models (section 2.1). A wide range of lengths also allows the investigation of the new interactions created by having many lengths of molecule. The full model is too complex to analyse analytically. To allow investigation by numerical and simulation methods it is necessary to use a finite population. This is also biologically realistic (as real populations are finite) and it has been shown finite population effects can significantly alter the behavior of models (Nowak *et al.* 2004). Finally the assumption of symmetry is biologically very unrealistic so it is removed. The only reason the symmetry assumption was made in the simple model was for mathematical tractability. The full model is already

intractable due to its high dimensionality so tractability is no longer an issue. Under the full model the catalyst of a molecule is randomly selected and is independent of which (if any) molecules that molecule catalysed that cycle. This violates symmetry because molecules will be distinguishable as they will only perform one of the roles of template or catalyst in each game and not both as previously. In choosing to use a range of molecular lengths we are increasing the number of available phenotypes, which is the same as the number of strategies, in the game. This requires a definition of how the new phenotypes interact (a new game matrix).

To maintain some similarity to the simple model the four values defined in the simple game matrix in table 2.8 are used as the 4 corner values of the game matrix G. These values are given in table 2.10.

Table 2.10: **Names of payoffs for the simple game and the reaction type they correspond to.**

Value	**Meaning**
SS	(**S**mall Template):(**S**mall Catalyst)
SL	(**S**mall Template):(**L**arge Catalyst)
LS	(**L**arge Template):(**S**mall Catalyst)
LL	(**L**arge Template):(**L**arge Catalyst)

This gives an $N \times N$ matrix G of the form shown below in which a general entry $g_{l,j}$ is the payoff for a molecule of length l being catalysed by a molecule of length j. G has been transposed in relation to table 2.9 in Example 2.3 to make it appropriate for matrix multiplications in later steps.

$$G = \begin{bmatrix} \text{SS} & \cdots & \text{LS} \\ \vdots & \ddots & \vdots \\ \text{SL} & \cdots & \text{LL} \end{bmatrix} \tag{2.11}$$

When thinking about the internal values of G it helps to visualize them in terms of points on a surface over an $N \times N$ square whose height at any integer coordinate (l, j) is $g_{l,j}$. Only the four corner values of G have been set so far, all remaining entries are yet to be specified. To maintain a degree of simplicity we set the remaining entries using linear interpolation. There are several ways of linearly interpolating a surface between the four boundary points specified in (2.11). The two classes of method used here and the games that result are described in sections 2.2.5 and 2.2.6. All these methods result in games for which no analytic solution could be found. Simulations, detailed in section 2.2.4, were used to investigate the properties of these games.

2.2.4 Simulating the Full Model.

Simulations were run to investigate the behavior of the games presented in sections 2.2.5 and 2.2.6. It should be noted that simulations have been used to test the nature of each of the models presented so far. However as they have agreed completely with the analytical results it has not been necessary to refer to them until now. Unfortunately the full model is not conducive to analytical solution due to its high dimensionality. We will look at this problem in more detail in section 2.2.7. To circumvent this difficulty we use simulations to investigate the full model. The simulation process is very similar to the process of the indel rate models. The process consists of three parts: The main process, the replication process and the scaling process. The main and scaling processes are identical to the model and replication processes of the indel rate models which are described in detail in sections 2.1.1.1 and 2.1.1.3 and briefly summarized here.

In the main process molecules are put in old and new population buckets. The old population bucket contains all molecules present at the start of the cycle. All molecules in the old bucket are given a chance to replicate in the replication process, with any offspring going into the new bucket. At the end of the cycle the new bucket is kept and the old bucket thrown out. The new bucket is scaled in the scaling process to have approximately 10 000 molecules and tipped into the old bucket ready for a new cycle.

At the end of each cycle the population of molecules in the new population bucket is scaled to be approximately 10 000. Let $\boldsymbol{x}_{\text{new bucket}}$ be the population vector for the molecules in the new bucket. Then $|x_{\text{new bucket}}|_1$ is the total number of molecules in the new bucket. The scaled population vector $\boldsymbol{x}$ is given by equation (2.12).

$$\boldsymbol{x} = \text{floor}\left(10000 \cdot \frac{\boldsymbol{x}_{\text{new bucket}}}{|\boldsymbol{x}_{\text{new bucket}}|_1}\right) \tag{2.12}$$

Applying the floor() function to the new population vector $\boldsymbol{x}$ rounds non integer values down to the nearest integer. A consequence of this rounding is that $|\boldsymbol{x}|_1$, the total number of molecules in $\boldsymbol{x}$, is described by the relation $10000 - N \leq |\boldsymbol{x}|_1 \leq 10000$.

The replication process for game simulations is as follows. The main process iterates through every molecule sequentially. When the main process passes a molecule to the replication process, that molecule becomes the template in a playing of the game. Thus every molecule in the old bucket plays a game in the role of template exactly once in each cycle. A catalyst for the template is randomly selected from the population. Let the template be of length l and the catalyst of length j. The replication parameter r for the selected template molecule this cycle is given by the payoff $g_{l,j} \in G$ for the template-catalyst interaction. Each interaction generates at least one new molecule of the same length as the template. There is a probability of $r - 1$ that one further offspring will be produced. All offspring are the same length as the template: there is no mutation in the game models.

2.2.5 The Planar Games.

The game matrix G is defined in equation (2.11). For continuity we take the corner values of G from the values of SS, SL, LS and LL used in Example 2.3. This allows easy

comparison with previous models. This gives G the form of (2.13).

$$G = \begin{bmatrix} \mathrm{SS} & \cdots & \mathrm{LS} \\ \vdots & \ddots & \vdots \\ \mathrm{SL} & \cdots & \mathrm{LL} \end{bmatrix} = \begin{bmatrix} 1 & \cdots & 2 \\ \vdots & \ddots & \vdots \\ \frac{5}{3} & \cdots & \frac{4}{3} \end{bmatrix} \tag{2.13}$$

The internal entries of G are yet to be determined. In the following step we visualize the internal values of G in terms of points on a surface over an $N \times N$ square whose height at any integer coordinate (i, j) is $g_{i,j}$. To construct the internal values of G we construct a surface using a piecewise linear representation containing as few planes as possible which together pass through all four corner points (this is to try and get as simple a surface as possible). Using the corner points defined in (2.13) we see not all the corner points lie on the same plane so two planes must be used for the construction. During this construction we must make an arbitrary choice about how the planes we use intersect. We have two options which are depicted in figure 2.6.

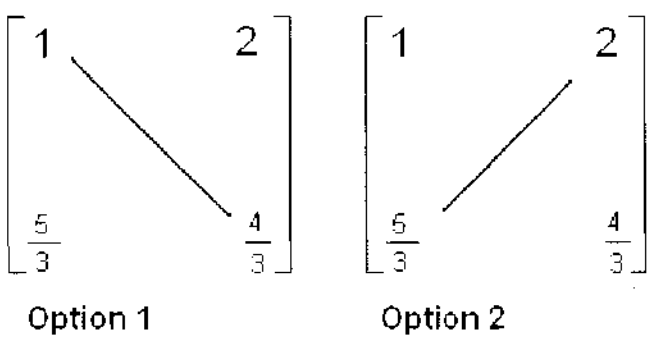


Figure 2.6: **Intersection Types: Option 1:** The intersection of the two planes is on a line between $g_{i_{\min},i_{\min}}$ and $g_{i_{\max},i_{\max}}$, this surface is called a concave surface.**Option 2:** The intersection of the two planes is on a line between $g_{i_{\min},i_{\max}}$ and $g_{i_{\max},i_{\min}}$, this surface is called a convex surface.

These two choices for the line of intersection give the two surfaces corresponding to two distinct game matrices. These surfaces are shown in figure: 2.7. Note that both surfaces have the same corner points.

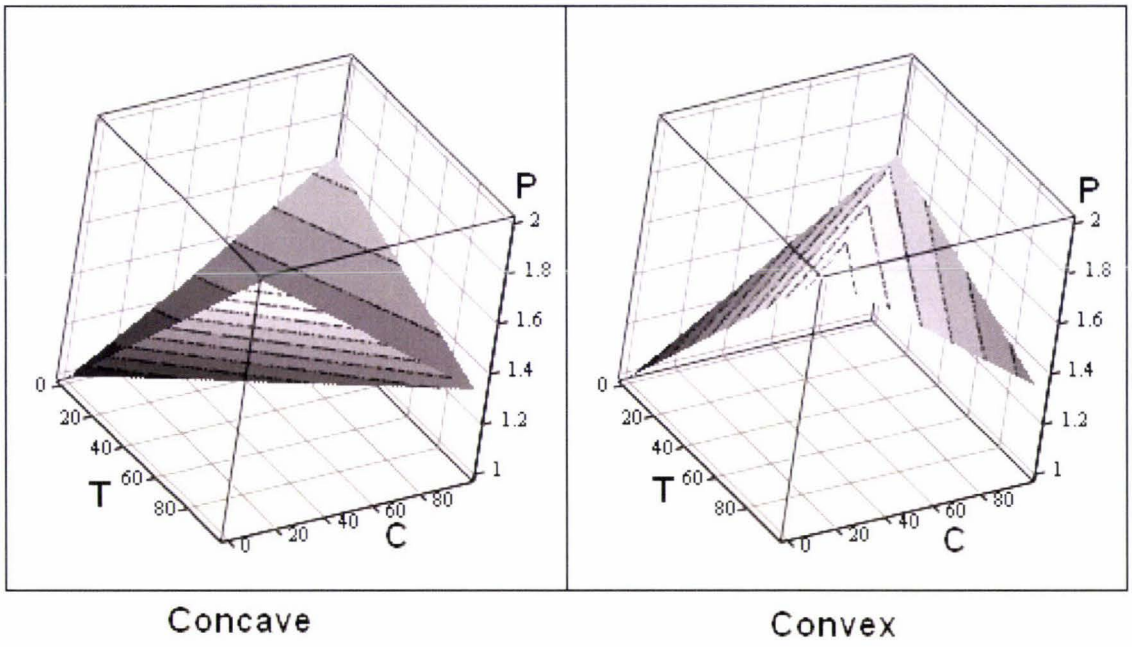


Figure 2.7: **Graphs showing the surfaces generated by the two subtypes of the game matrix** G. In each graph the T axis gives the length of the template (minus $l_{\min}$). The C axis gives the length of the catalyst (minus $l_{\min}$). Finally the P axis gives the size of the payoff. The concave matrix is generated when the two planes have corner $g_{l_{\min},l_{\min}}$ and $g_{l_{\max},l_{\max}}$ in common. The convex matrix is generated when the two planes have corner points $g_{l_{\min},l_{\max}}$ and $g_{l_{\max},l_{\min}}$ in common.

We refer to these games as the planar games collectively and as the convex game (with matrix G_{convex}) and concave (with matrix G_{concave}) games separately.
Simulations were run to investigate the behavior of the planar games. Typical results for each of the subtypes are presented below in figure 2.8 and figure 2.9.

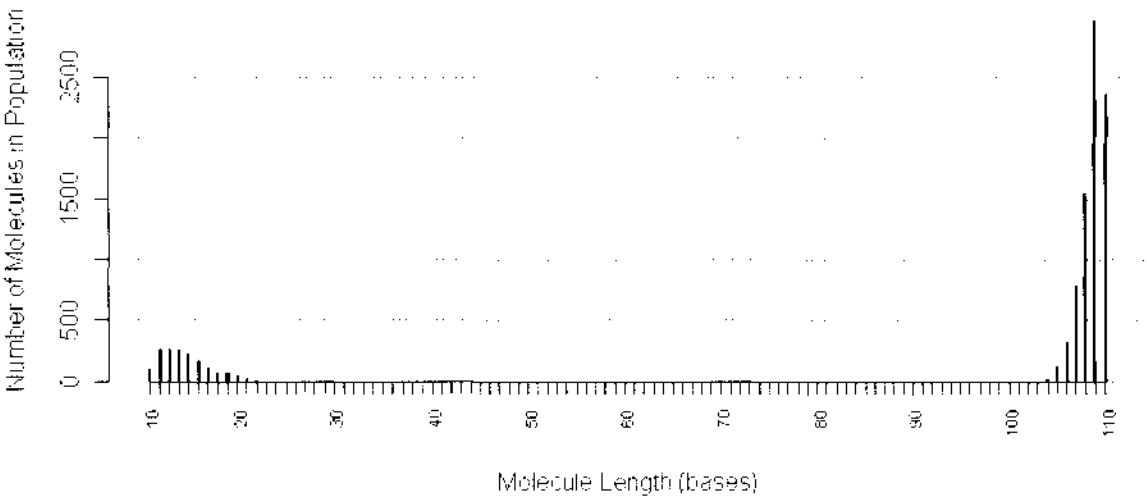


Figure 2.8: **Simulation results after 10^4 cycles using the concave game matrix starting from a uniform distribution.** Bars represent the proportion of molecules of each length in the population.

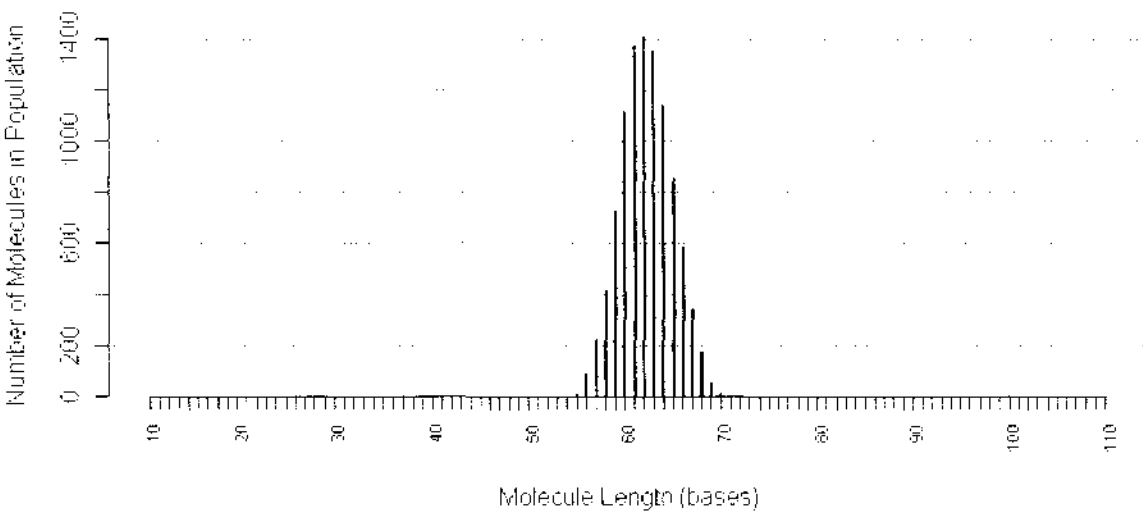


Figure 2.9: **Simulation results after 10^4 cycles using the convex game matrix starting from a uniform distribution.** Bars represent the proportion of molecules of each length in the population.

The above figures display the simulation results after 10 000 cycles of the simulation. These distributions were observed to be stable when running for a further 90 000 cycles and thus we can be fairly certain they are the steady state distributions. Each of the

simulations was started with a uniform distribution of molecules. It is interesting to note that both types of planar game result in a final distribution containing several different lengths of molecules. It is also interesting that the mean of these final distributions is quite different from that of the simple model (defined in section 2.2.2). In the concave case there are many more long molecules than short as predicted in the simple model. In both models we see that slowly and quickly replicating molecules can coexist.

The effect of initial conditions on the final outcome of the model differs between the concave and convex models. In the concave model the final distribution is qualitatively the same except when the initial population are all longer than approximately 100 bases when the short mode of the distribution does not form. In the concave model the initial conditions do not seem to effect the long term outcome.

There is no basis for choosing either of the subtypes of the planar models as better representing what we are trying to model. Therefore that they differ qualitatively is interesting. Thus we look for a model that combines the features of both planar models.

2.2.6 The Bilinear Game

The two planar game matrix surfaces are distinguished from each other by an arbitrary choice of the line of intersection of the two planes used during the construction of each of the two surfaces. We can remove this arbitrary choice by making the surface generated by a new game matrix G_{Bilinear} a bilinear interpolation between the four corner points. A bilinear interpolation is defined in Definition 2.4.

Definition 2.4. *Bilinear Interpolation*

A bilinear interpolation defines a height field over a plane. Figure 2.10 illustrates how a bilinear interpolation works and we will refer to it in the following discussion. We start with the heights of the four points on the corners of the area we are interpolating over (the grey square in figure 2.10) as given. The red spheres in part A of the figure represent these 4 corner points. We will call lines joining adjacent corner points boundary lines. Boundary lines are calculated by linear interpolation between adjacent corner points (part B of the figure).

We look at the calculation of an arbitrary internal point to demonstrate how these are calculated. Let the point p we are finding the height for have x-y coordinates (x, y). Then the height h of the point can be found by linear interpolation between the two points which lie on boundary lines and have the same x-coordinate as p (part C of the figure). The two points on lying on boundary lines which have the same y-coordinate as p can be also be used with identical results. A mathematical description of this is given below.

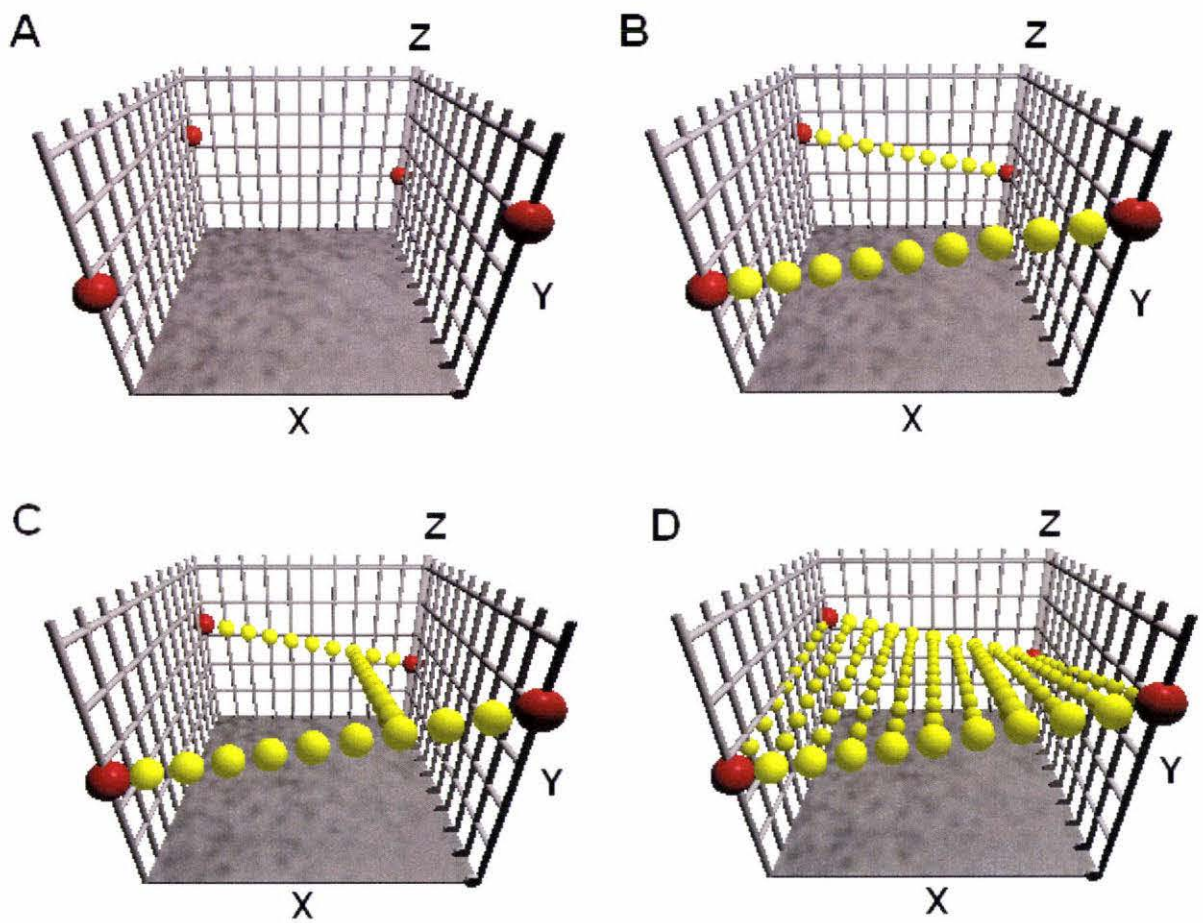


Figure 2.10: **Four steps of a bilinear interpolation.** An example using an 11 by 11 matrix. **A)** The four corner points. **B)** Boundary lines defined by linear interpolation between corner points. **C)** Internal points defined by linear interpolation between boundary lines. **D)** All values determined.

Let $f(\boldsymbol{v}) : \mathbb{R}^2 \to \mathbb{R}$ be a function that defines the height of a surface S over the domain

$0 \leq x \leq N, 0 \leq y \leq N, N \in \mathbb{R}$. *Let the corner points be defined such that*

$$f\left(\begin{bmatrix}0\\0\end{bmatrix}\right) = a \in \mathbb{R}$$

$$f\left(\begin{bmatrix}0\\N\end{bmatrix}\right) = b \in \mathbb{R}$$

$$f\left(\begin{bmatrix}N\\N\end{bmatrix}\right) = c \in \mathbb{R}$$

$$f\left(\begin{bmatrix}N\\0\end{bmatrix}\right) = d \in \mathbb{R}$$

Define $h(\boldsymbol{u}, \boldsymbol{v}, p) = \frac{f(\boldsymbol{v}) - f(\boldsymbol{u})}{|\boldsymbol{u} - \boldsymbol{v}|_2} p + f(\boldsymbol{u}), u \neq v$ *which gives the height of a line joining* $\begin{bmatrix} u_x \\ u_y \\ f(\boldsymbol{u}) \end{bmatrix}$ *to* $\begin{bmatrix} v_x \\ v_y \\ f(\boldsymbol{v}) \end{bmatrix}$ *at a point* p *units along the line. To obtain a bilinear interpolation between the four corner points we define* f *on the boundary of its domain as:*

$$f\left(\begin{bmatrix}0\\y\end{bmatrix}\right) = h\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}0\\N\end{bmatrix}, y\right) = \frac{b-a}{N}y + a$$

$$f\left(\begin{bmatrix}N\\y\end{bmatrix}\right) = h\left(\begin{bmatrix}N\\0\end{bmatrix}, \begin{bmatrix}N\\N\end{bmatrix}, y\right) = \frac{c-d}{N}y + d$$

$$f\left(\begin{bmatrix}x\\0\end{bmatrix}\right) = h\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}N\\0\end{bmatrix}, x\right) = \frac{d-a}{N}x + a$$

$$f\left(\begin{bmatrix}x\\N\end{bmatrix}\right) = h\left(\begin{bmatrix}0\\N\end{bmatrix}, \begin{bmatrix}N\\N\end{bmatrix}, x\right) = \frac{c-b}{N}x + b$$

For internal values of the domain f *is defined as*

$$f\left(\begin{bmatrix}x\\y\end{bmatrix}\right) = h\left(\begin{bmatrix}x\\0\end{bmatrix}, \begin{bmatrix}x\\N\end{bmatrix}, y\right) = h\left(\begin{bmatrix}0\\y\end{bmatrix}, \begin{bmatrix}N\\y\end{bmatrix}, x\right)$$

If the four corner points of the surface S, $\begin{bmatrix}0\\0\\a\end{bmatrix}$, $\begin{bmatrix}0\\N\\b\end{bmatrix}$, $\begin{bmatrix}N\\N\\c\end{bmatrix}$, $\begin{bmatrix}N\\0\\d\end{bmatrix}$ *, lie on the same plane, S will be a plane containing those points. Otherwise S will be a saddle.*

In the case treated here the four corner points do not lie on the same plane so G_{Bilinear} generates a smooth saddle shape. This is shown in figure 2.11.

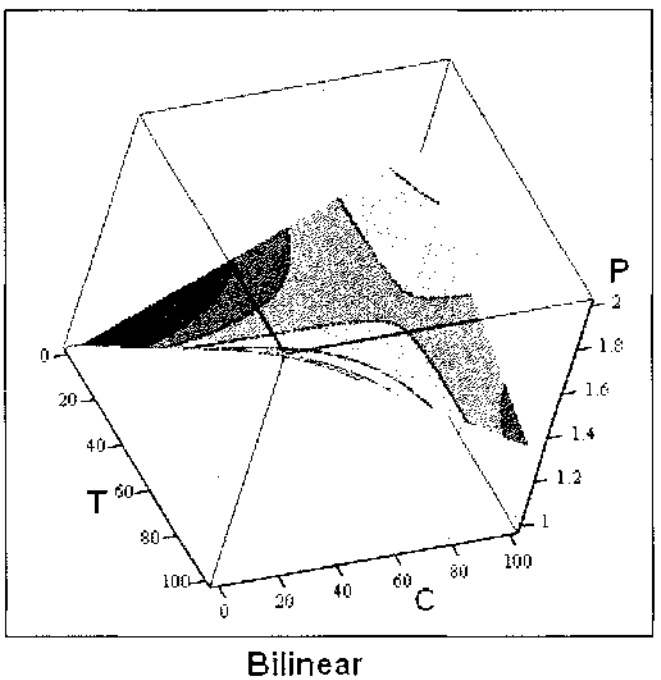


Figure 2.11: **Surface generated by the bilinear game matrix.** The T axis gives the length of the template, the C axis the length of the catalyst and the P axis the size of the payoff.

Simulations using G_{Bilinear} give results such as shown in the figures 2.12 and 2.13.

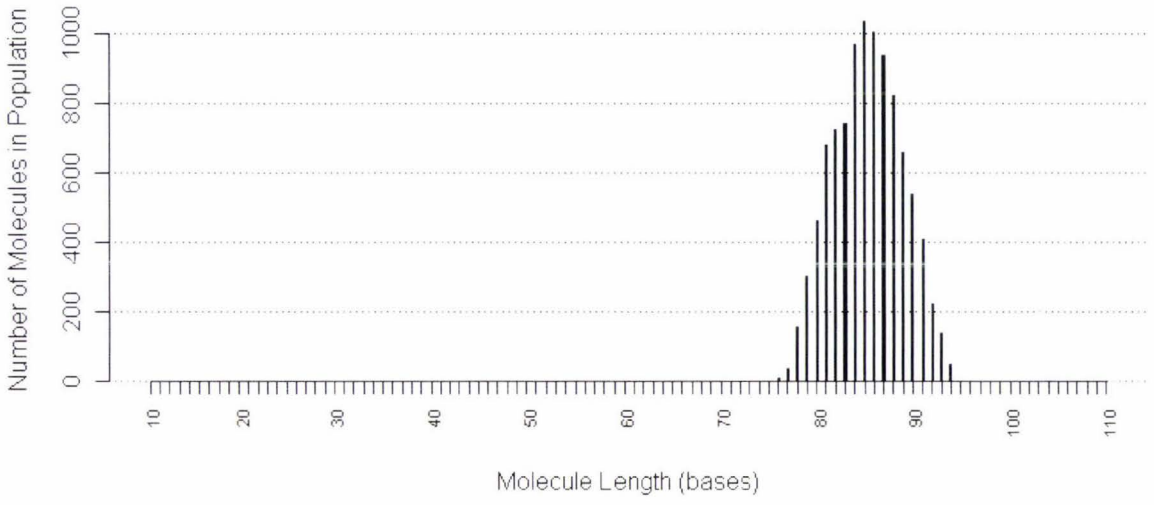


Figure 2.12: **Simulation results using the bilinear game matrix with random seed s_1 after 1 million cycles** starting from a uniform distribution. Bars represent the proportion of molecules in the population of each length. This distribution displays a single mode at 85.

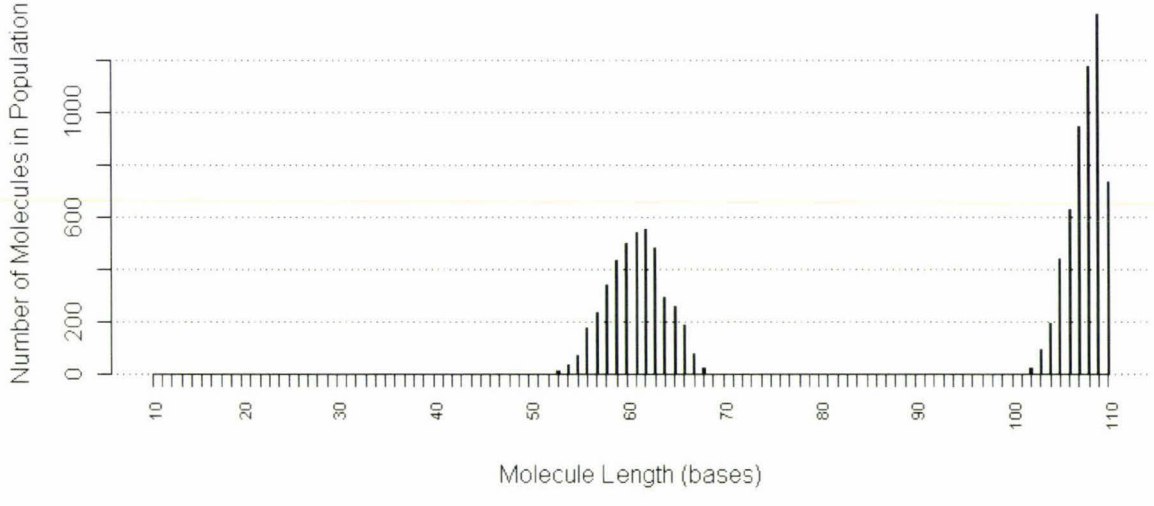


Figure 2.13: **Simulation results using the bilinear game matrix with random seed s_2 after 1 million cycles** starting from a uniform distribution. Bars represent the proportion of molecules in the population of each length. This distribution displays a two modes, one at 62 the other at 109.

These simulations have been run for 1 000 000 cycles. The observed distributions did

not change appreciably (retaining their modality with minor fluctuations in shape) for the last 900 000 cycles of the simulation run. Similarly to the planar models we see a unimodal and a bimodal result. However with the bilinear game the unimodal result is seen in more than 95% of the simulation runs. More than 200 simulation runs used each having a unique random seed. The only difference between the two simulation runs shown above is the initial random seed. The mode of the distribution in figure 2.12 is 85. The modes of the distribution in figure 2.13 are 62 and 108. Though the distributions differ in their modes their means are the same. Both have a mean of 85, corresponding to the 1/4 short replicators for an ESS in the simple model (section 2.2.2, remembering $l_{\min} = 10$). Both these distributions have been reached from the same initial conditions, the only difference between them is their random seed. This suggests that either the initial distribution lies near a catchment boundary between two ESS or that only one of these distributions can be an ESS. Arbitrarily changing the initial conditions still gives situations where we get both distributions. It is highly unlikely that several arbitrary changes in initial conditions will still leave us on the same catchment boundary. Thus it is most likely that one of these distributions does not represent an ESS. Given that this is the case it is likely that as the unimodal distribution occurs in more than 95% of model runs that this is the true ESS whilst the bimodal distribution is merely a long term instability in the simulation. Taking snapshots of the simulations while they are running shows that in the initial 100 000 cycles before the simulations settle in to one distribution, oscillations between the two modalities are observed. That the system generally moves away from the bimodal distribution into the unimodal distribution in the long term further supports the hypothesis that the bimodal distribution is unstable.

It is not possible to prove things conclusively using a simulation; only to gather corroborating evidence. However there are several factors which lead us to believe the unimodal distribution we have found is an ESS of the game. The occurrence of the unimodal distribution as a stable state in 95% percent of simulations after a million cycles lends some weight to this hypothesis. Further support is given in that the unimodal distribution is similar to the ESS predicted for the simplified two strategy game. Thus it is not unreasonable to suppose that the unimodal distribution is an ESS of the game. No conclusion

can be drawn about other ESS(s) of the game other than if they exist their catchment area is likely to be small.

2.2.7 Proving the hypothesized ESS.

To prove what we have found is an ESS we need to find a analytic or numerical solution to the game. Several attempts were made to provide analytical descriptions of this game. None of them succeeded although it may prove useful to further work in this field to explore the nature of that failure. We will look at the most promising of the methods tried, which was attempting to model the game using differential equations. Other approaches were attempted such as looking at the properties of the surface formed by the game matrix. However though these methods seem to have some intuitive value, there was not time to flesh this intuition into a quantitative result.

2.2.8 Differential Equation Model of the Bilinear Game.

Several different approaches were taken in this respect. First we explain the notation used for these equations, which should be familiar from earlier sections. Let $l_{\min}$ be the smallest length of molecule possible in the population x with largest molecular length $l_{\max} = l_{\min} + N - 1$. Let l index the lengths of the molecules in the population therefore $l = l_{\min}, \ldots, l_{\max}$. Let t (time) index the cycles of the game. Then x_l^t is the number of molecules in the population of length l on cycle t. $\boldsymbol{x}^t$ will refer to the population vector $\boldsymbol{x}$ at time t. We will use the bilinear game matrix G_{Bilinear}. Initially the model was approximated by the difference equation

$$\boldsymbol{x}^{t+1} = G_{\text{Bilinear}}\boldsymbol{x}^t \tag{2.14}$$

If we use $\dot{\boldsymbol{x}} = \boldsymbol{x}^{t+1} - \boldsymbol{x}^t$ to make a continuous approximation to this system we get the approximation to the game given in equation (2.15).

$$\dot{\boldsymbol{x}} = G_{\text{Bilinear}}\boldsymbol{x} - \boldsymbol{x} \tag{2.15}$$

The steady state distribution generated by this model can be easily calculated by finding the dominant eigenvector of G_{Bilinear}. This gives the distribution shown in fig: 2.14.

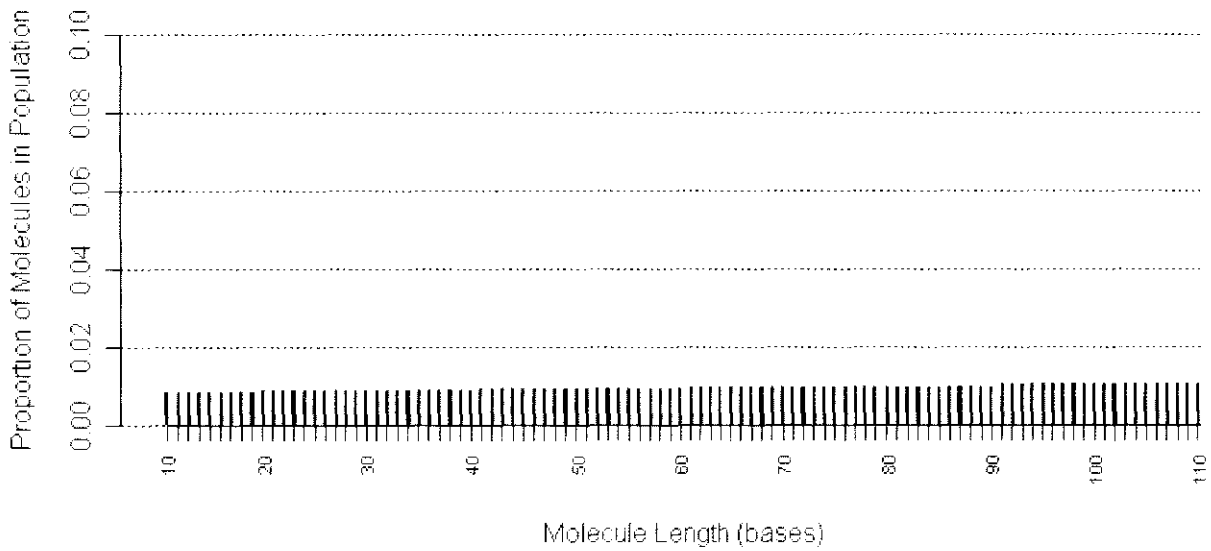


Figure 2.14: **Steady state distribution of molecules generated by equation** (2.15). Bars represent the proportion of molecules in the population of each length.

This is clearly quite a different result to the simulations. On reflection (2.15) fails to capture an important aspect of the game. This is the dependence of the number of molecules in the next generation of each length on the *proportion* of molecules of each length and not the absolute number in the current generation. If we let x_l^t represent the *proportion* of molecules of length l at time t we get a difference equation as in equation (2.16).

$$\boldsymbol{x}^{t+1} = \frac{G_{\text{Bilinear}}\boldsymbol{x}^t}{|G_{\text{Bilinear}}\boldsymbol{x}^t|_1} \tag{2.16}$$

Therefore $\dot{x}$ is given as in equation (2.17).

$$\dot{\boldsymbol{x}} = \boldsymbol{x}^{t+1} - \boldsymbol{x}^t = \frac{G_{\text{Bilinear}}\boldsymbol{x}^t}{|G_{\text{Bilinear}}\boldsymbol{x}^t|_1} - \boldsymbol{x}^t \tag{2.17}$$

Thus a better differential equation model than (2.15) for the bilinear game is given in equation (2.18).

$$\dot{\boldsymbol{x}} = G_{\text{Bilinear}}\frac{\boldsymbol{x}}{|\boldsymbol{x}|_1} - \boldsymbol{x} \tag{2.18}$$

The non-linear nature of these equations renders their solution difficult. I was unable to find a solution to this system of equations within the time constraints of this project.

Other strategies such as looking at vector plots of the differential equation were also attempted. This works for low dimensional problems but fails for higher dimensional problems. If we use a grid with m points in each dimension we need m^N grid points for an N phenotype game. This quickly becomes computationally infeasible for any useful grid size with large N (for example $m = 10,\ N > 15$).
For these and other reasons it was not possible to produce an analytic description for the ESS of the bilinear game within the time constraints of this project.

2.2.9 Conclusion: Game Theory Models.

We must therefore conclude that although the full bilinear game theory model (section 2.2.3) seems to produce a steady state distribution of mixed length molecules, there is not enough time in terms of this project to prove that this is true. To prove this we would need to demonstrate using analytical or numerical methods that the putative ESSs of the bilinear game observed are real ESSs.

2.3 Conclusion: All tested models.

A study of the models presented leads us to conclude that a simple linear model (section 2.1), if we use biologically reasonable parameter values, is insufficient to generate a steady state population of molecules with mixed length. The evolutionary game theory model (section 2.2.3), with its feed back mechanism, shows evidence that it is sufficient to generate a steady state population of molecules with mixed length. It would be interesting in future work to attempt to prove the evolutionary game model is sufficient for our purposes and to investigate additional possible models and find if a feedback mechanism is *necessary* to produce a stable population of molecules with a balanced length distribution.

3 Respiratory Syncytial Virus (RSV)

RSV is a single stranded negative sense RNA virus approximately 15 Kilobases in length. RSV infects most people multiple times throughout their life. In adults it causes symptoms like those of a cold; however in infants and the immunocompromised (such as some elderly or those undergoing immune depressant treatment) RSV can cause bronchiolitis, and in some cases death. In New Zealand two thirds of infants have contracted RSV by the age of one. New Zealand has a higher rate of hospital admissions due to RSV bronchiolitis than other developed countries and it is currently not known why this is so (Vogel *et al.* 2003). Its prevalence, effects and lack of effective treatments makes RSV a virus worthy of study.

In this thesis we look to answer the questions:

1. How has RSV evolved within New Zealand over the last forty years?
2. Are the strains of RSV found in New Zealand significantly different to those found overseas?
3. How does RSV in New Zealand interact with RSV internationally?

The New Zealand RSV samples used in this study were serendipitously stored in an ESR[1] freezer over a 30 year period from 1967 to 1997. The viruses had been put in culture, generally for 2 or 3 repassages in varying cell lines, before being frozen. Most samples are on a one sample one patient basis. However, in some cases, samples were either transferred into multiple cell lines, or multiple samples were taken from the same patient (not necessarily contemporaneously). These multiple samplings give a way to assess the strength of some sources of phylogenetic noise within the population. The serendipitous nature of the sampling means the sample we have is neither a full nor representative sampling of the virus population during the sampling period. This is to be expected when handling biological data, and it is important to keep this fact in mind when interpreting analyses.

[1]Environmental Science Research Ltd. http://www.esr.cri.nz/

There are two subtypes of RSV; A and B. The two subtypes differ from each other predominately in the composition of the Attachment Glycoprotein (G) (Polak 2004). The predominance of the different subtypes varies with epidemic year, though A seems to be more common than B, being the most frequently isolated strain in most epidemics (Hall *et al.* 1990). There are no documented symptomatic differences between the two subtypes.

3.1 The RS Virus: An Introduction.

RSV is a single stranded negative sense RNA virus from the Paromoxyviradae family about 80 to 350nm in size. It is related to viruses such as measles, mumps, and parainfluenza. A cartoon showing the structure of RSV is shown in figure 3.1.

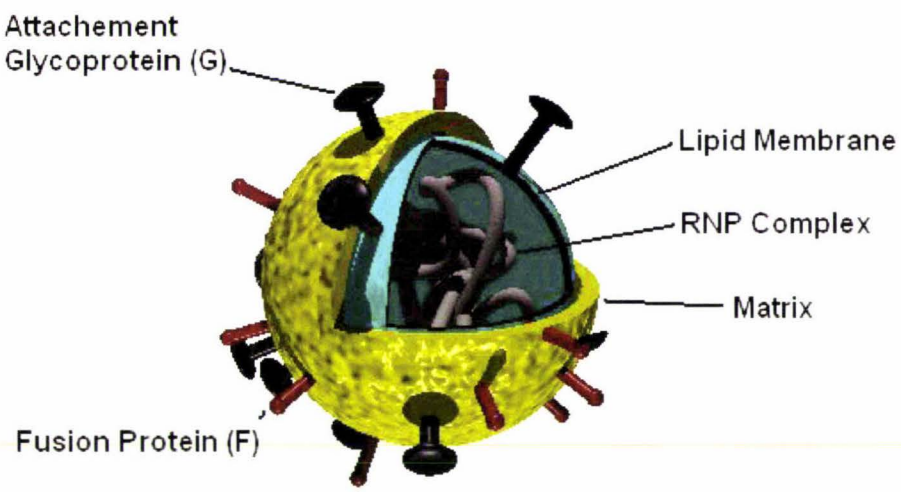


Figure 3.1: **Cartoon of the RS Virus.** The two surface proteins in this study are the Attachment Glycoprotein (G) and the Fusion protein (F). Figure created by the author using POVRay 3.0

The following description of RSV and its infective cycle is taken from Polak (2004) and Collins and Pollard (2002). RSV is highly transmissible and is spread through person to person contact or through contaminated surfaces in the environment (such as a tabletop that has been sneezed on). Once the virus has been contracted the incubation period is three to five days. Infection most often starts in the nasal epithelial cells and spreads

through the upper respiratory tract. In severe cases there will also be lower respiratory tract infection. Symptoms of upper respiratory tract infection can include coughing, sneezing and a mild fever. These persist for one to three weeks in infants but only 4 to 12 days in adults. Lower respiratory tract infection can result in wheezing, rapid breathing, pneumonia and death. Lower Respiratory tract infection rarely occurs in adults, as they will typically already have sufficient immunity from previous exposures to RSV earlier in life to prevent it.

Initial attachment to epithelial cells is initiated by the G protein, though viruses without a G-protein have been observed to still be infectious. The F protein allows fusion with the cell wall and entry into the cell. The virus replicates inside the infected cell and the infected cell is then destroyed to continue infection. The virus can also spread by using the F protein to fuse the infected cell to an adjacent cell to create a larger mutinucleated cell called a syncytia. This helps shield the virus from any humoral immune response. For this reason it is necessary to have high levels of antibodies for the body to be able to fight RSV infection.

3.2 General Methods

Extraction and sequencing of the viral sequences was done by Fenella Rich and Catherine Cohet at the Malaghan Institute for Medical Research[2] and is not part of this thesis. The isolation dates of the sequences used in this study are given in figures 3.2 and 3.3.

[2]Malaghan Institute for Medical Research *http://www.malaghan.org.nz*

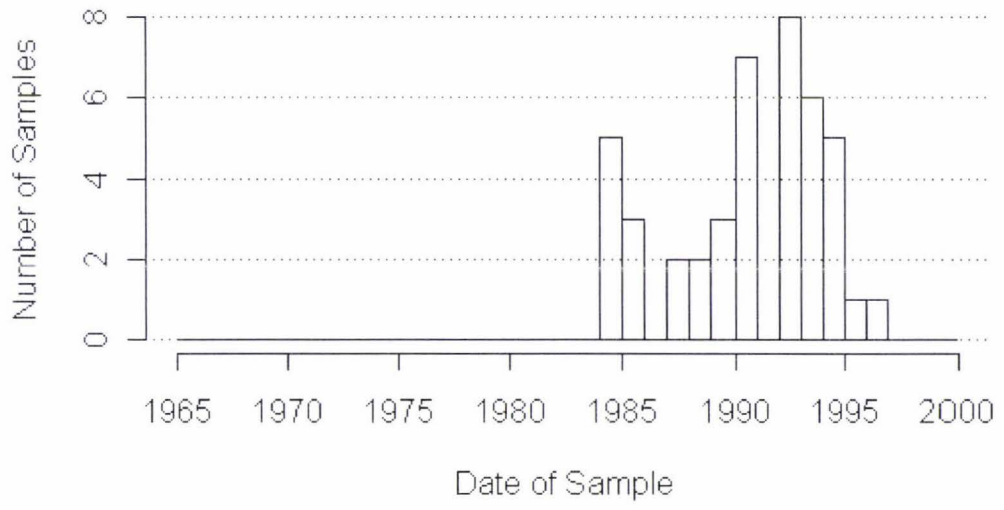


Figure 3.2: **RSV B Sampling dates.**

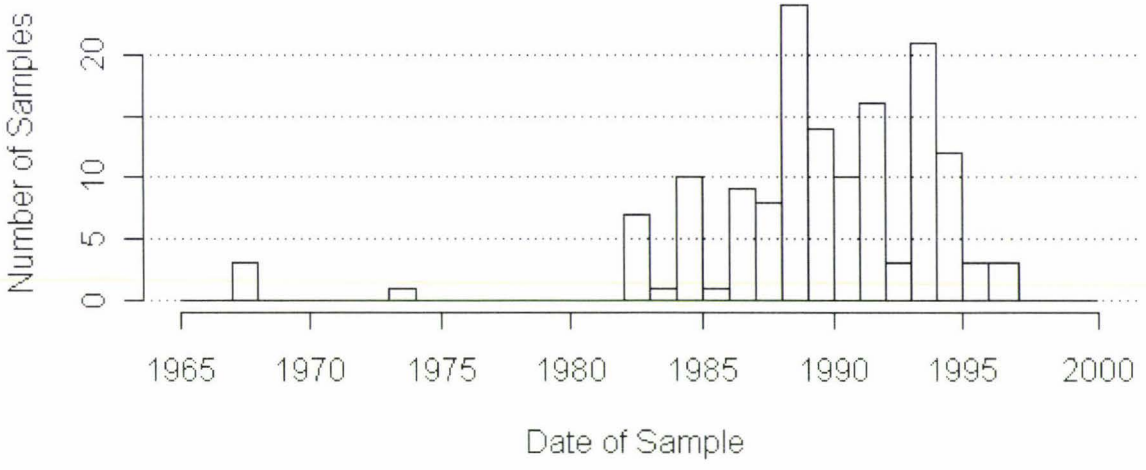


Figure 3.3: **RSV A sampling dates.**

The area of the virus sequenced contains parts of the the attachment glycoprotein (G) ectodomain, a 50bp spacer region (J) and part of the Fusion (F) protein cytodomain. Figure 3.4 shows the regions of the genome that were sequenced.

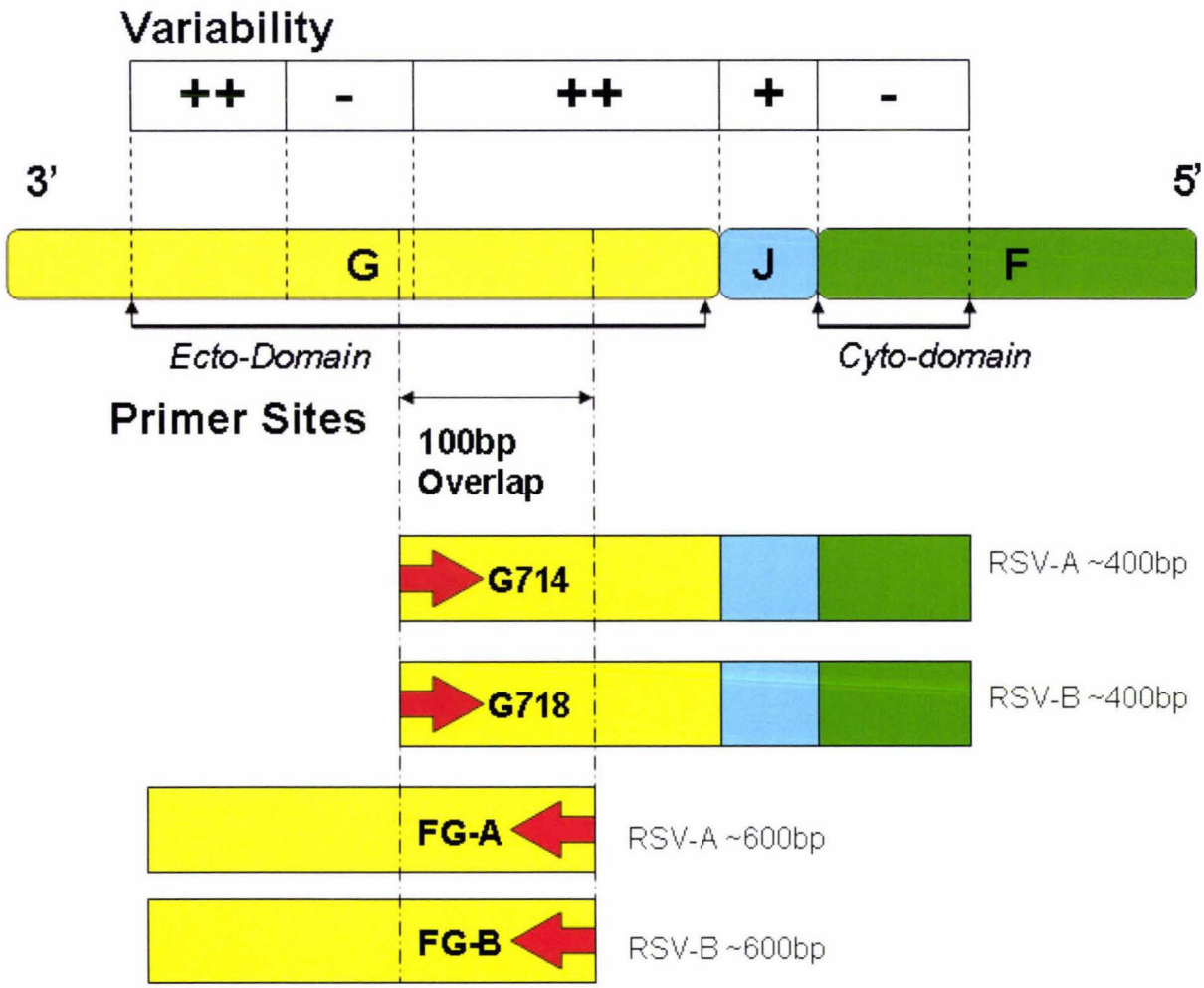


Figure 3.4: **Section of RSV Genome under study.** This figure shows the region of the RSV Genome covered by the sequenced regions and the degree of variability present in those areas. The degree of variability is indicated at the top of the diagram. (++) marks an area of high variability, (+) an area of medium variability and (-) an area of low variability. The bottom of the diagram shows the coverage of each of the primers used. Colours represent which region a part of the sequence comes from (yellow=G, blue=J, green=F)

Due to problems with sequence quality and availability in this thesis we look only at sequences from the RSV B G718 region and the RSV A FG-A region.

The changes in degree of variability along the length of the sequenced region can be attributed to differing levels of structural and environmental constraints on different regions of the proteins. The end of the G protein is exposed to the immune system and contains several recognized immune epitopes. The region also has little tertiary structure. This area is not functionally constrained and under strong selective pressure so it evolves very quickly. The intergenic spacer region (J) is unconstrained (or at least not known to be constrained) and evolves at a moderate rate. The section of the F protein sequenced transverses the lipid membrane and is thus highly conserved.

Once sequencing was complete the electropherograms were made available to me which is where my involvement in the project began. Moving from the electropherograms to the alignment involves several labour intensive steps. First the electropherograms must be individually checked for ambiguous base calls and as many ambiguous bases as possible correctly classified. Once this is completed an initial alignment is constructed using Clustal X. This alignment is then checked for any obviously bad gap placements and these are corrected. The final step, to produce as accurate an alignment as possible, is to check putative mutations in the alignment against the electropherograms to see if there have been any base miscalls. Also bases that were ambiguous after the first step can often be classified with reference to the alignment at this step. Depending on the size of the dataset, this process takes from one to three weeks to complete.

Once we have an alignment we need a method to analyze the data contained within that alignment. Several tools were used in this respect. Initially we tried using a tree representation of the data which was constructed using sUPGMA with the PEBBLE software (Drummond and Rodrigo 2000). The sUPGMA method assumes that sequences are mutating fast enough to have a substitution rate significantly different from zero and that the substitution rate is the same for all branches of the tree in a given sampling period (Drummond and Rodrigo 2000). We used this method as it takes into account the date of the samples, unlike a standard parsimony or maximum likelihood analysis. An

sUPGMA tree for RSV B sequences in New Zealand using data sequenced from the G718 primer site is shown in figure 3.5

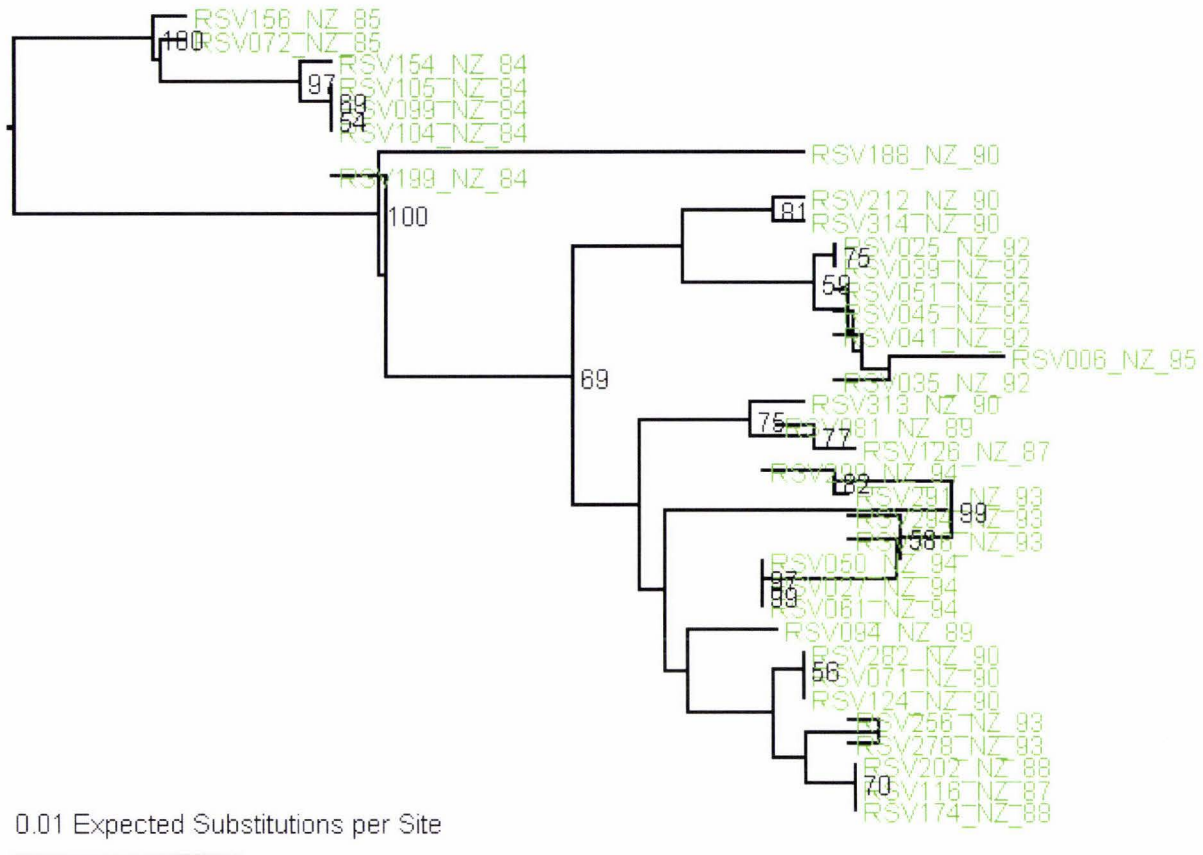


Figure 3.5: **sUPGMA tree for RSV B G718 primer site data**. The labels in the graph are in the format [Sample ID] [Sample Country] [Sample Date].

Unfortunately many branches have large negative branch lengths. This and facts such that the 1985 sequences are shown as occurring before the 1984 sequences (a negative substitution rate has been estimated for this time period) indicate that the data is violating the assumptions made by the sUPGMA method. Exactly how the data violates assumptions of sUPGMA is not clear at this stage. It could be that substitution rates among lineages differ or that the virus is mutating too slowly for the method to be effective. The

substitution rate of the virus is investigated in section 3.3.3. sUPGMA method was originally developed to analyse the evolution of HIV, which it does very effectively. However HIV differs significantly from RSV in its biology and life cycle. RSV is transmitted in a different manner and infects a much greater proportion of the population. sUPGMA was not used for further analyses because it does not suit this data.

It was decided to take an exploratory approach to the data analysis. Preliminary analysis of the data shows considerable conflicting phylogenetic signal, so I want to investigate the data in a way that takes this conflicting signal into account. This conflict is demonstrated by the more than 10000 trees that fit the data when using heuristic maximum parsimony analysis in PAUP* (Swofford 1998). It was decided to look at the data in terms of networks (Bandelt and Dress 1992). Networks allow us to see conflicting signal in the data and allow us to easily take an exploratory approach to the analysis of the data.

3.2.1 Networks

Before we continue let us look at what networks are. Trees can be considered to display the first principle component of the data. A network extends the dimensionality of the analysis, looking at the first two or three principle components. Networks are used to display collections of splits on a dataset. Thus to understand networks we must first understand splits. Splits are defined in Definition 3.1 (Semple and Steel 2003).

Notation. *Sequences.*
Inside algorithm descriptions and definitions the word sequence refers to a sequence in the mathematical sense and not a nucleotide sequence, unless explicitly qualified. Sequences are used to simplify later algorithm descriptions. We will often have need to describe subsequences of a sequence. Let S be a sequence. The subsequence U of S can be described by a sequence X of positive integers specifying by index which terms of S appear at what position in U. This is written $U = S|_X$.

Definition 3.1. *Split*

Let Σ be a set. A **split** *on Σ is a bipartition of Σ. For example if $\Sigma = \{a, b, c, d\}$ then a split is formed by the partitioning of Σ into $\{a, b\}$ and $\{c, d\}$. We can write this as a split* $Y = \text{ab|cd}$. *The two parts $\{a, b\}$ and $\{c, d\}$ are called* **sides** *of the split Y.*

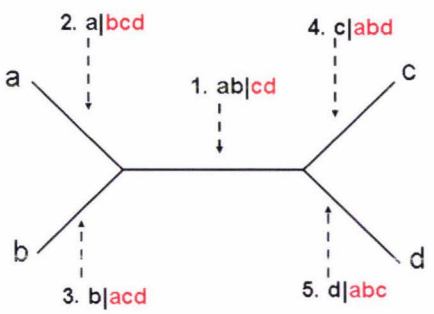
In later definitions and algorithms we will be working with sequences so we will also define the equivalent of a **split on a sequence**. *A split on a sequence T is defined by two subsequences of T , L and R. These are called the sides of the split and have the following properties:*

1) No term of T occurs in both L and R.

2) All terms of T occur exactly once in either L or R.

We can write a split on a sequence in the same way as we can on a set. For example if $T = (a, b, c, d)$ then $Y =$ ab|cd *represents a situation where one of L or R is (a, b) and the other is (c, d). In considering splits on sequences the order of the terms in L and R is unimportant.*

Each edge in a phylogenetic tree represents a split as demonstrated in figure 3.6.



Split No. > / Taxon V	1	2	3	4	5
a	A	T	T	G	C
b	A	C	G	G	C
c	T	C	T	C	C
d	T	C	T	G	A

Figure 3.6: **Splits on a tree.** Each edge is labeled with the split it represents. The table shows what the data generating the displayed splits might look like. If two taxa share the same base at a site then they are on the same side of the split generated by that site. The numbers in the tree edge labels refer to the split number in the table. Items on the same side of a split at a site are given the same colour in the table. The split corresponding to an edge can be determined by looking at the taxa in each of the two subtrees formed by removing that edge.

It is not always possible to display all the splits in a dataset on the same tree. Two splits that cannot be displayed on the same tree are termed *incompatible*. Let $\Sigma = \{a, b, c, d\}$ be a set. $Y_1 = \text{ab|cd}$ and $Y_2 = \text{ac|bd}$ are an example of incompatible splits. This is shown in figure 3.7.

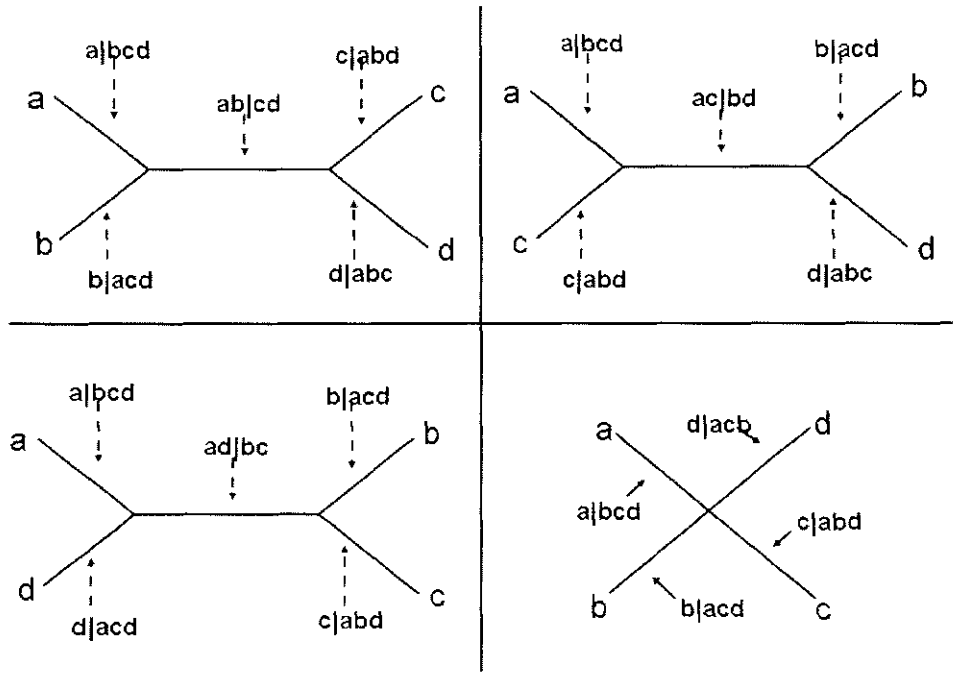


Figure 3.7: **Incompatible Splits.** Here we show all possible trees on 4 taxa and the splits they contain. The splits Y_1 and Y_2 are each represented singly on one of the trees but they do not appear together on any of the trees. These splits do not appear together on the same tree when we look at all four taxon trees thus they cannot appear on the same four taxon tree.

The inability to display conflicting splits holds for n-taxon trees (Bandelt and Dress 1992). To display incompatible splits we use a network such as the one in figure 3.8.

Splits can be taken directly from an alignment. If a substitution occurs in the column of an alignment then all samples with that substitution form one side of a split (see figure 3.6). If three or four bases are present at the same site the site needs to be resolved to Purine-Pyrimidine (R-Y) coding . As an example the alignment column AAGGCC would be resolved to RRRRYY. It is also possible to resolve a 3- or 4 way set partition into several fractional weight splits. This had little effect on the results of the analysis when trialed however so was not used in the full analysis.

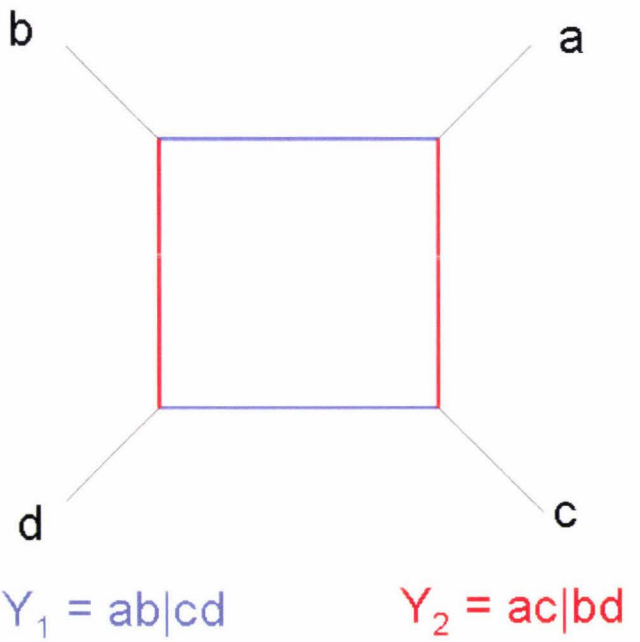


Figure 3.8: **Example Network**. This is a network showing both Y_1 and Y_2. Parallel edges (coloured) represent a single split.

In the networks used in this project edge lengths correspond to the number of splits contributing to an edge. Even using networks we cannot usually look at all of the splits in a dataset, as this will often result in a network of too high a dimension to be useful. Networks lie on a scale which has trees at one end (no incompatible splits) and high dimensional object displaying all the splits at the other. This is displayed in the treeness scale in figure 3.9.

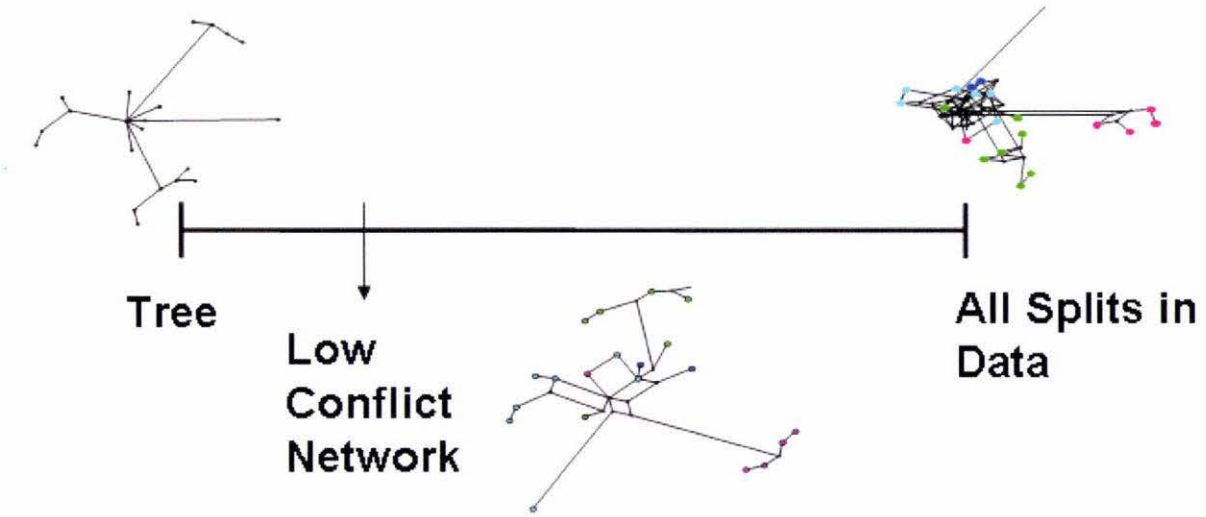


Figure 3.9: **Treeness Scale.** The representations obtained by looking at different levels of conflicting splits in a dataset. By choosing the maximum amount of allowable conflict in the splits in a data set to display we can create anything from a tree (no conflict) to a phylogenetic inkblot (all splits). By displaying only a few more splits than we see on a tree we can get a human interpretable network (middle).

Figure 3.9 demonstrates the need for filtering of splits when displaying them for human interpretation. The splits displayed in the networks presented in later sections generally contain only those splits which occur more than once in the data plus all those which occur only once but for which there are less then a threshold number of conflicting splits. All global tests done on the networks are done on the full selection of splits in the dataset as well as the reduced sets displayed in the networks. An example of looking at different levels of split filtering is shown below in figures 3.10 to 3.12 for the RSV B G718 data. Small vertices in these networks are intermediate sequences for which no actual sequence exists.

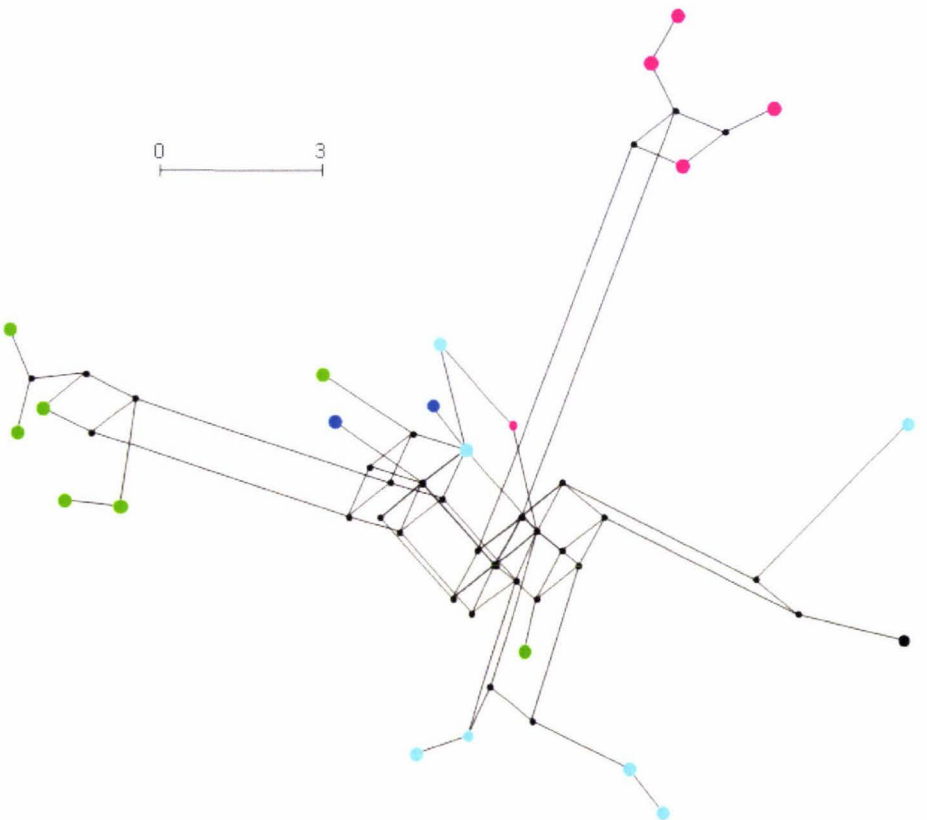


Figure 3.10: **All Splits in the RSV B G718 Alignment** (The G718 primer site is defined in figure 3.4). Groups of parallel lines in this figure represent splits. The length of each line represents the number of times a split occurs as specified by the scale in the upper left corner. The small black vertices are intermediate vertices, the larger coloured vertices are vertices representing sequences. Notice the cuboids in the center of the graph, these are caused by a triplet of mutually incompatible splits. Also notice the large number of intermediate vertices and the two strongly grouped clusters of green and purple vertices. (colours represent date of isolation but this is not important in this context)

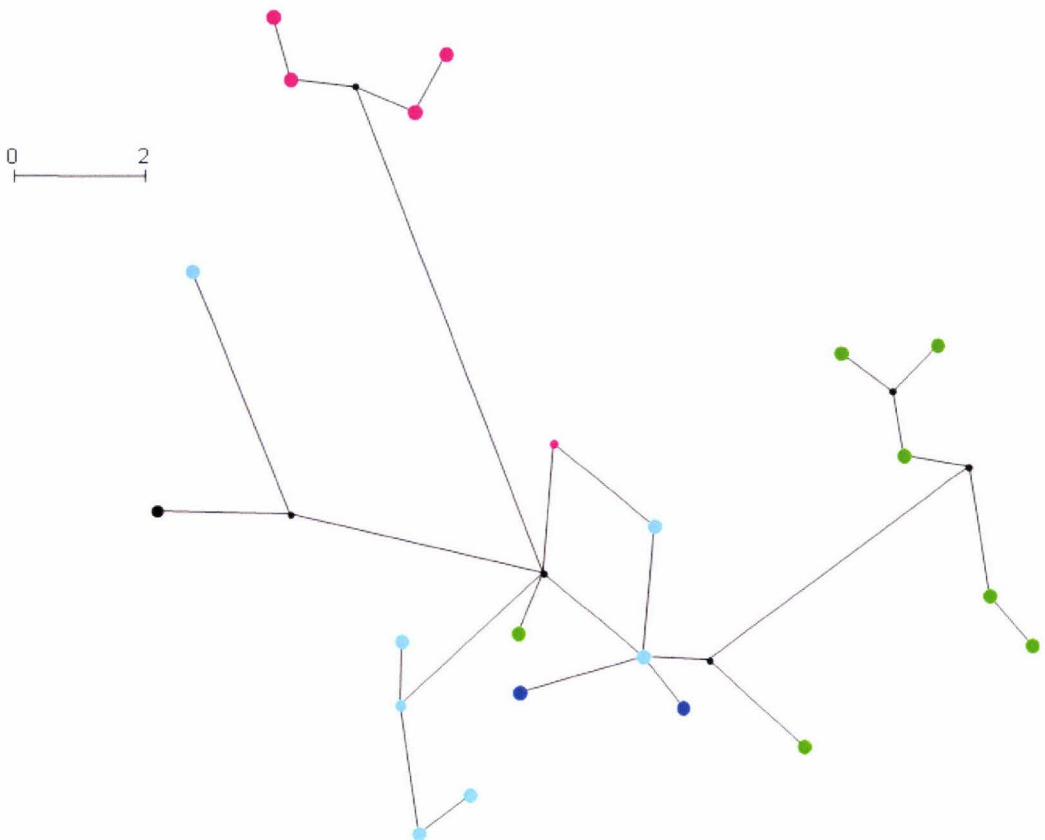


Figure 3.11: **Medium Number of Splits Displayed:** All splits that occur more than twice *plus* splits which have less than two conflicting splits in the RSV B G718 alignment. Notice there are no cubes and only one square in the graph. Also there are far fewer intermediate vertices. We still see the distinctive green and purple groupings.

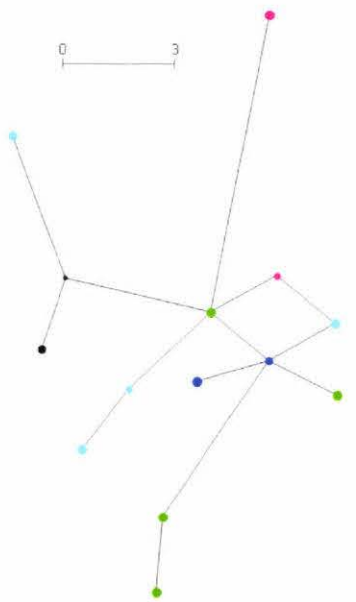


Figure 3.12: **Least Number of Splits Displayed:** All splits occurring at least twice in the RSV B G718 alignment. We still have one square, but now only one intermediate vertex. The number of vertices in the green and purple groups has decreased, as there are now more sequences at each vertex.

We define the concept of a split threshold in Definition 3.2.

Definition 3.2. *Threshold n split-network.*
A network with threshold n displays all splits that occur more than once as well as all splits that occur exactly once and have less than n conflicting splits in the dataset. For example figure 3.11 is a threshold 2 network.

3.2.2 Permutation Tests

Many of the tests and methods used in the analysis that follow use permutation tests. Permutation tests allow us to gain a picture of where a value lies within a distribution without knowing what that distribution is a-priori (Higgins 2004). To avoid unnecessary repetitions in later algorithm descriptions we briefly look at what is involved in a typical permutation test in this project.

Definition 3.3. *Property function*

Let S be a finite sequence of items. Items may be anything, numbers: DNA sequences or fruit for example.

Let $q : S \mapsto Q$ be a **property function** *defined on S. q maps each term of S to a term of Q a second sequence of items giving the value of a property of the items in S. A* **property** *in this context follows the normal English language definition of the term, "an attribute or quality" belonging to the items in S (Onions 1959). Q, the image of $q(S)$ is of the same length as S. Each term of Q gives the property value for the corresponding term in S.*

Example:

Let S = (apple, orange, banana, lettuce). A property function q on S for the property colour could be defined by $q(S)$ = (green, orange, yellow, green). Note that property values are not necessarily unique as demonstrated in this example.

Let X be an integer sequence defining a subsequence of S.

q is defined on a subsequence of S as $q(S|_X) = Q|_X$

Example:

Let S = (apple, orange, banana, lettuce) as above with colour property function $q(S)$ = (green, orange, yellow, green). Let U = (apple, lettuce) be a subsequence of S (here $X = (1, 4)$) then $q(U)$ = (green, green).

We call the values a property of a sequence can assume the property labels for that sequence.

Example:

Let S = (apple, orange, banana, lettuce). Let the property function q on S be defined by $q(S)$ = (green, orange, yellow, green). The property labels for q on S are {green, yellow, orange}. *Notice that this is a set.*

Algorithm 3.4. *Permutation Test*

Inputs:

S a sequence of items.

A subsequence of S that we will investigate defined by integer sequence X.

A property of the items in S with property function $q : S \mapsto Q$.

$f : Q \mapsto \mathbb{R}$ a function used to take some measure of q on the subsets of S. An example of such a measure for numeric Q is the mean of Q.

Parameter c, the level of confidence of the test. A typical value of c is 95%.

Parameter j, the number of iterations used. A typical value of j is 1000 to 10000. j is chosen to minimize the computational time required while maximizing the quality of the result.

Outputs:

Answer to one of the questions:

"Given S and X, is the observed value of $f(q(S|_X))$ greater than would be expected if q was random with respect to f?"

"Given S and X, is the observed value of $f(q(S|_X))$ smaller than would be expected if q was random with respect to f?"

Procedure:

Step 1: Record $x = f(q(S|_X))$.

Step 2: Permutation Step.

Let $n = |S|$.

Let P be sequence containing a random permutation of the integers $\{1, 2, \ldots, n\}$.

Let $S' = S|_P$. Thus S' is a random reordering of the terms of S.

Step 3: Sampling Step.

Record $x' = f(q(S'|_X))$.

Step 4:Iteration Step.

Construct a distribution D by repeating steps 2 and 3 j times, recording the value of x' in D each time.

Step 5: Evaluation Step: Assess the significance of x in light of distribution D.

To answer the question "Given S and X, is the observed value of $f(q(S|_X))$ greater than would be expected if q was random with respect to f?" we do the following

Find z; the percentile of D corresponding to c. For example if $c = 95$ then z is the 95th percentile of D.

If $x > z$ then we can answer "Yes" with a confidence of c.

To answer the question "Given S and X, is the observed value of $f(q(S|_X))$ smaller than would be expected if q was random with respect to f?" we do the following

Find z; the percentile of D corresponding to $100 - c$. For example if $c = 95$ then z is the 5th percentile of D.

If $x < z$ then we can answer "Yes" with a confidence of c.

3.3 Evolution of RSV in New Zealand

3.3.1 Introduction

The data available allows us to investigate several interesting aspects of the evolution of RSV in New Zealand in the 1967 to 1997 period. We look at the pattern of evolution and in particular we ask is it characteristic of antigenic drift? This question is of interest is there is disagreement over whether RSV undergoes antigenic drift or not (Cane and Pringle 1995, Sullender 2000). Antigenic drift is a process where immune system pressure forces the continuous but gradual evolution of a virus within its host population (Janeway *et al.* 2001). In a situation where a virus is undergoing antigenic drift we would expect viral evolution to be correlated with time, with new strains replacing old. We would not expect to see the extended persistence of old strains of the virus. When translated to a network this means that a virus population undergoing antigenic drift will show strong clustering by the year of isolation. The nature of this clustering will depend upon the number of different subpopulations the virus population can be split into. This is discussed in figure 3.13.

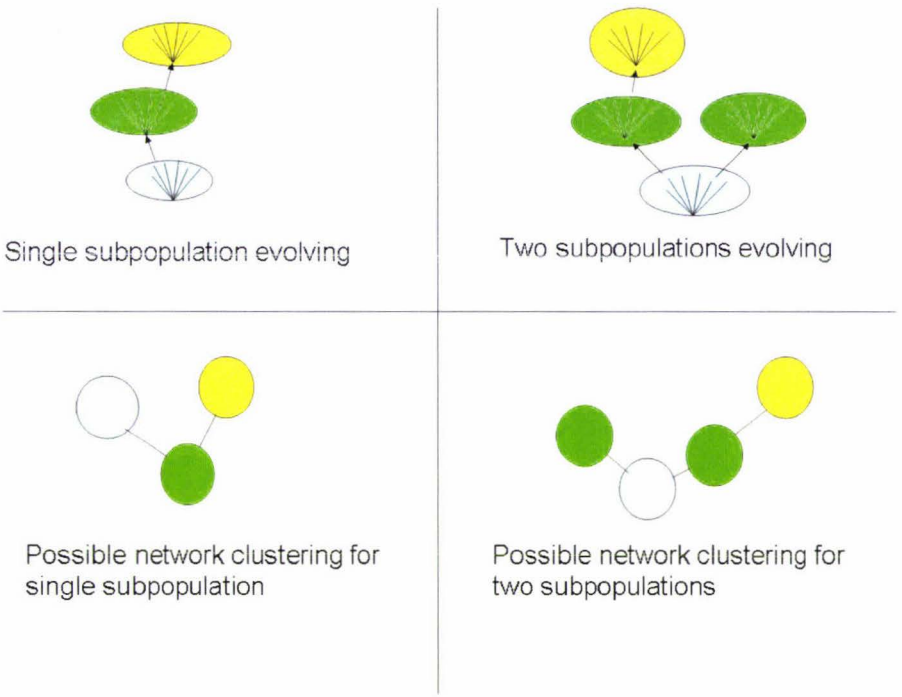


Figure 3.13: **Antigenic Drift Effects Example**. Kinds of clustering that can be formed by different population structures under antigenic drift. Ovals enclose distinct clusters. Colours represent the average time of isolation for the cluster. The top half of the figure gives an evolutionary history, the bottom how it might be represented on a network.

By looking at clustering by sample isolation time in the graph we may find some indication of the presence or absence of antigenic drift. It should be noted that antigenic drift is not the only factor that can contribute to temporal clustering of sequences. This can also arise through neutral genetic drift under the Fisher-Wright model (Fisher 1930, Wright 1930). The methods used to look at clustering are detailed in section 3.3.2.
Also of interest is how fast the virus is evolving and the nature of the changes observed. We look at the substitution rate of both RSV A and B using the method detailed in section 3.3.2.3.

3.3.2 Methods

3.3.2.1 Global Clustering Test A permutation test called the 'Global Clustering Test' was developed by the author to investigate the clustering of the RSV samples within New Zealand on the networks produced in the alignment step and to test for antigenic drift. This algorithm was implemented (also by the author) in R (R Development Core Team 2004). Before looking at the global clustering test algorithm (Algorithm 3.11) we define the notion of split congruence (Definition 3.5) and related concepts which are used in the global clustering test.

Definition 3.5. *Split Congruence.*

Let Y be a split on the sequence S. We will arbitrarily label the two sides of the split L and R.

Let $q : S \mapsto Q$ be a property function of S.

The split Y is congruent to q if $q(L) \cap q(R) = \phi$. Thus if any given label defined by q falls on only one side of the split Y then Y is congruent to the property q.

Definition 3.6. *Degree of Congruence*

Let Y be a split on the sequence S. We will arbitrarily label the two sides of the split L and R.

Let $q : S \mapsto Q$ be a property function of S. Let the unique terms of Q be called the property labels of q.

Let $Q_L = q(L)$ and $Q_R = q(R)$. If Q_L and Q_R have no terms in common then q is **fully** *congruent to Y.*

The **degree of** *congruence of q to Y is a summary measure reflecting to what extent property values defined by q fall on only one side of the split Y. For each property label of q we can count the number of times that label occurs in both L and R. If there is a high degree of congruence between q and Y then most property labels will fall on only one side of Y, or will occur at high frequency on one side of Y and at low frequency on the other. If there is a low degree of congruence between q and Y then a majority of the property labels will fall with equal frequency on both sides of Y.*

As before let Y be a split on the sequence S and $q : S \mapsto Q$ be a property function of S. Remember the unique terms of Q are called the property labels of q (Definition 3.3).
To measure the degree of congruence in a quantitative manner we might try taking a summary statistic over all property labels using the ratio of the number of times a property label occurs on one side of the split Y to the number of times it occurs on the other side. However this method introduces a problem. If the split Y is unbalanced, such that one side contains many fewer terms than the other, then there will be a bias towards calling Y congruent. This bias will occur as fewer terms occurring on one side means it is less likely any given property label will occur on that side.

We can avoid this bias. First we we define the concept of a large and small side of a split.

Definition 3.7. *The large and small sides of a split.*
Let Y be a split on the sequence S. The large side of Y is the side which contains the greatest number of terms. If both sides have an equal number of terms then the large side of Y is the side containing the term s_0 of S. The small side of Y is the side of Y which remains unlabeled once the large side of Y has been determined.

To avoid bias, instead of comparing the two sides of Y, we compare a single side of Y (the large side) to a background distribution of sequences, generated by taking the large side of Y applied to S with property function $q' : S \mapsto Q'$, where Q' is a random permutation of the terms of Q. This is done by means of a permutation test as documented in Algorithm 3.11. We use the large side of Y as it contains more information than the small side for unbalanced Y and thus gives a more varied and hence informative distribution for the purposes of comparison. We use only one side to avoid scaling problems that arise in attempting to correctly scale results from each side to give equivalent contributions to the measure.

To test for degree of congruence in the manner detailed above, we must use a measure which gives a score proportional to the degree of congruence of q with Y from a single side of Y. Such a measure can be created using the concept of degree of homogeneity.

Definition 3.8. *Degree of Homogeneity*

Let S be a sequence. S is homogeneous if all terms of S are identical. The **degree of homogeneity** *of S is defined by a scale going from fully homogeneous where all terms of S are identical to fully heterogeneous where all terms of S are unique.*

If q has a high degree of congruence to Y, then the number of property labels occurring on on each side of Y will be small compared to a situation of low congruence, because the number of property labels contributing to the counts of more than one side of the split will be reduced. Thus the degree of homogeneity in both sides of Y will be larger than in a situation of low congruence. Conversely if the degree of homogeneity of both sides of Y is large, then the number of property labels occurring on each side of Y must be small compared to a low homogeneity situation, therefore the level of congruence will be high. Thus degree of homogeneity can be used to measure degree of congruence. Algorithm 3.11 uses degree of homogeneity to estimate degree of congruence.

Definition 3.9. *Cluster Measure*

Let S be a sequence of items.

Let $q : S \mapsto Q$ be a property function on S. A **cluster measure** *$C : Q \mapsto \mathbb{R}$ on q measures the degree of homogeneity of q in S.*

In the tests used in this project the sequence S always represents one side of a split.

Definition 3.10. *Global Clustering Test (GCT)*

Let $N = \{Y_1, \ldots, Y_n\}$ be a network defined by the n splits $\{Y_1, \ldots, Y_n\}$ on the sequence S of nucleotide sequences. Let q be a property function on S. The Global Clustering Test (GCT) on N uses a cluster measure(Definition 3.9) to determine the degree of congruence between Y_i and q for each split $Y_i, i \in \{1, \ldots, n\}$ in N. These values are used to determine if N shows clustering by property q.

The algorithm used in the Global Clustering Test is defined in Algorithm 3.11.

Algorithm 3.11. *GCT Algorithm*

This algorithm is based on the permutation test algorithm detailed in Algorithm 3.4. The main differences from Algorithm 3.4 are in the sampling step.

Inputs:

S *a sequence of nucleotide sequences.*

$N = \{Y_1, \ldots, Y_n\}$ *a network defined by the* n *splits* $\{Y_1, \ldots, Y_n\}$ *on the sequence* S.

q *a property of* S *with property relation* Q.

$C(U) : Q \mapsto \mathbb{R}$ *a cluster measure on* q.

Parameter c *the level of confidence of the test.*

Parameter j *the number of iterations to use.*

Outputs:

Answer to the question: "Is the mean degree of congruence over each of the splits in N *with* q *greater than would be expected if* q *were randomly distributed with respect to the splits in* N*?"*

Process:

Let the large side of split $Y_i \in N$ *be designated* L. *From Definition 3.1 we know* L *is a subsequence of* S. *Let the integer sequence defining* L *for* Y_i *be referred to as* $X[i]$.

The function $C(q(\ldots))$ *gives us a measure of the degree of homogeneity for each split in the network* N. *This test aims to produce a single score for the entire network. We will refer to this score a* **network score**. *We calculate the network score as follows:*

Let $\mu(Z)$ *be the mean of sequence* Z *where the terms of* Z *are real.*

Let $A = \{C(q(S|_{X[i]})) \mid i \in \{1, \ldots, |N|\}\}$ *be a set of cluster measures for the splits in network* N.

The network score for A *is* $\mu(A)$

Step 1: Score the original network

Let $A = \{C(q(S|_{X[i]})) \mid i \in \{1, \ldots, |N|\}\}$

Record $x = \mu(A)$, *the network score for* A.

Step 2: Permutation Step (identical to Algorithm 3.4)

Let $m = |S|$.

Let P *be sequence containing a random permutation of the integers* $\{1, 2, \ldots, m\}$.

Let $S' = S|_P$. *Thus* S' *is a random reordering of the terms of* S.

Step 3: Sampling Step

Let $A' = \{C(q(S'|_{X[i]})) \mid i \in \{1, \ldots, |N|\}\}$, *the cluster measures for* N *when* S *has been reordered.*

Record $x' = \mu(A')$ *the network score for* A'.

Step 4: Iteration Step (identical to Algorithm 3.4)

Construct a distribution D *by repeating steps 2 and 3* j *times, recording the value of* x' *in* D *each time.*

Step 5: Evaluation Step.

Determine if x *is smaller than would be expected were* q *not congruent to the splits in* N. *Use the method detailed in Algorithm 3.4's Evaluation step (step 5).*

We use the GCT to study grouping by date on our networks. To do this we need a cluster measure for sampling date. This is defined in Definition 3.12.

Definition 3.12. *Date Clustering Measure*

The date Clustering Measure is defined as $\sigma(Z)$ *where* $\sigma(Z)$ *is the standard deviation of the sequence* Z *where the terms of* Z *are real numbers. This measure measures both the spread and homogeneity of the data.*

Using the global clustering test with the date clustering measure tells us if the data is split on the basis of date. This does not give us any information on the nature of any vertex clustering in a network. To investigate this we use the Local Clustering test.

3.3.2.2 Local Clustering Test This test was developed and implemented in R (R Development Core Team 2004) by the author. First we define the notions of path length (Definition 3.13) and Topological clustering (Definition 3.14) which are used in the Local clustering test.

Definition 3.13. *Path Length* $d(x, y) : V \times V \rightarrow \mathbb{R}$*:*
Let x *and* y *be two vertices in a connected phylogenetic network* N*. Let* P *be the set of all paths joining* x *and* y*. The path length* $d(x, y)$ *between* x *and* y *is the number of edges on the shortest path in* P*. This defines a metric on* N*.*

Definition 3.14. *Topological Clustering*
Let $N = \{Y_1, \ldots, Y_n\}$ *be a network defined by the* n *splits* $\{Y_1, \ldots, Y_n\}$*on the sequence* S*.*
Let $q : S \mapsto Q$ *be a property function on* S*.*
Let v *be a vertex in network* N*.*
There is a full **topological cluster** *of size* k *on* N *by property* q*, centered on vertex* v*, if all vertices within path length* k *of* v *have labels* $L \subset S$ *that all yield the same value under* q*. For example if we are looking at topological clustering by date then if all vertices within path length* k *of* v *have the same date associated with them then* v *clusters topologically perfectly by date for path length* k*.*

Definition 3.15. *Local Clustering Test*
Let $N = \{Y_1, \ldots, Y_n\}$ *be a network defined by the* n *splits* $\{Y_1, \ldots, Y_n\}$*on the sequence* S*. Let* $q : S \rightarrow Q$ *be a property function on* S*. The local clustering test uses a cluster measure to investigate the degree of topological clustering of property* q *on network* N*.*
Each labeled vertex is tested to see if it lies in the center of a cluster for cluster sizes $k \in \{1, \ldots, \gamma\}$ *where* γ *is the maximum cluster radius.*

Algorithm 3.16. *Local Clustering Test Algorithm*

Inputs:

S *a sequence of items.*

$N = \{Y_1, \ldots, Y_n\}$ *a network defined by the* n *splits* $\{Y_1, \ldots, Y_n\}$*on the set* S*.*

A property of the items in S *with property function* $q : S \mapsto Q$*.*

$C(U) : Q \mapsto \mathbb{R}$ *a cluster measure on* q*.*

Parameter γ *the maximum cluster radius of the test.*

Parameter j *the number of iterations to use.*

Outputs:

For each vertex in network N *answers the question: "What confidence do I have that this vertex lies at the center of a statistically significant topological cluster with radius* γ *or less?"*

Procedure

Let $V = \{v_1, \ldots, v_m\}$ *be the* m *labeled vertices of network* N*. A labeled vertex is a vertex which represents one or more of the terms in* S*. Let the terms of* S *represented by vertex* v_i *define the set* $T_i \subset S$*.*

Step 1: Determine initial score for each vertex.

For each vertex $v \in V$*.*

For each $k \in \{1, \ldots, \gamma\}$

Let T_v^k *a subsequence of* S *be the terms of* S *represented by vertices with a path length of* k *or less from* v*. Let* $X[v, k]$*, be the integer sequence defining the terms of* S *in* T_v^k

Record the value of the cluster score $x_v^k = C(q(S|_{X[v,k]}))$ *for vertex* v *and radius* k*.*

Step 2: Permutation Step (identical to Algorithm 3.4)

Let $r = |S|$.

Let P *be sequence containing a random permutation of the integers* $\{1, 2, \ldots, r\}$.

Let $S' = S|_P$. *Thus* S' *is a random reordering of the terms of* S.

Step 3: Sampling Step.

For each vertex $v \in V$.

For each $k \in \{1, \ldots, \gamma\}$

Record the value of the cluster score $x'^k_v = C(q(S'|_{X[v,k]}))$ *for vertex* v *and radius* k.

Step 4: Iteration Step.

Construct a distribution D^k_v *for each vertex and cluster radius by repeating steps 2 and 3* j *times, recording the value of* x'^k_v *in* D^k_v *each time.*

Step 5:Evaluation Step

For each vertex $v \in V$ *in* V

For each $k \in \{1, \ldots, \gamma\}$

Determine the confidence with which v *can be at the center of a significant topological cluster of size* k *by testing* x^k_v *is smaller than expected given distribution* D^k_v *at 95% and 99% confidence (using the methods in Algorithm 3.4). Thus* v *will be classified into one of three categories.*

1) Not significantly topologically clustered.

2) Significantly topologically clustered at 95% confidence.

3) Significantly topologically clustered at 99% confidence.

If D^k_v *has zero variance then we are unable to tell anything about the significance of the topological clustering for* v *at radius* k.

The result of the local clustering test tells us with what confidence each labeled vertex in the network N is at the center of a cluster in terms of the property being tested. For some vertices the cluster measure will be the same for all permutations of the labels. These vertices are marked as not returning a result as we can gain no information about them from this test. We first use the Local Clustering test to look at topological clustering by date. We use the clustering measure defined in Definition 3.12 to do this. In the analysis documented here we use $\gamma = 3$. This value of γ was selected as visual inspection of the graphs showed that this is the largest significant cluster.

3.3.2.3 Substitution Rate Estimation The substitution rate of the sequences was also estimated from the alignment. Using the number of observed substitutions from pairwise comparisons between all of the sequences in the alignment will not give a good estimate for the substitution rate, as there will be significant interdependence between these distances. A permutation test solution was developed by the author to estimate the background noise generated by dependencies in the data set. This test works by looking at what the relationships between time and genetic distance would be if sampling time was randomized. This allowed estimation of the substitution rate from the distance values. This test (presented in Algorithm 3.17) was implemented in R (R Development Core Team 2004) by the author.

Algorithm 3.17. *Substitution Rate Estimation*

Inputs:

S *a sequence of* n *nucleotide sequences.*

$q_d : S \mapsto Q_d$ *a property function giving date of isolation for nucleotide sequences in* S.

$q_m : S \times S \mapsto Q_m$ *a property function giving the absolute number of substitutions separating pairs of nucleotide sequences in* S *(generally determined from an alignment).*

Parameter c *the confidence level for the test.*

Parameter k *the number of iterations to use in the test.*

Outputs:

An estimate of the mean substitution rate for the nucleotide sequences in S.

Procedure:

Let $M(S, P)$ *be a function that takes the two arguments,* S *a sequence of nucleotide sequences of length* n *and the sequence* P*, a permutation of the numbers* $\{1, 2, \ldots, n\}$.
Let $S' = S|_P$.
The output of $M(S, P)$ *is a matrix where the element* $m_{i,j}$ *counts the number of times two nucleotide sequences* $s_a, s_b \in S$ *satisfy the condition* $q_m(s_a, s_b) = i$ *and* $|q_d(S'|_a) - q_d(S'|_b)| = j$. *Note that* q_m *is always applied to the sequence* S *while* q_d *is applied to* S'.
Let $m_{\max}$ *be* $\max(Q_m)$*, the largest difference in number of substitutions between two nucleotide sequences in* S.
Let $d_{\max}$ *be* $\max(|q_d(s_a) - q_d(s_b)|)$ *over all* $s_a, s_b \in S$*, the largest difference in date of isolation between two nucleotide sequences in* S.
The matrix produced by $M(S, P)$ *has dimensions* $m_{\max} \times d_{\max}$.

Step 1:
Record $D_0 = M(S, (1, 2, \ldots, n))$.

Step 2: Permutation Step
Let P *be sequence containing a random permutation of the integers* $\{1, 2, \ldots, n\}$.

Step 3: Sampling Step
Let $D' = M(S, P)$

Step 4: Iteration Step

Repeats steps 2 and 3 k times each time recording D' in the distribution D.

Step 5: Calculation Step
Associated with each entry $d_{i,j}$ in the matrix D_0 is a sequence of values $W_{i,j}$, constructed by taking the value of entry (i, j) from each of the matrices in distribution D. Construct a new matrix R in which $r_{i,j}$ is the percentile corresponding to c (the confidence level of the test) of $W_{i,j}$.

Step 6: Calculation Step
Let $E = D_0 - R$. Set any negative entries of E to zero.

Step 6: Calculate Results
Each entry $e_{i,j}$ in E represents the weight of a point at coordinate (i, j) in the x-y plane.
Perform a weighted linear regression on these points.
The slope m of the regression line gives the average substitution rate of the sequences in S in bases per year. To obtain a rate in substitutions per base per year divide m by the length of sequences in S. The R^2 value of the regression gives an idea of how accurate an estimate m is.

3.3.2.4 Selection at Individual Codon Sites Selection pressure at individual codon sites is measured using the *codeml* program from the PAML software package of Yang (1997) (version 3.14). Sites are classified into one of three selection classes; purifying, neutral and positive selection. *codeml* can not process stop codons so the one data set containing premature stop codons (RSV B G protein) is analysed twice. The first time all sequences are shortened to the length of the first stop codon. The second time sequences with premature stop codons are excluded and all other sequences analysed over their full length.

Alignments for this analysis are constructed by aligning the nucleotide alignments used

for the networks with GenBank nucleotide sequences for the protein being investigated. As with the initial alignments, Clustal X with manual editing is used for this step. If more than one sequence in the new alignment is identical then all instances of that sequence are removed and replaced with a single sequence representing all of the original sequences. We do this because for the purposes of this analysis, which ignores time data, these sequences are indistinguishable and keeping them increases the run time of the analysis dramatically. If a single nucleotide gaps occurs in the new alignment (probably due to sequencing error) the codon that gap occurs in is excluded from the analysis. To check the alignment is in the correct reading frame we check it against the the translation of the GenBank sequences and trim any partial codons from the start and end of the alignment.

codeml also requires the input of an evolutionary tree for the sequences under consideration. To calculate these trees we used maximum parsimony searches in PAUP* (Swofford 1998). For data which is amply sampled (any two nucleotide sequences can be joined by a sequence of nucleotide sequences, each only one substitution apart) maximum parsimony is a maximum likelihood estimator (Steel and Penny 2004). Our sample is not ample, however it is close, most links in the networks on the data are of length one. Thus we expect maximum parsimony to closely approximate maximum likelihood. Where possible, trees are found using the branch and bound algorithm to perform a full search of the tree space. For alignments with many taxa (such as RSV A G-Protein) where branch and bound becomes infeasible, heuristic searches are used. If multiple optimal trees are found, multiple trees are used in the analysis. In general the variations in tree topology for this data have little effect on the results of the analysis.

We also attempted to measure selective pressure at individual codon sites using the *sgi* program of Suzuki and Gojobori (1999) for comparative purposes; however it failed to work on our datasets. The *anc-gene* program used by the *sgi* program to derive ancestral sequences went into an infinite loop when attempting to find a binary parsimony tree (of which there are none) on the data. The source of the problem was confirmed by recompiling the *anc-gene* program with debug routines to check its progress.

3.3.3 Results.

3.3.3.1 New Zealand RSV B

3.3.3.1.1 Global Cluster Test. The results of the analysis of NZ RSV B are presented in the network shown in figure 3.14.

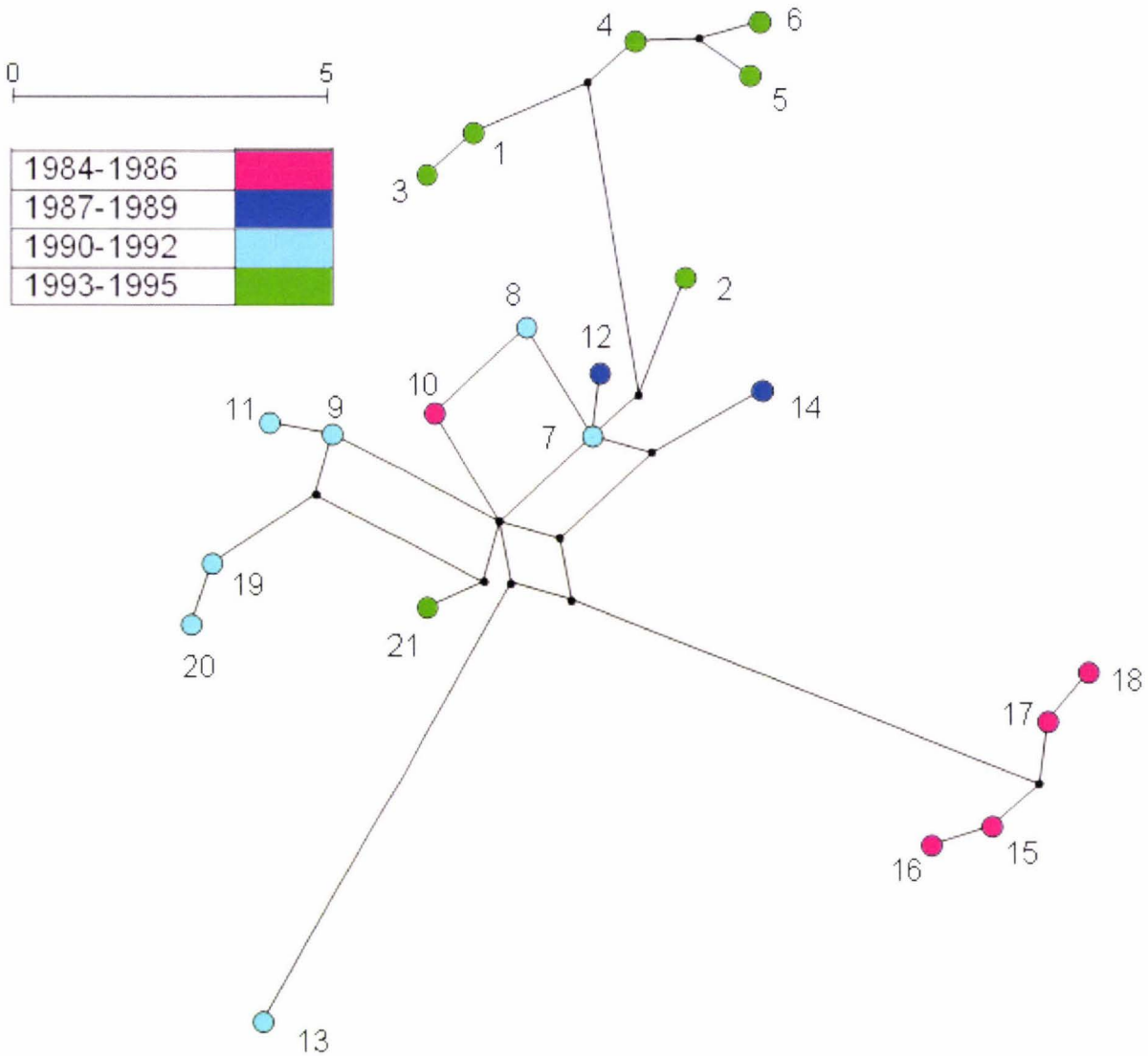


Figure 3.14: **RSV B in New Zealand G718 Data**. Entries are colour coded by mean date of isolation. Edges in the network represent splits in the data. This is a threshold 4 network. Information on how many sequences there are at each node and their date of isolation and patient number are given in section 5.1.

Visually this network seems to contain significant grouping by date of isolation, with groups at the top (1993-1995) middle (1990 - 1992) and right (1984-1986). Running the

global clustering test on this network confirms that there is clustering by date with 95% confidence at 10 000 iterations, telling us that most splits in the graph coincide with grouping sequences by date of isolation. A network containing all splits also passes the Global clustering test at 95% confidence. In general there was little difference between low conflict networks and networks containing all of the splits in the results of the global clustering test. The results for networks containing all splits will not be mentioned further in these analyses, except when they differ from those on the low conflict networks.
The RSV B sequences contain two sequences from the same patient stored in different cell lines. Both cluster at node 4, suggesting that differential storage of the viruses has caused little phylogenetic noise in RSV B.

3.3.3.1.2 Local Cluster Test. Figure 3.15 shows the results of running the local clustering test on the network in figure 3.14 for $\gamma = 3$ with 3 000 iterations per cluster radius.

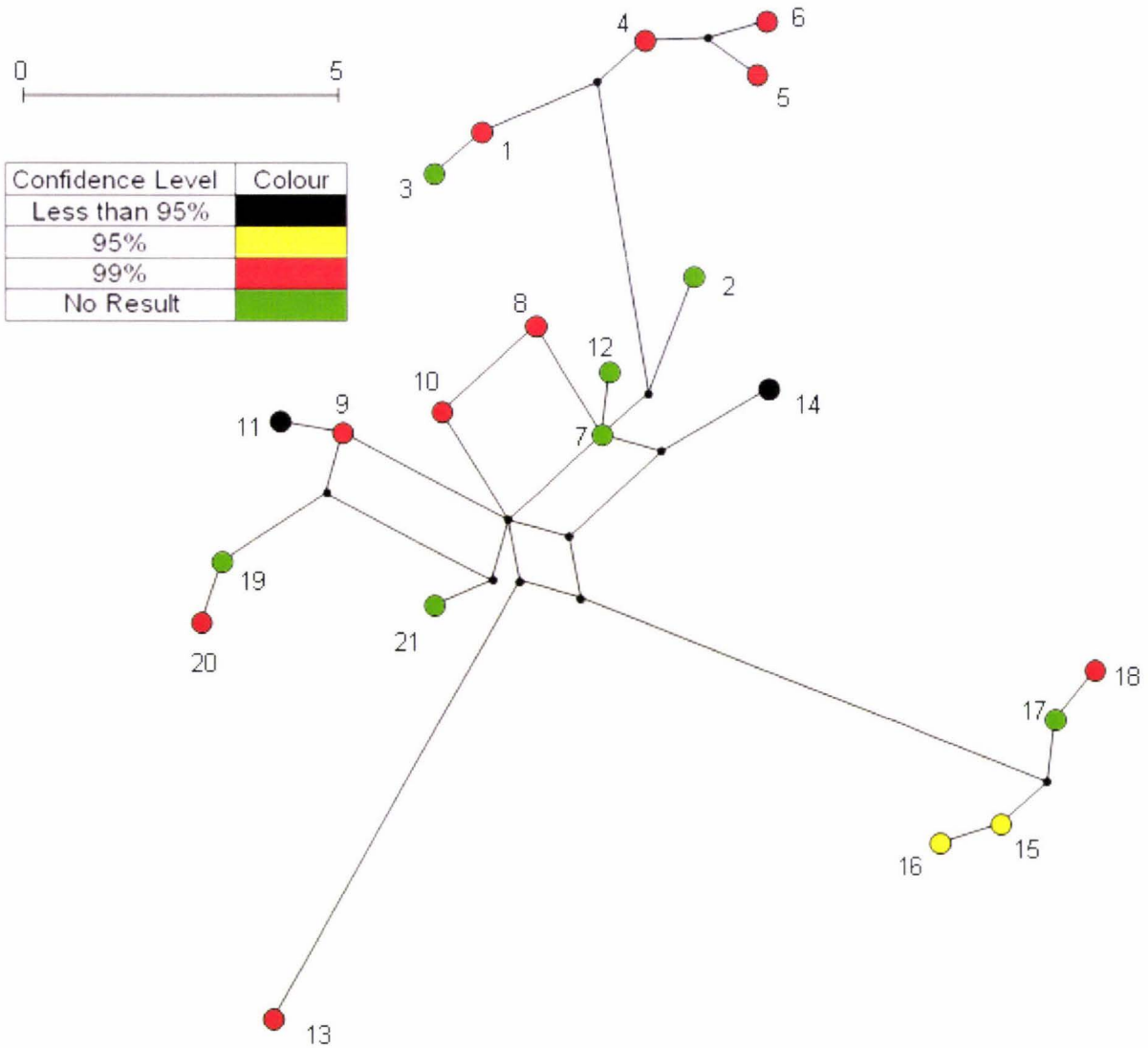


Figure 3.15: **Local clustering summary results for RSV B in New Zealand** for $\gamma = 1, 2, 3$ with 3000 iteration per cluster radii. The highest level of clustering observed at each node over the three cluster radii is displayed. This is the same network as that displayed in figure 3.14. Information on how many sequences there are at each node and their date of isolation and patient number are given in section 5.1.

In figure 3.15 we can see that each of the three broad clusters observed in figure 3.14 are displayed as clustering strongly at at least one level. The fact that the clustering responsible for a significant global test is caused by clustering at most nodes in the graph and

not merely in one location indicates that clustering by year is occurring consistently. This in turn implies that sequences are changing through time with old sequences becoming extinct. This is evidence for antigenic drift. Looking at both figures 3.14 and 3.15 we can also see that there is only one cluster in each time period. this suggests that there is only one major population of RSV B in our sample. As we do not have a comprehensive sample we can only conclude there is probably only one dominant subpopulation of RSV B.

3.3.3.1.3 Substitution Rate The substitution rate for RSV B in NZ is calculated to be 2 (2sf) substitutions per year per sequence in the G718 region. This translates to a rate of 0.05 (2sf) substitutions per base per year. This rate is probably higher than the rest of the virus as the region sequenced contains areas of significant variation. The R^2 regression statistic for this calculation is 0.7376 which means the input data is a fair approximation to a line. These calculations were done using Algorithm 3.17 implemented in R (R Development Core Team 2004).

3.3.3.1.4 Selection at Individual Codon Sites The G718 sequence covers both the F and G proteins. Selective pressure analysis was done on each of these proteins individually. Because the F and G have different levels of substitutions, the groupings of identical sequences for each are different. The sequences in each group are described in appendix 1 (section 5).

The F protein is the simplest case so we consider it first. The amino acid alignment for the F protein is shown in figure 3.16

```
                10        20        30        40        50
          ....|....|....|....|....|....|....|....|....|....|....|
Group_1   MELLIHRSSAIFLTLAINALYLTSSQNITEEFYQSTCSAVSRGYFSALRTGWYTI
Group_2   ..................................................
Group_3   ..................................................
Group_4   ..................................................
B188_90   ..................................................
Group_5   .D................................................
B094_89   ..............T.....................................
B278_93   ..............I.....................................
Group_6   ..................................................
B291_93   ........N.........................................
B299_94   ........N.........................................
```

Figure 3.16: **F-Protein Amino Acid 1 to 55 (C-terminus).** There are very few amino acid changes in this section of the protein. Before translation identical nucleotide sequences were replaced by a single sequence labeled Group_[Number]. Which sequences are in which group is described in section 5.2.

There is a single most parsimonious tree on this data that was found using the branch and bound algorithm in PAUP* (Swofford 1998). *codeml* results for this data show that all bases are under purifying selection. This is not surprising as this is the cross membrane section of the F-protein.

The G-protein Alignment is pictured in figure 3.17.

```
           260        270        280        290
           -|----|----|----|----|----|----|----|--
Group_1    HTIQQQSLHSTPENTPNSTQTPTASEPSTSNSTQTQSYA
B006_95    ....................................
B212_90    ....L........T.......................
B314_90    ....L...............................
B256_93    ..........I.......................N..H.
Group_2    ..T........S.....F..I............*N..H.
B154_84    ..T........S.....F..I............*N..H.
Group_3    ..T........S.....F..I............*K..H.
B188_90    ................................Y..QD
Group_4    .......F.........................G..H.
Group_5    ..............S..................S..H.
Group_6    ................................S..H.
B094_89    ..............F.................S..H.
B278_93    ..V.............................S..H.
Group_7    ..V........L....................G..H.
B294_93    ..V........L..........I..........G..H.
B316_93    ..V........L....................G..H.
Group_8    ..V.............................G..H.
```

Figure 3.17: **RSV B G-Protein Amino Acid 259 to 299 (N-terminus)**. 6 sequences in 3 groups (Group 2, Group 3 and B154_84) have premature stops at position 293 .6 amino acids before the end of the N-terminus of the protein. The two gaps are due to single nucleotide gaps in the nucleotide alignment for the affected codons. Before translation identical nucleotide sequences were replaced by a single sequence labeled Group_[Number]. Which sequences are in each group is described in section 5.3.

Notice the three sequences with premature stop codons at position 293 in figure 3.17. Premature stop codons of this nature have been observed in previous studies (Martinez *et al.* 1999) so their appearance here is not unprecedented. To allow full analysis of all sequences this alignment was tested twice. The first analysis looked at all sequences from position 259 to 292, the position preceding the premature stop codons. The second analysis looked at the full length (259 to 299) of all sequences which do not have premature stops.

For the first analysis there is a single most parsimonious tree found by branch and bound. The results of this analysis classified the amino acid sites in the interval 259 to 292

into those under purifying and neutral selection at a ratio of 53:47 although no site was firmly placed in either category at 95% confidence. Sites were instead assigned nonzero probabilities of being in either category. No sites were found to have any support above 95% for being under positive selection.

For sequences without premature stop codons there are 49 most parsimonious trees found by branch and bound in PAUP* (Swofford 1998). All 49 trees were used for the analysis, results differed little between the different trees. This analysis assigned a much larger proportion of the sites previously analysed to the purifying selection category. One site (294) was identified as being under positive selection with 95% confidence. This is the site directly preceding the premature stops. This placement suggests the premature stop may confer advantage on the virus by eliminating the amino acids after position 294.

Previous studies have found evidence for positive selection at several places on the G protein of RSV A and B (Woelk and Holmes 1998). Unfortunately the sequence we have covers a different portion of the G protein to that sequenced in Woelk and Holmes (1998) so a full comparison is not possible. Comparing the sequence common to both studies shows the positively selected residue at 294 was also noted in Woelk and Holmes (1998).

3.3.3.2 New Zealand RSV A

3.3.3.2.1 Global Cluster Test The results for RSV A are much more complex than those for B.

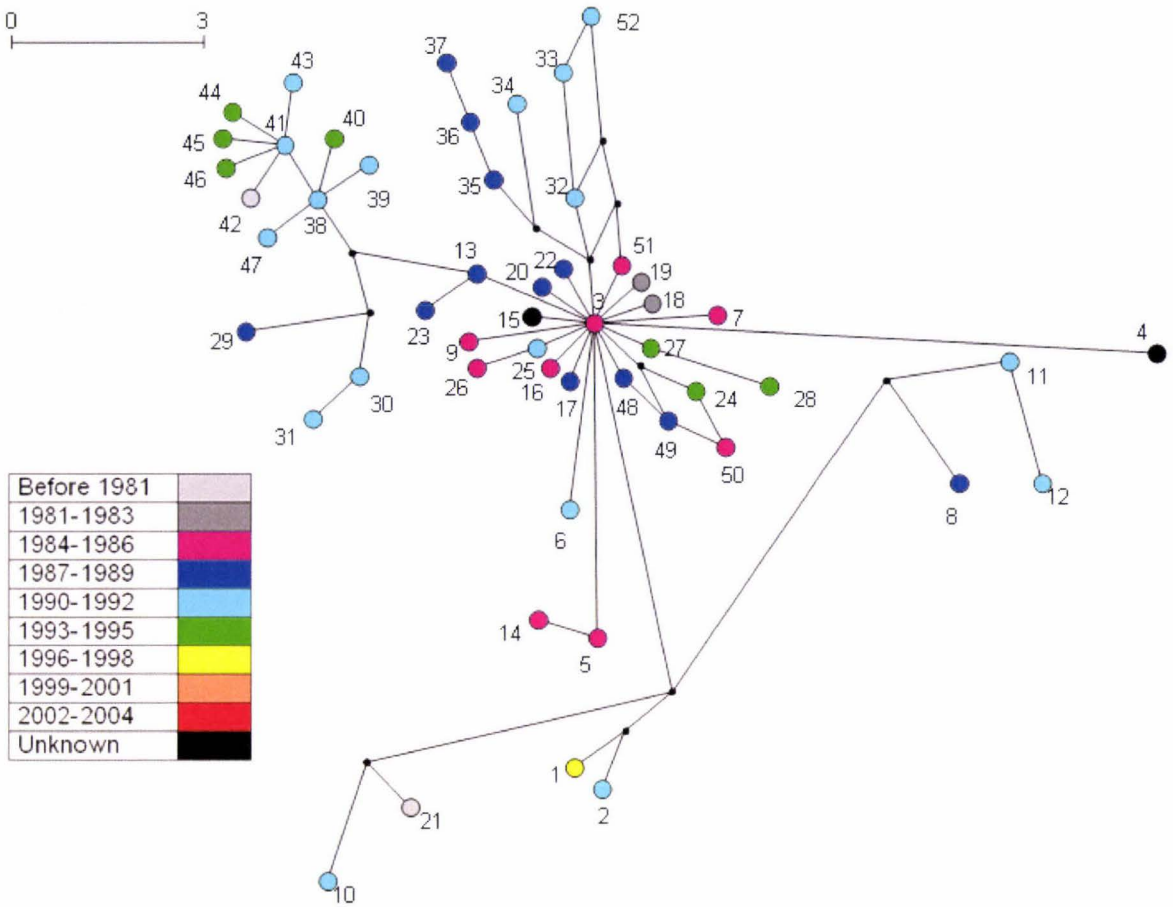


Figure 3.18: **RSV A In New Zealand FG-A Network.** Nodes are colour coded by mean year of isolation. Edges represent splits in the data. This is a threshold 10 network. Note the placement of very old sequences at nodes 21 and 42 among much younger sequences. Information on how many sequences there are at each node and their date of isolation and patient number are given in section 5.4.

The clustering by date in the RSV A network (figure 3.18) is not so obvious to the eye as for RSV B, however one can still see two large clusters from approximately the same time period. The global clustering test indicates that there is clustering by year on this graph at 95% percent confidence. We will look further at this clustering in the local clustering test for this graph. An interesting feature of this graph is the placement of the pre-1981

strains. There are two nodes in the graph representing pre-1981 strains from the 60's (node 21) and 70's (node 42). The oldest of these nodes, node 21, (which contains our oldest sequences from 1967) is highly diverged from the two main clusters, however it is closely related to virus from 1992-1993 season. Node 42 containing a very good quality sequence from 1973 is part of a major clusters around node 41. We expect the oldest sequences to be highly diverged, however it is interesting that the 1973 virus is similar to viruses from 1990-1992. There are several possible explanations for this. One is that we are not using enough sequence data and if were able to sequence more genes we would find greater divergence between these sequences. However given the greatest variation between RS Viruses occurs in the G protein this explanation seems unlikely. Another explanation is that the 1973 virus has stayed at a low level in the population and a close relative has hence emerged from it at a later date. Or the virus could have reinvaded from overseas. To test the local persistence hypothesis we would need to look for further evidence of older strains within the population. Unfortunately pursuing this possibility is outside the scope of this thesis, however it would be an interesting avenue for further investigation. The reinvasion hypothesis can be investigated by comparing international and New Zealand strains which we do in section 3.4.

The RSV A sequences contain several sequences from the same patient stored in different cell lines. All sequences from the same patient cluster at the same node, suggesting the differential storage of the viruses has caused little phylogenetic noise in RSV A.

3.3.3.2.2 Local Cluster Test We can gain more information about the clustering of RSV A by looking at the local cluster test results in figure 3.19.

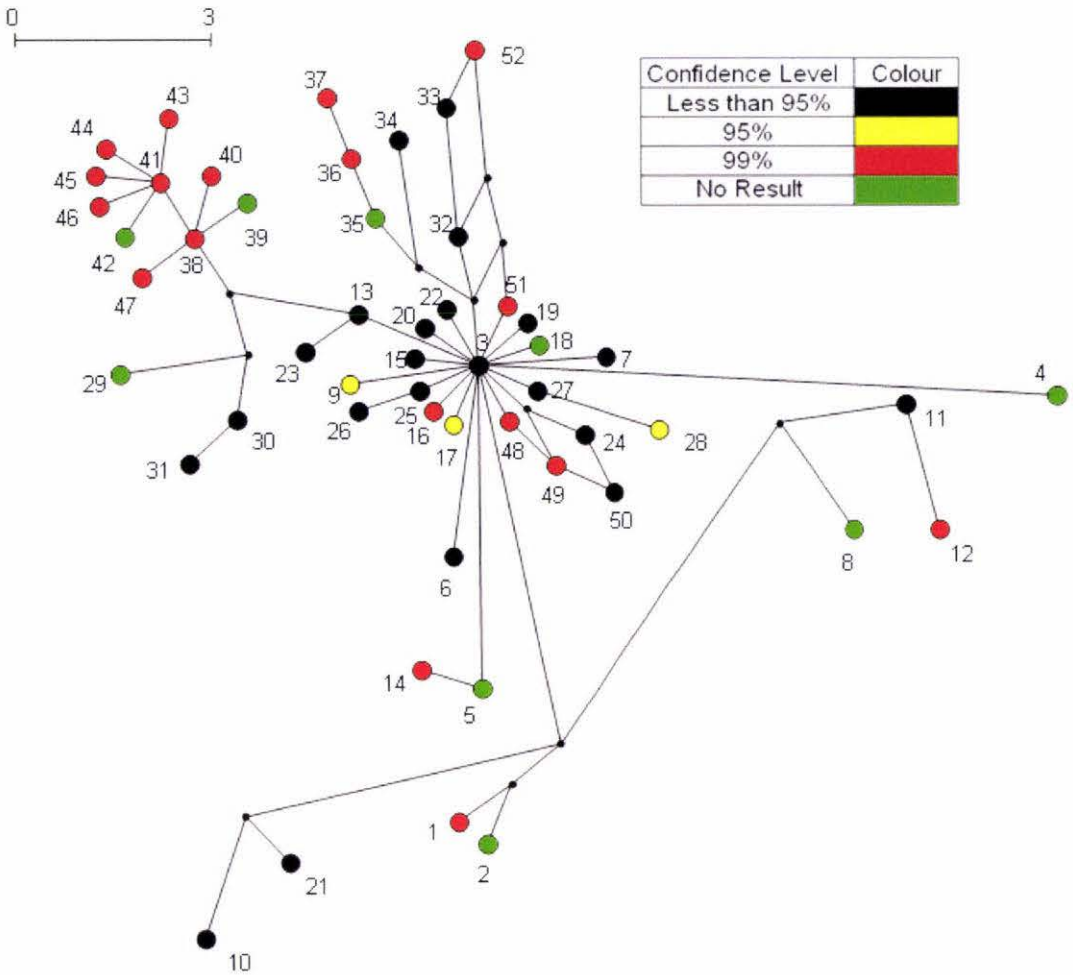


Figure 3.19: **Local clustering to summary results for RSV A in New Zealand** for $\gamma = 1, 2, 3$ after 10000 iterations. The highest level of clustering observed at each node over the three cluster radii is displayed. This is the same network as displayed in figure 3.18. Information on how many sequences there are at each node and their date of isolation and patient number are given in section 5.4.

The local clustering test results for RSV A tell us that the cluster centered on node 41 is clustered strongly. Clustering by year around the central node 3 is very weak. The are several vertices in distinct groups measured as clustering by date from the 1990-1992 sampling period. The clustering observed here is much weaker clustering than we observed for RSV B. The wide spread of clusters from the same time period indicates that there

are several dominant subpopulation of RSV A, perhaps with different founders.

3.3.3.2.3 Substitution Rate The substitution rate calculation for RSV A has very low support with an R^2 regression statistic of only 0.3536. This means there is poor support for the data being modeled by a single line. The calculated value is much lower than for RSV B with an estimated 0.87 (2sf) substitutions per year. This translates to a substitution rate of 0.014 (2sf) bases per site per year. These calculations were done using Algorithm 3.17 implemented in R (R Development Core Team 2004). To understand the low confidence level of this estimate we can look at a plot of time versus substitutions as in figure 3.20.

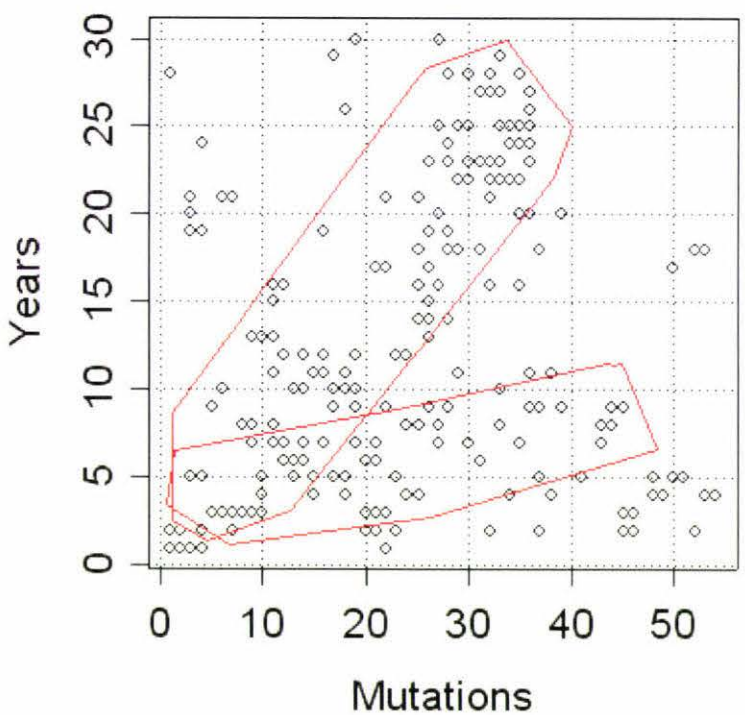


Figure 3.20: **Graph of Number of substitutions vs Number of years separating samples for RSV A.** Each point represents a comparison between two sequences.

In Figure 3.20 we can see that the data points form two overlapping groups. The positioning of these groups suggest the data points could be interpolated by two lines; which would indicate the presence of two subpopulations of RSV A in New Zealand with different

substitution rates. This pattern would also support a hypothesis where RSV evolution is cyclic, with the virus evolving back towards old strains once immunity to those old strains has disappeared from the population. These hypothesis are tentative and more evidence is needed to confirm them. It will be interesting to investigate these hypotheses once the data for 2003 season becomes available.

3.3.3.2.4 Selection at Individual Codon Sites The FG-A sequence covers only the G protein so we only look at the G-Protein for RSV A. The amino acid alignment for the New Zealand RSV A is too large to fit on a single page so it is split into two contiguous pieces shown in figures 3.21 and 3.22.

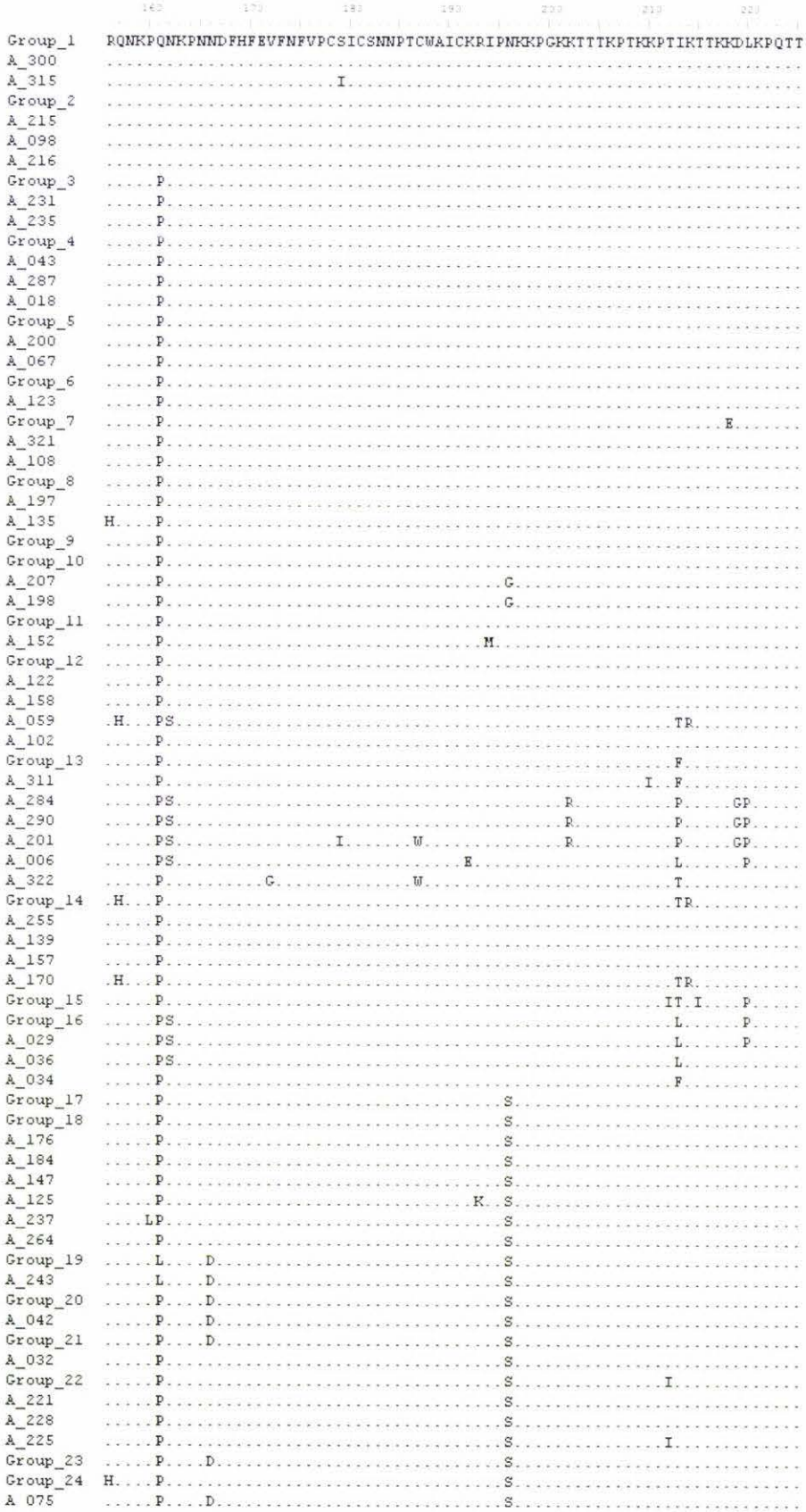


Figure 3.21: **New Zealand RSV A G-Protein amino acids 156 to 225.** This area includes the conserved area of the G-Protein. Before translation identical nucleotide sequences were replaced by a single sequence labeled Group_[Number]. Which sequences are in which group are given in section 5.5.

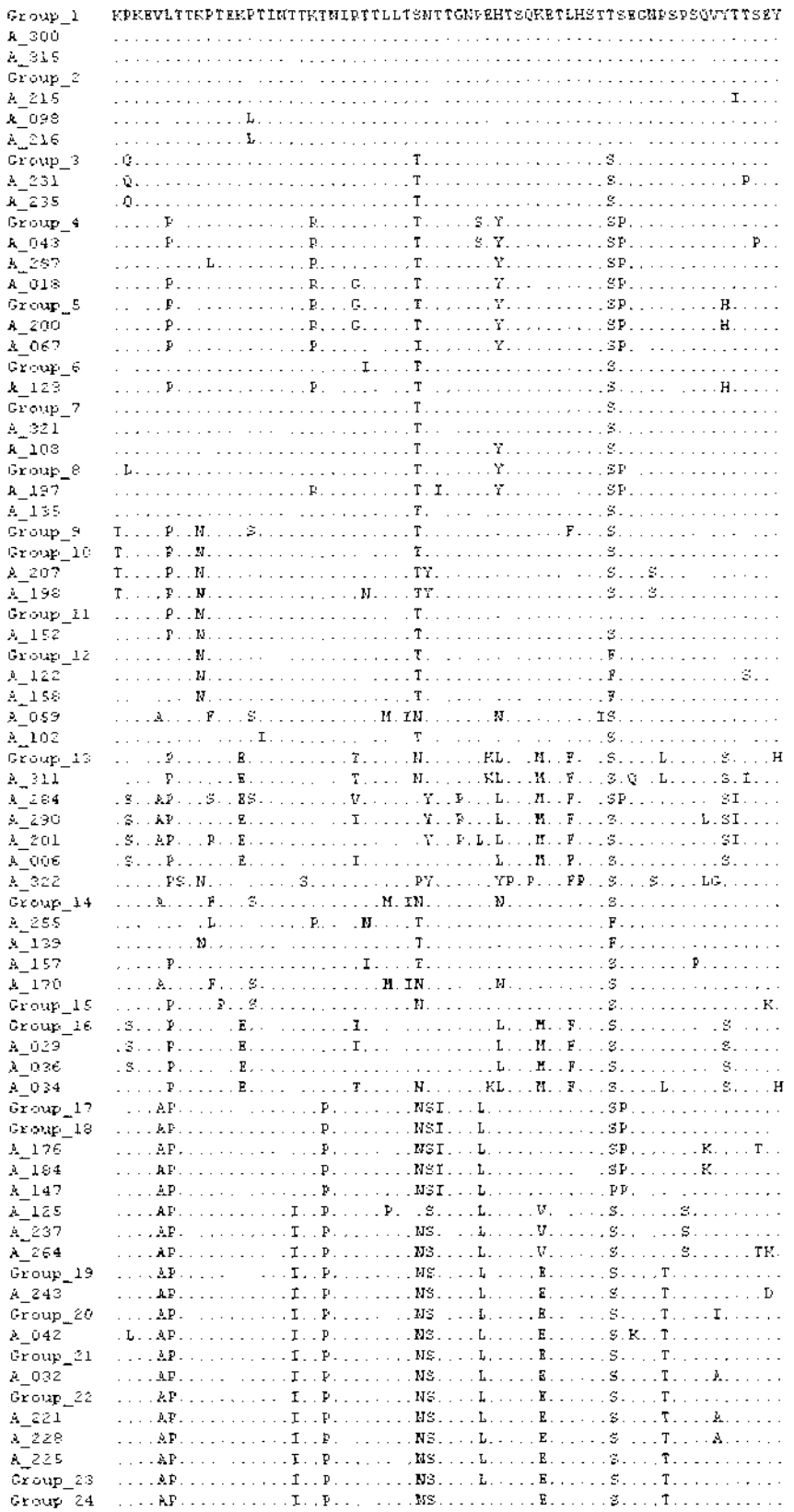


Figure 3.22: **New Zealand RSV A G-Protein amino acids 226 to 290.** This area includes some of the hypervariable region at the N-terminus end of the G-protein. Before translation identical nucleotide sequences were replaced by a single sequence labeled Group_[Number]. Which sequences are in which group are given in section 5.5.

This data set is too large to use a branch and bound search on. A heuristic search was run in PAUP* (Swofford 1998) to find a set of most parsimonious trees on this data. 15 664 optimal trees were found. We attempted to prove these trees were optimal using the MinMax squeeze method (Holland *et al.* In Press). Unfortunately the ratio of number of sites to estimated upperbound was too high so the optimality of these trees was unable to be proved. The fact we are looking at population data means it is unlikely the data can be accurately represented by a binary tip labeled tree (Holland *et al.* In Press).

Analysing just one of these trees on a 2Ghz machine takes 4 hours so a small sample of trees was selected to analyze from those available. It should be noted that a quick inspection of these 15 664 trees using tree view showed them to be fairly similar to one another. 10 trees were selected in such a way as to get a wide variation. The analysis run on these 10 trees showed little variation in results from tree to tree.

73% of sites were classified as under purifying selection while the remaining 27% were classified as being under positive selection. 16 sites were found to be under positive selection in all of the trees sampled: these were sites 213, 227, 231, 235, 239, 249, 255, 261, 263, 267, 274, 275, 284, 285 and 289. The large number of sites and their concentration in the same area of the protein implies the area of the G-protein from 213 to 289 is under strong positive selective pressure in RSV A. This is consistent with an antigenic drift hypothesis.

Previous studies have found evidence for positive selection at several places on the G protein of RSV A and B (Woelk and Holmes 1998). The portion of RSV A sequenced is much closer to the area used in Woelk and Holmes (1998) than our B sequence. Comparing the sequence common to both studies shows that the sites found to be positively selected on A in Woelk and Holmes (1998) are also found as positively selected here. There seem to be many more sites designated as positively selected in this study than in Woelk and Holmes (1998). Interestingly some of the sites found to be under positive selection on A in this study are found to be under selection on B but not on A in Woelk and Holmes

(1998) (sites 231 and 235 for example). These differences could be caused by methodology (such as the different sequence portions used) or by differences in the viruses themselves.

3.3.4 New Zealand Conclusions.

RSV B evolution in New Zealand shows some evidence of antigenic drift. The network data is consistent with this hypothesis and areas of the G-protein are under positive selection. The areas of the G-protein under strong positive selective pressure in RSV A are not available in the G718 data. Investigating the FG-B data should give more information on antigenic drift in RSV B.

RSV A also shows evidence of antigenic drift. The evidence in the networks is less decisive than for RSV B. However the large number of sites under positive selection strongly supports an antigenic drift hypothesis.

Selective pressure on RSV A and B in New Zealand is similar to that found in other studies. The cause of the differences between this and previous studies in this respect is unclear.

In general there is evidence RSV does undergo antigenic drift.

There appear to be few strains of RSV B present in New Zealand at one time. Given the nature of the sampling this conclusion must be tentative however it will be interesting to investigate this hypothesis in the prospective samples (which are unfortunately not available until after the projected completion date of this thesis).

There have been several dominant subpopulations of RSV A in New Zealand, possibly with different founders. This is a firm conclusion. A more representative sample could only provide evidence for more strains.

Distinct subpopulations of RSV in New Zealand may be mutating at different rates. This

is a tentative conclusion.

There is no evidence that differential storage of samples has caused significant phylogenetic noise.

3.4 International comparisons.

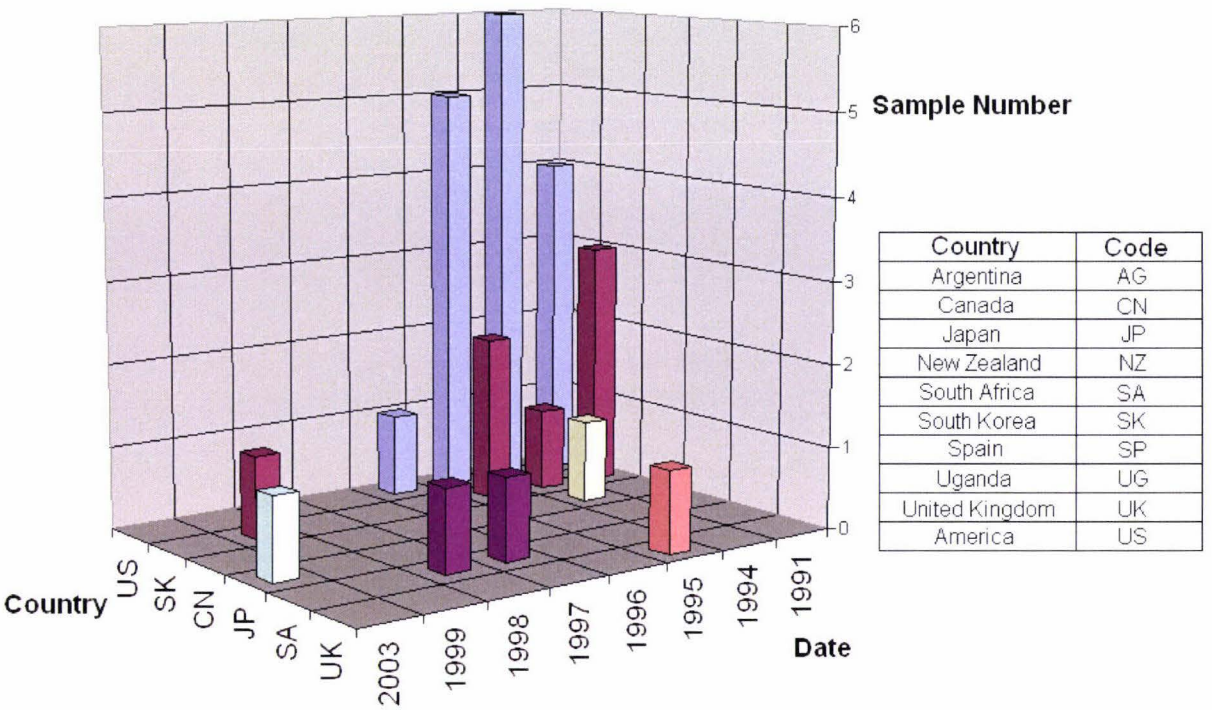
3.4.1 Introduction

Through the comparisons of international and New Zealand historical RSV samples we can come to useful conclusions about the rate of transmission of RSV into New Zealand from outside the country. New Zealand has a higher rate of severe RSV cases than other countries so it would be interesting to see if this is due to a specific New Zealand strain. This would be indicated by strong clustering of NZ strains together. To understand how RSV flows into NZ we conducted an analysis of the clustering of the samples both by age and by country.

3.4.2 Methods

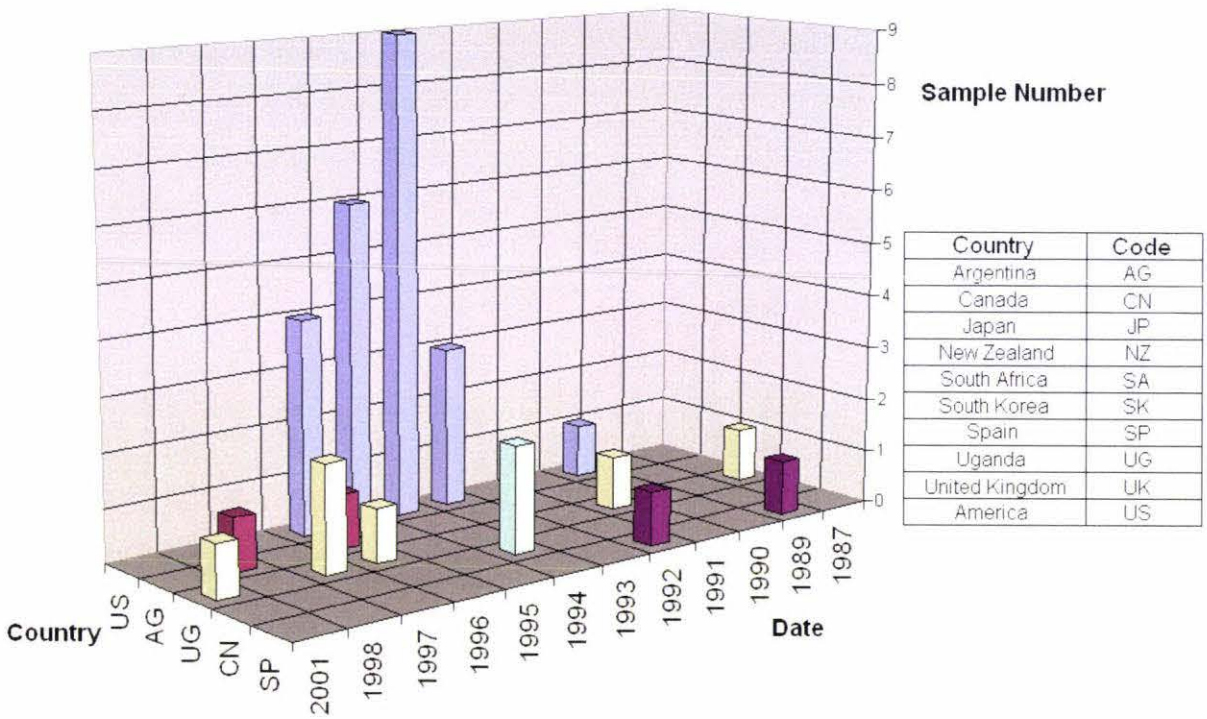
To make International comparisons both NZ and international sequences were aligned and plotted on the same network. A selection of immunologically interesting international sequences from a range of years were selected from GenBank by Dr Joanna Kirman of the Malaghan Institute[3] for use in these comparisons. The countries and dates of the samples used are displayed in figures 3.23 and 3.24.

[3]Malaghan Institute for Medical Research *http://www.malaghan.org.nz*



Country	Code
Argentina	AG
Canada	CN
Japan	JP
New Zealand	NZ
South Africa	SA
South Korea	SK
Spain	SP
Uganda	UG
United Kingdom	UK
America	US

Figure 3.23: **Sample Date and location for RSV B International Samples**



Country	Code
Argentina	AG
Canada	CN
Japan	JP
New Zealand	NZ
South Africa	SA
South Korea	SK
Spain	SP
Uganda	UG
United Kingdom	UK
America	US

Figure 3.24: **Sample Date and location for RSV A International Samples.**

The sequences were tested to see how they clustered by location and date. Sequences were tested for clustering by date and location independently. There were four possible results of these tests:

1. No Clustering
2. Clustering by date only.
3. Clustering by location only.
4. Clustering by date and location.

The absence of clustering would tell us little. Clustering by date only would tell us that the viruses group independently of location. This in turn would indicate that the virus spreads very quickly around the world resulting in a homogeneous worldwide RSV population. Conversely clustering only by isolation date would indicate that each location had its own

unique strain with very little mixing taking place between geographical subpopulations. Clustering by both date and location would indicate a situation where there are many small subpopulations with regular mixing.

Clustering is tested using the global and local clustering algorithms detailed in section 3.3.2. Clustering by year is measured in an identical method to that detailed in section 3.3.2. Clustering by geographical location however requires a cluster measure on categorical data. The method we already have for continuous data will clearly not work here so a different cluster measure was used

The discrete cluster measure $C_D(s)$ is defined in Definition 3.18

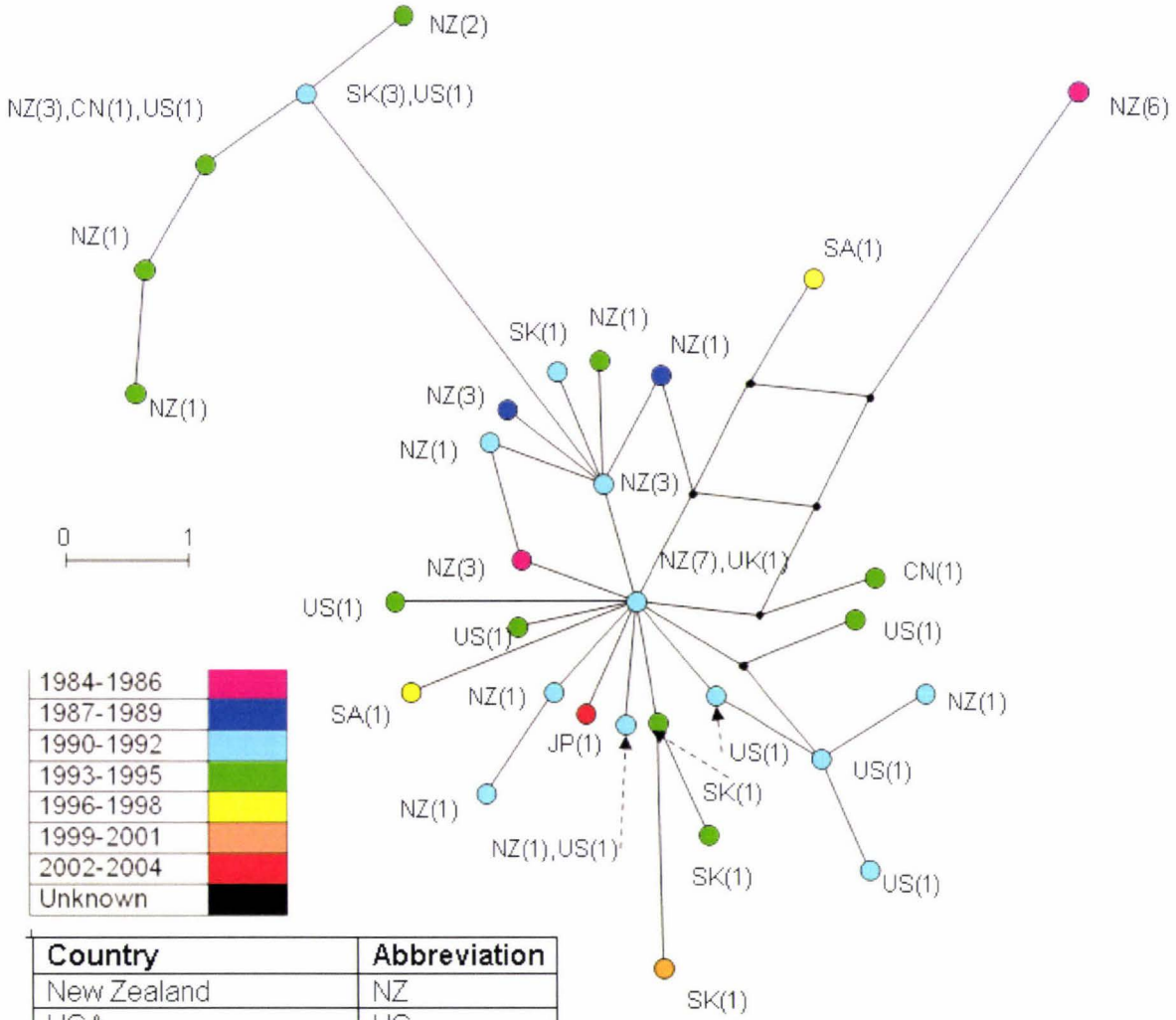
Definition 3.18. *Discrete Cluster Measure:* $C_D(S)$.
Let S *be a sequence.*
Let $x \in X$ *be the most numerous element in* S. *If there is more than one maximally numerous element in* S *let* x *be picked randomly from these.*
Let S_x *be the number of items in* S *which equal* x.

$$C_D(S) = \frac{S_x}{|S|} \tag{3.1}$$

Defining $C_D(S)$ *as in* (3.1) *gives a value between 0 and 1 for the cluster measure with a value of 1 indicating strongest clustering*

3.4.3 Results.

3.4.3.1 International RSV B The network constructed from RSV B International comparisons is shown in figure 3.25.



Country	Abbreviation
New Zealand	NZ
USA	US
Japan	JP
South Africa	SA
South Korea	SK
Canada	CN
United Kingdom	UK

Figure 3.25: **RSV B International comparisons.** This is a threshold 4 network. Dotted lines with arrow heads connect vertex labels to their vertices when necessary.

There appears to be weak clustering by date in figure 3.25. This network passes the global clustering test at 95% confidence. If we look at all splits clustering is observed at 99% confidence. However looking at geographical distribution it is hard to see much pattern by

eye. This dataset fails a global clustering test by country for all splits and for all subsets of splits. Clustering only by date supports a model where RSV B spreads quickly into New Zealand from other countries. There is no support for New Zealand having a unique subpopulation of RSV B. We need to be cautious about conclusions from this data due to the small alignment overlap between NZ and international sequences. However the wide distribution of International sequences throughout the graph is unlikely to be counteracted by more sequence data. We would expect more sequence data to accentuate rather than ameliorate observed differences. The local clustering test results (figures 3.26 and 3.27) for this data are also interesting.

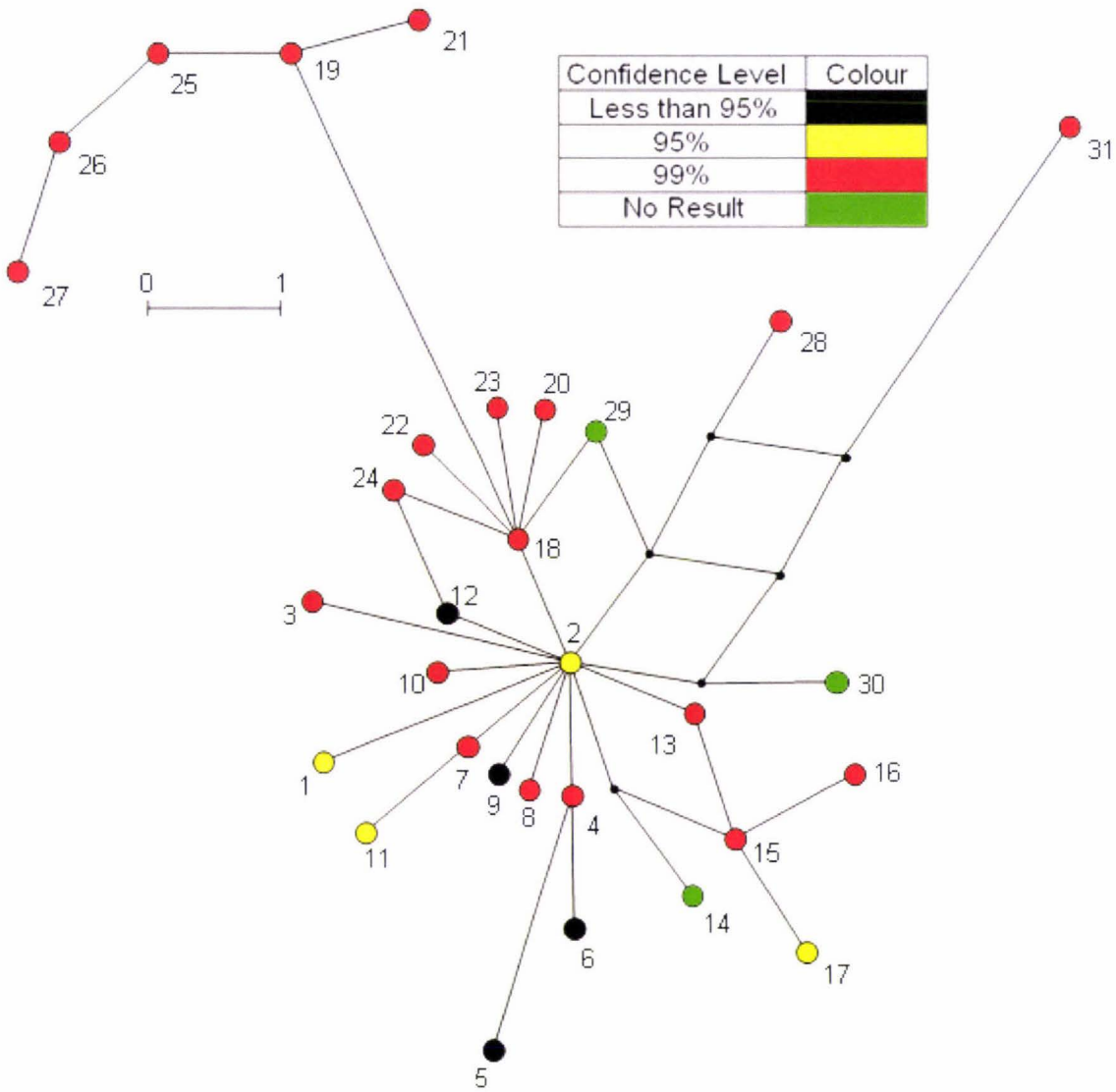


Figure 3.26: **Local Clustering by Date for the International RSV B alignment.** Summary of results over $\gamma = 1, 2, 3$ with 3000 iterations per level. The highest level of clustering observed at each node over the three clustering levels is displayed. This is the same network as displayed in figure 3.25.

Figure 3.26 shows several centers of clustering by date on the International B Network and supports the results of the global clustering test that RSV B sequences tend to group by year internationally.

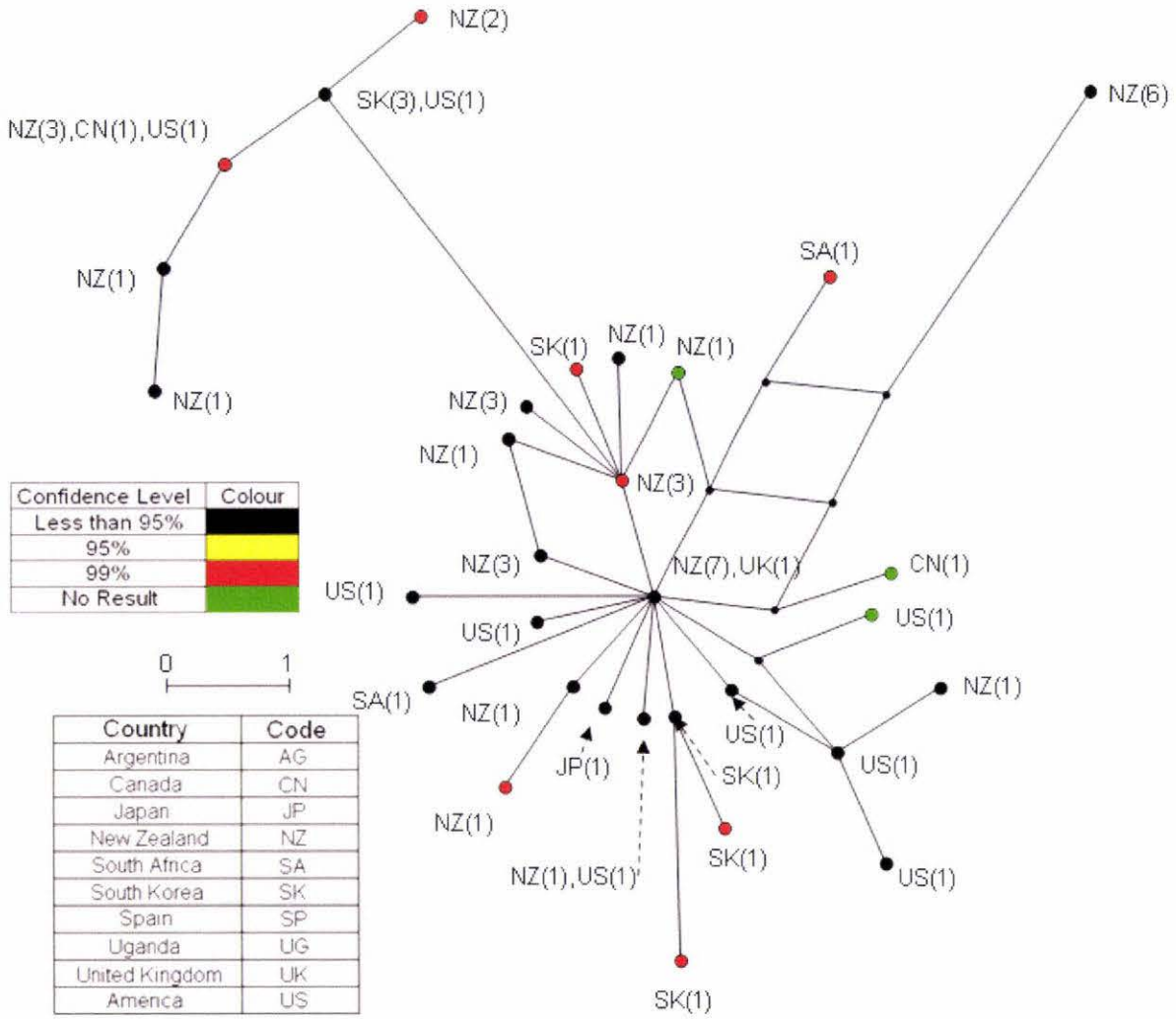


Figure 3.27: **Local Clustering by Country for the International RSV alignment.** Summary of results over $\gamma = 1, 2, 3$ with 3000 iterations per level. The highest level of clustering observed at each node over the three clustering levels is displayed. This is the same network as displayed in figure 3.25.

In figure 3.27 we can see there are few significant topological clusters by country. There is very little evidence in the data for any local strains of RSV B. We can also see that NZ RSV B does not seem to be significantly different from RSV B in other countries, else we would expect it to form topological clusters. It is interesting to note that the little

clustering that does occur seems to do so independently of the sampling date. The two South Korean samples clustered at the bottom of figure 3.27 are not centers of clusters in figure 3.26 (nodes 5 and 6). However the South Korean and New Zealand samples near the center of the graph are also clustering by year in figure 3.26 (nodes 18 and 23).

3.4.3.2 International RSV A The network for International RSV A is shown in figure 3.28.

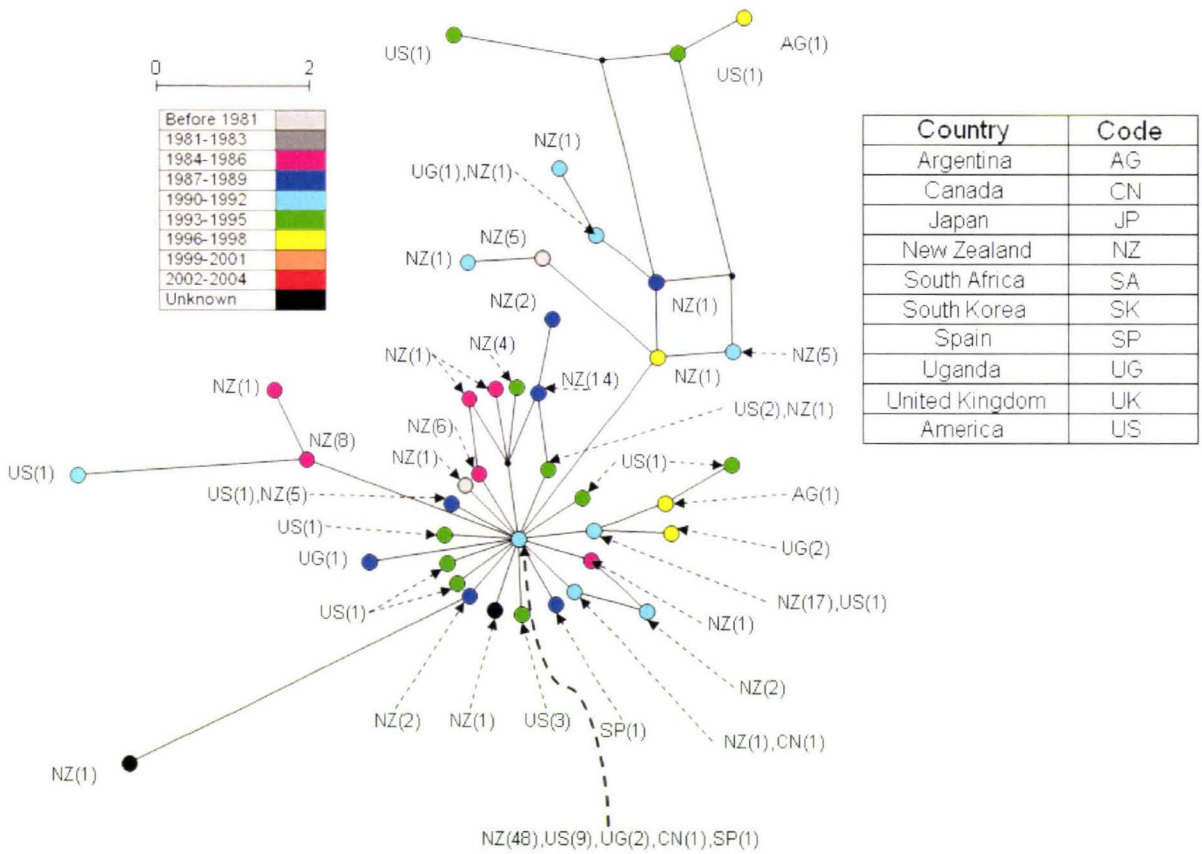


Figure 3.28: **RSV A International comparisons.** This is a threshold 10 network.

We can see little clustering by date on this network. The network fails a test for significant clustering by date at 95% confidence. If we consider all the splits in the data, the data barely passes a global test for clustering by date at 95% confidence. Visual inspection of

this network shows that sequence from different countries seem to be dispersed through out the graph. Results of the global clustering test by country show this alignment fails tests for clustering by country for all splits and for the displayed network. Though the evidence is weaker than for RSV B this shows support for a model where RSV A spreads quickly into New Zealand from other countries. There is no support for New Zealand having a unique subpopulation of RSV A.

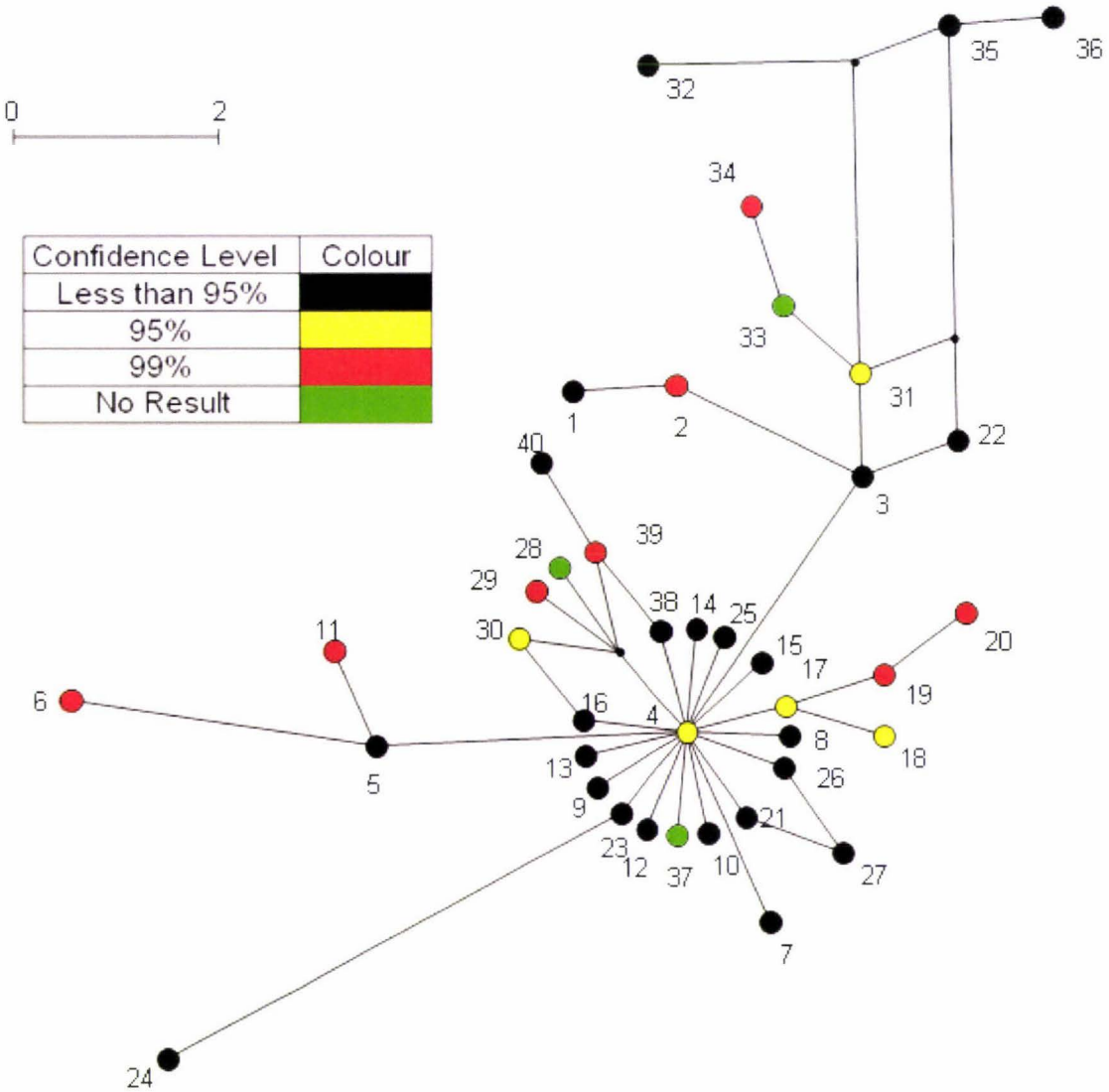


Figure 3.29: **Local Clustering by Date for the International RSV B alignment.** A summary of results $\gamma = 1, 2, 3$ with 3000 iterations per level. The highest level of clustering observed at each node over the three clustering levels is displayed. This is the same network as displayed in figure 3.28.

Figure 3.29 shows us where on the network the weak clustering by year occurs. The large unresolved clump of sequences around node 4 makes significant clustering of any kind unlikely.

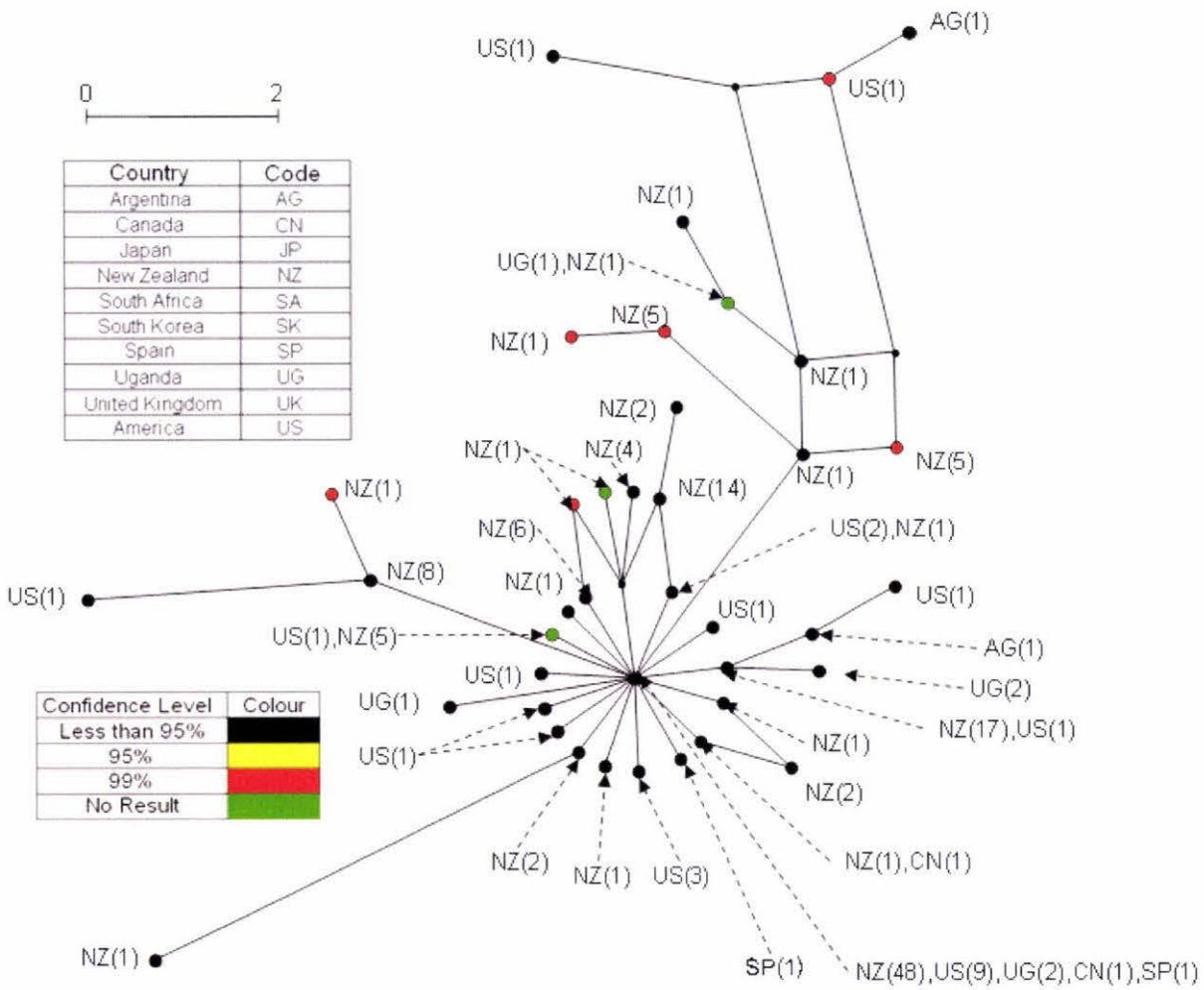


Figure 3.30: **Local Clustering by Country for the International RSV B alignment.** A summary of results over $\gamma = 1, 2, 3$ with 3000 iterations per level. The highest level of clustering observed at each node over the three clustering levels is displayed. This is the same network as displayed in figure 3.28.

Figure 3.30 shows there is little topological clustering by country for the International RSV A sequences. The almost complete lack of topological clustering gives good support to the hypothesis that New Zealand does not have significantly different strains of RSV A than other countries.

The lack of clustering by country for RSV A supports the reinvasion explanation for old

samples of RSV A being similar to new (section 3.3.3.2.1) but a more comprehensive international and local sampling would be required to make any firm conclusions in this regard.

3.4.4 International Conclusions.

RSV is spread to New Zealand on a fairly regular basis, and NZ strains are not significantly different from overseas strains. Though the New Zealand sample being used for comparison is not a representative sample, were it true that New Zealand had strains of RSV significantly different from overseas strains, we would expect a small sampling of the New Zealand distribution would also contain viruses significantly different from viruses overseas. Thus there is good support for the conclusion that New Zealand does not have strains of RSV that are significantly different from those in other countries. That New Zealand and other countries have similar strains suggests those strains are mixing on a regular basis. This means the greater number of hospitilisations due to RSV bronchiolitis in New Zealand must be contingent on factors other than the strain of the virus that we have in New Zealand.

3.5 Final Conclusions

The evolutionary behavior of RSV A and RSV B does not seem to be qualitatively different, although RSV A is far more diverse than RSV B in this sample. If this diversity is representative of the entire RSV population then it is a possible explanation for the greater success of RSV A as an infective agent.

RSV in New Zealand shows some evidence of being in a process of antigenic drift. This is interesting as RSV is not thought to give a strong long term immunity, which is commonly the factor behind antigenic drift.

The strains of RSV present in New Zealand and the strains present in other countries do not differ significantly. The greater frequency of hospitalization with RSV bronchiolitis in New Zealand must be due to factors other than virus strain.

4 References

Bandelt, H-J. and Dress, A.W.M., 1992, *Split Decomposition: A new and useful approach to phylogenetic analysis of distance data.*, Molecular Phylogenetics and Evolution, 1:242-252.

Bergstrom, C. T., Lachmann, M., 2003,*The Red King effect: When the slowest runner wins the coevolutionary race.*, PNAS, 100:593-598.

Biebricher, C.K., Eigen, M. and Luce, R, 1981, *Kinetic analysis of template-instructed and de novo RNA synthesis by QBeta replicase.*, J. Mol. Biol., 148:391-410.

Boerlijst, C., Hogeweg, P., 1991, *Spiral wave structure in pre-biotic evolution: Hypercycles stable against parasites.*, Physica D: Nonlinear Phenomena, 48:17-28.

Cane, P.A. and Pringle, C. R., 1995, *Evolution of Subgroup A Respiratory Syncytial Virus: Evidencefor Progressive Accumulation of Amino Acid Changes in the Attachment Protein.*, Journal of Virology, 69:29182925.

Collins, C.L. and Pollard, A.J, 2002, *Respiratory Syncyitial Virus Infections in Children and Adults.*, Journal of Infection, 45:10-7.

Drummond, A. and Rodrigo, A.G., 2000, *Reconstructing Genealogies of Serial Samples Under the Assumption of a Molecular Clock Using Serial-Sample UPGMA*, Molecular Biology and Evolution, 17:1807-1815.

Eigen, M. and Schuster P., 1977, *The hypercycle. A principle of natural self-organization. Part A: Emergence of the hypercycle.*, Naturwissenschaften, 64:541-565.

Eigen, M., 1971 *Self Organization of Matter and the Evolution of Biological Macromolecules.*, Naturwissenschaften, 58:464-523.

Fisher, R. A., 1930, *The Genetical Theory of Natural Selection*, Claredon Press, Oxford.

Hall, C.B., Walsh, E.E., Schnabel, K.C., Long, C.E., McConnochie, K.M., Hildreth, S.W. and Anderson, L.J., 1990, *Occurrence of groups A and B of respiratory syncytial virus over 15 years: associated epidemiologic and clinical characteristics*

in hospitalized and ambulatory children., J Infect Dis., 162:1283-90.

Higgins, J. J., 2004, *Introduction to modern non-parametric statistics.*, Thomson/Brooks-Cole, Pacific Grove, California.

Holland, B. R., Huber, K. T., Penny, D., and Moulton, V., In Press, *The Min-MaxSqueeze: Guaranteeing a Minimal Tree for Population Data.*, Mol. Bio and Evo., 22:235-242.

Huber, K.T., Langton, M., Penny, D., Moulton, V. and Hendy, M., 2002, *Spectronet: A package for computing spectra and median networks.*, Applied Bioinformatics, 1:159-161.

Janeway, C.A. et. al, 2001, *Immunobiology 5.*, Garland Publishing, NY, pg 427.

Lawrence, M.S. and Bartel, D.P., 2003, *Processivity of ribozyme-catalyzed RNA polymerization.*, Biochemistry, 42:8748-8755.

Martinez, I., Valdes, O., Delfraro, A., Arbiza, J., Russi, J. and Melero, J.A., 1999, *Evolutionary pattern of the G glycoprotein of human respiratory syncytial viruses from antigenic group B: the use of alternative termination codons and lineage diversification.*, J. Gen. Virol., 80:125 - 130.

Maynard Smith, J., 1982, *Evolution and the theory of games.*, Cambridge University Press.

Nowak, M.A., Sasak, A., Taylor, C. and Fudenberg, D., 2004, *Emergence of cooperation and evolutionary stability in finite populations*, Nature, 428:646 - 650.

Onions, C.T., Ed., 1959, *The Shorter Oxford English Dictionary*, Third edition, Oxford Clarendon Press.

Osborne, M. J., 1994 *A course in game theory.*, MIT Press, Cambridge, Mass.

Polak, M. J., 2004, *Respiratory Syncytial Virus (RSV) Overview, Treatment and Prevention Stategies.*, Newborn & Infant Nursing Reviews, 14:15-23.

R Development Core Team, 2004, *R: A language and environment for statistical computing.*, R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org.

Semple, C. and Steel, M., 2003, *Phylogenetics*, Oxford University Press.

Spiegelman, S., Haruna, I., Holland, I.B., Beaudreau, G. and Mills, D.R, 1965,

The synthesis of a self-propagating and infectious nucleic acid with a purified enzyme., PNAS, 54:919-927.

Steel, M. A. and Penny D. , 2004, *Two further links between MP and ML under the Poisson model.*, Appl. Math. Lett., 17:785-790.

Sullender W.M., 2000, *Respiratory Syncytial Virus Genetic and Antigenic Diversity.*, Clinical Microbiology Reviews, 13:115.

Suzuki, Y., and Gojobori, T., 1999, *A method for detecting positive selection at single amino acid sites.*, Mol Biol Evol, 16:1315-1328.

Swofford, D.L., 1998, *PAUP*. Phylogenetic Analysis Using Parsimony (*andOther Methods). Version 4.*, Sinauer Associates, Sunderland,Massachusetts. Copyright © David L. Swofford, 1989-1997. Copyright © Smithsonian Institution, 1998. All Rights Reserved.

Vogel, A.M., Lennon, D.R., Harding, J.E., Pinnock, R.E., Graham, D.A., Grimwood, K. and Pattemore, P.K., 2003, *Variations in bronchiolitis management between five New Zealand hospitals: Can we do better?*, J. Pediatr. Child Health, 39:40-45.

Woelk C.H. and Holmes E.C., 1998, *Variable Immune-Driven Natural Selection in the Attachment (G)Glycoprotein of Respiratory Syncytial Virus (RSV)*, J Mol Evol, 52:182192.

Wright, S., 1930, *Evolution in Mendelian Populations*, Genetics 16:96-159

Yang, Z., 1997, *PAML: a program package for phylogenetic analysis by maximum likelihood.*, Computer Applications in BioSciences, 13:555-556.

Yarus M., 1999, *Boundaries for an RNA world.*, Curr Opin Chem Biol., 3:260-267.

Zintzaras, E., Santos, M., Szathmary, E., 2002 ,"*Living under the challenge of information decay: the stochastic corrector model vs. hypercycles.*, Journal Theoretical Biology, 217:167-181.

5 Appendix 1: Sequence Group and Number Keys.

5.1 Number Key for figures 3.14 and 3.15.

Virus names in the labels below are in the format [ID]_[Year of Isolation]_[Patient ID].

1) 299_94_0000008B

2) 278_93_00002238

3) 291_93_0000000B

4) 027_94_00003240, 061_94_00001896, 050_94_00001896

5) 294_93_0000003B

6) 316_93_0000235

7) 124_90_00001867, 071_90_00000792, 282_90_00000742

8) 313_90_00000022

9) 051_92_00002412, 006_96_00004195, 035_92_00002510, 041_92_00001224, 045_92_00001574

10) 081_89_00002171, 199_84_00002114, 126_87_00001997 11) 025_92_00002359, 039_92_00002359

12) 116_87_00008938, 174_88_00002139, 202_88_00001698

13) 188_90_00002215

14) 094_89_00002380

15) 099_84_00006766, 104_84_00007933, 105_84_00007071

16) 154_84_00007620

17) 156_85_00005633

18) 072_85_00005687

19) 314_90_00000023

20) 212_90_00002540

21) 256_93_00002455

5.2 Groupings for Figure 3.16: RSV B F-Protein Alignment.

Virus Names are in the labels below are in the format Subtype[ID_[Year of Isolation].

Group_1: B025_92, B039_92, B051_92, B006_95, B035_92, B041_92, B045_92, B212_90,

B314_90

Group_2: B256_93, B116_87, B174_88, B202_88, B124_90, B071_90, B282_90

Group_3: B099_84, B154_84, B104_84, B105_84

Group_4: B072_85, B156_85

Group_5: B081_89, B199_84, B126_87, B313_90

Group_6: B027_94, B061_94, B050_94, B294_93, B316_93

5.3 Groupings for Figure 3.17: RSV B G-Protein Alignment.

Virus Names are in the labels below are in the format Subtype|ID_[Year of Isolation].

Group_1: B025_92, B039_92, B051_92, B035_92, B041_92, B045_92

Group_2: B099_84, B104_84, B105_84

Group_3: B072_85, B156_85

Group_4: B081_89, B199_84, B126_87, B313_90

Group_5: B116_87, B174_88, B202_88

Group_6: B124_90, B071_90, B282_90

Group_7: B027_94, B061_94, B050_94

Group_8: B291_93, B299_94

5.4 Number Key for Figures 3.18 and 3.19.

Virus names in the labels below are in the format [ID]_[Year of Isolation]_[Patient ID].

1) 006_96_0000419

2) 029_94_00003304, 095_88_00001781, 031_94_00002697, 046_92_00001719, 036_95_00003992

3) 150_84_00006797, 234_84_00006572, 223_84_00007651, 158_84_00006510, 129_87_00000758, 145_87_00003841, 287_90_00002393, 255_90_00001516, 197_89_00002358, 109_82_00007035, 149_83_00000315, 139_84_00005988

4) 322_NA_0000003

5) 056_86_00000956, 161_86_00000956, 165_86_00000956, 168_86_00000956, 160_82_00006079, 170_86_00000956, 162_86_00000956, 166_86_00000956

6) 073_91_00002549, 246_91_00002549

7) 102_84_00008530

8) 201_89_00002079

9) 157_84_00005742

10) 311_90_00003252

11) 284_90_00002263

12) 290_90_00002394

13) 151_87_00008673, 117_87_00008851, 211_87_00001248, 132_88_00001489, 180_88_00001453, 147_88_00002109, 286_87_00012485, 118_87_00009076, 136_88_00001723, 179_88_00002037, 185_88_00002016, 210_88_00001770, 196_88_00001769, 194_88_00001733

14) 059_86_00000956

15) 321_NA_00000030

16) 108_84_00006965

17) 178_88_00002013, 181_88_00001960, 177_88_00002000, 195_88_00002000

18) 110_82_00006029, 111_82_00005960

19) 152_82_00005308

20) 135_88_0000202

21) 077_67_12567OLD, 079_67_19967OLD, 078_67_01867OLD, 034_94_00003831, 141_88_0000020

22) 207_88_00001938, 198_88_00001965

23) 184_88_00001656, 176_88_00001656

24) 018_94_00001723

25) 003_96_00002847, 167_86_00006136

26) 070_85_00000294, 224_84_00007650

27) 067_94_00002494

28) 014_94_00002027, 026_94_00002508, 016_94_00002163, 043_94_00003346

29) 125_89_00001946

30) 264_91_00001034

31) 237_91_0000300

32) 226_91_00002228, 232_91_00002177, 239_91_00002054, 229_91_00002465, 251_91_00002131

33) 235_91_00001766

34) 273_93_00002581, 320_93_00000006, 319_93_00000026, 090_89_00002522, 297_93_00000006, 302_93_00000011, 300_93_00000009, 305_93_00000011, 315_93_00000024

35) 085_89_00001602, 215_89_00002355, 218_89_00001602, 283_89_00001632

36) 216_89_0000160

37) 098_89_0000174

38) 230_91_00002215, 222_91_00002436, 245_93_00002406, 267_93_00002578, 257_93_00002406, 296_93_00002406, 268_93_00002580

39) 228_91_00002232

40) 032_94_00002845

41) 042_94_00002397, 266_90_00002392, 193_90_00002392, 254_93_00001218, 009_95_00001479

42) 075_73_31873OLD

43) 065_92_00002505, 060_92_00002263

44) 243_93_0000309

45) 248_93_00002582

46) 279_93_00002582

47) 225_91_00002220, 221_91_00002823

48) 123_89_0000211

49) 200_88_0000198

50) 206_88_00001825, 097_89_00002359, 134_82_00005308

51) 122_84_00005372

52) 231_91_00002287

5.5 Groupings for Figures 3.21 and 3.22: RSV A G-Protein Alignment.

Virus Names are in the labels below are in the format Subtype|ID_[Year of Isolation].

Group_1: A_297, A_302, A_305, A_273, A_320, A_319, A_090

Group_2: A_085, A_218, A_283

Group_3: A_226, A_232, A_229, A_251, A_239

Group_4: A_016, A_014, A_026

Group_5: A_206, A_134, A_097

Group_6: A_109, A_149

Group_7: A_111, A_110

Group_8: A_178, A_195, A_181, A_177

Group_9: A_003, A_167

Group_10: A_070, A_224

Group_11: A_129, A_145

Group_12: A_150, A_234, A_223

Group_13: A_141, A_077, A_079, A_078

Group_14: A_166, A_056, A_161, A_165, d08_A_168, A_160, A_162

Group_15: A_073, A_246

Group_16: A_095, A_031, A_046

Group_17: A_151, A_117, A_211, A_180, A_286, A_118, A_136, A_179, A_210, A_194

Group_18: A_132, A_185, A_196

Group_19: A_279, A_248

Group_20: A_065, A_060

Group_21: A_009, A_254

Group_22: A_222, A_230

Group_23: A_266, A_193

Group_24: A_245, A_267, A_257, A_296, A_268