

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

On the Use of Optimal Search Algorithms with Artificial
Potential Field for Robot Soccer Navigation

DONG, Chen

Chief Supervisor

Reyes, Napoleon H., Ph.D

Co-Supervisor

Barczak, Andre L., Ph.D

Computer Science

Master of Science

January 23, 2018

Contents

1	Introduction	1
1.1	The Problem Domain: Robot Soccer Game	2
1.2	Research Objectives	3
1.3	Significance of the Research	4
1.4	Structure of the Thesis	5
2	Literature Review	7
2.1	Optimal Search Algorithms	7
2.1.1	Dijkstra Algorithm	8
2.1.2	A* Search Algorithm	13
2.1.3	Object Representation in the 2D Gridworld	17
2.1.4	Search in Vertex-Based and Cell-Based Worlds	20
2.1.5	Any-Angle Search and the Line-of-sight	21
2.1.6	Post-smoothing of the A* Search Algorithm	23
2.1.7	Theta* Search Algorithm	23
2.1.8	Limitations of the Optimal Search Algorithms	28
2.2	The Artificial Potential Field Algorithm	29
2.2.1	Artificial Potential Field for Navigation	29
2.2.2	Simplification of the Artificial Potential Field	30
2.2.3	Limitations of the Artificial Potential Field	31
2.2.4	Related Works based on the Artificial Potential Field	32
2.3	Robot Soccer	34
2.3.1	Dimensions of the Playing Field and the Agents	34
3	Preliminary Experiments	37
3.1	Platform	37
3.1.1	Terminologies and Statistical Measurements	37
3.1.2	Search Field	38
3.2	Optimal Search: Case Studies	40
3.2.1	One Big Obstacle in the Centre, Size 20x20	40
3.2.2	Four Medium Obstacles, Size 20x20	42
3.2.3	Five Small Obstacles, Size 20x20	44

3.2.4	Random Dots, Size 30x30	46
3.2.5	Four Walls with a Wiggled Lane, Size 20x20	48
3.2.6	A Series of Walls with a Lane in the middle, Size 30x30	50
3.2.7	Maze, Size 30x30	52
3.2.8	Potential Well, Size 30x30	54
3.3	Analysis	56
3.4	Conclusion	57
4	APF-Optimal Search	59
4.1	Overview	59
4.2	Potential Field with Optimal Search	59
4.2.1	Safety Factor	60
4.2.2	The Basic Artificial Potential Field Generator	62
4.2.3	Linear Functions	64
4.2.4	Hyperbola function	67
4.2.5	Sigmoid Function	67
4.2.6	Double Thresholds	69
4.2.7	Comparison with Related Study	70
4.3	Alternative Heuristics	72
4.3.1	Sharp Bend Problem	72
4.3.2	Dynamic Attractive Point of the Artificial Potential Field	73
4.3.3	The Angular Factor in Optimal Search	74
4.3.4	Involving the Angular Factor in the Search Algorithm	75
4.3.5	The Second Key Value	75
4.3.6	Unified Key with Separate Coefficients	75
4.3.7	Alternative Heuristic Function	76
4.4	Line-of-sight	78
4.4.1	Simple Line-of-Sight Checking for Vertex-based Search	79
4.4.2	Modified Bresenham's Algorithm for Line-of-Sight Detection	80
4.4.3	Obstacle as Polygon	83
4.4.4	Modified Cohen-Sutherland Algorithm	85
5	Building Robot Behaviours	89
5.1	Generic Strategies	89
5.1.1	Target Pursuing	89
5.1.2	Obstacle Avoidance	90
5.2	Attacker Strategies	92
5.2.1	Dynamic Attacking Position	92
5.2.2	Finite State Machine for Attacker Intelligence	93
5.3	Goal Keeper Strategies	94
5.3.1	Defensive Blocking Position	94
5.3.2	Parking	94

5.3.3	Finite State Machine for Keeper Intelligence	95
6	Simulation Platform	97
6.1	Implementing System Dynamics	97
6.1.1	Collision Detection	99
6.1.2	The Force and the Impulse Direction	102
6.1.3	Applying the Artificial Potential Field	104
6.2	Implementation Details	105
6.2.1	Grid mapping and Dynamic Cell Size	105
6.2.2	Use of the Artificial Potential Field	106
6.2.3	Multi-Linked List	106
6.3	Architecture Design	106
6.3.1	Main-loop and FPS Limits	107
6.3.2	Structure of the Rigid Body Management	108
6.3.3	Messages and Event Handling	111
7	Experiments	113
7.1	Experiment Introduction	113
7.1.1	Platform of the Search Algorithm Experiment	113
7.1.2	Definitions and the Statistics	113
7.1.3	Safety Factor of the Potential Field	114
7.2	APF-Optimal Search	114
7.2.1	Comparison Between the APF-A* and the Original A*	115
7.2.2	One Big Obstacle in the Centre, Size 20x20	117
7.2.3	Four Medium Obstacles, Size 20x20	119
7.2.4	Five Small Obstacles, Size 20x20	121
7.2.5	Random Dots, Size 30x30	123
7.2.6	Four Walls with a Wiggled Lane, Size 20x20	125
7.2.7	A Series of Walls with a Lane in the Middle, Size 30x30	127
7.2.8	Maze, Size 30x30	129
7.2.9	Potential Well, Size 30x30	131
7.2.10	Trap, Size 30x30	133
7.2.11	GNRON Case	135
7.2.12	Summary	137
7.3	APF Generator	137
7.3.1	The Distribution of the Magnitude Using Different Generators	138
7.4	Comparison with Other Studies	142
7.4.1	Experiments	142
7.5	Robot Behaviours	146
7.5.1	Platform Introduction	146
7.5.2	Experiments of the Performance	147
7.5.3	Experiment Cases and Results	149

7.5.4	Summary	159
7.6	Analysis and Discussion	160
7.6.1	The Performance of the APF-Optimal Search Algorithms	160
7.6.2	The Performance of the Artificial Potential Field Generator	160
7.6.3	The Performance of the Line-of-sight Algorithms	161
7.6.4	The Effects of Using the Safety Factor on Performance	161
8	Conclusion	165
8.1	Future Works	167
8.1.1	Alternate Optimal Search Algorithms	167
8.1.2	Decision Making	167
8.1.3	Use of Fuzzy System with Artificial Potential Field	168
8.1.4	A* Search in Vector Space	168
	Bibliography	169

List of Figures

2.1	A directed map for dijkstra algorithm	8
2.2	Step 1 - Insert node A into S as initial	10
2.3	Step 2 - Pop A and insert B, C into S	10
2.4	Step 3 - Pop C which cost is 2, insert D, E	11
2.5	Step 4 - Pop E, insert F, but not end	11
2.6	Step 5 - Pop B, D is in S but no update	12
2.7	Step 6 - Pop D, F is in S which has new lower cost	12
2.8	The initial status of A* Search	15
2.9	Two successors of S	16
2.10	Only one new node, the other are blocked	17
2.11	Further step	17
2.12	A 2D-Plain with obstacle	18
2.13	The cell partly or totally covered by the obstacle are marked	19
2.14	The marked cells are blocked after removing the obstacle	19
2.15	The original field that has not been divided by the grid	20
2.16	Search based on the vertices	21
2.17	Search based on the cells	21
2.18	A path between A4 and D1 is valid	22
2.20	The same path of the figure 2.19 could be smoothed	22
2.19	The path found by A* is wiggled	23
2.21	A possible path found by Theta* Search	24
2.22	Examine the vertex D2	25
2.23	Visit the successor of D2: C2 and D3	25
2.24	Examining the line-of-sight from E1 to the successors	26
2.25	Remove D2 on the path since there is a line-of-sight	26
2.26	Search has be advanced to D4	27
2.27	Checking the line-of-sight for the successors of D4	27
2.28	The line-of-sight examining for the successors of C5 is from D4 but not E1	28
2.29	Magnitude of the APF	31
2.30	Target is not reachable due to the obstacles around	32
2.31	Size of the field	35

3.1	Black cells are blocked, light green is the start cell and cyan is the goal cell	39
3.2	A sample path found by a search algorithm	39
3.3	Test map of a big obstacle in the centre	41
3.4	The path found by A* Search	41
3.5	The path found by Theta* Search	41
3.6	The map of 4 medium obstacles	43
3.7	The path found by A* Search	43
3.8	The path found by Theta* Search	43
3.9	The map of 5 small obstacles	45
3.10	The path found by A* Search	45
3.11	The path found by Theta* Search	45
3.12	The ma of a set of dots	47
3.13	The path found by A* Search	47
3.14	The path found by Theta* Search	47
3.15	The map of 4 walls with a wiggled lane	49
3.16	The path found by A* Search	49
3.17	The path found by Theta* Search	49
3.18	The map of multiple walls with a lane in the middle	51
3.19	The path found by A* Search	51
3.20	The path found by Theta* Search	51
3.21	The map of maze	53
3.22	The path found by A* Search	53
3.23	The path found by Theta* Search	53
3.24	The map have a potential well	55
3.25	The path found by A* Search	55
3.26	The path found by Theta* Search	55
4.1	The shortest path vs. The safest path	61
4.2	The distance from s to the obstacles	63
4.3	The diagram of the margin function	64
4.4	Search with Margin function	64
4.5	The drawback of margin function	64
4.6	The gray scale illustrates the magnitude of the artificial potential field .	65
4.7	The diagram of the linear function	65
4.8	Manhattan Distance	66
4.9	Manhattan Distance plus artificial potential field magnitude	66
4.10	The diagram of the hyperbola function	67
4.11	The diagram of the sigmoid function	68
4.12	The gap where the robot cannot pass	69
4.13	Logistic sigmoid with two thresholds	70

4.14	According to the study in [51], the position relationship between the robot, goal and obstacle in 1-D scenario.	70
4.15	According to the study in [51], the magnitude distribution of the APF in 1-D scenario.	71
4.16	The magnitude distribution by linear function	71
4.17	The magnitude distribution by hyperbola function	72
4.18	The magnitude distribution by Sigmoid function	72
4.19	The sharp bend in the path	73
4.20	Two paths have same fcost but different angular cost	74
4.21	Same node may have different angular heuristic from different parent . .	76
4.22	The same path of the figure 2.19 could be smoothed	79
4.23	An intermediate step when processing LOS checking	79
4.24	The light gray cells are where should be examined	81
4.25	The dark gray cells will not be found by the original algorithm but should be examined	83
4.26	The intersection between a segment and a polygon	84
4.27	The flags of the Cohen-Sutherland Algorithm	85
5.1	The flow chat of path planning and re-planning	90
5.2	A possible path found by the A* Search	91
5.3	The distribution of the artificial potential field around the obstacle 2 . .	91
5.4	The expected actual trail of the agent	92
5.5	The attacking position	93
5.6	The states and transition conditions of the attacker	93
5.7	Moving to the blocking position E for coming attacking	95
5.8	The states and transition conditions of the keeper	96
6.1	Use 5 parameters to describe a rectangle	98
6.2	Two rectangles are overlapped	99
6.3	The distance between a circle and a rectangle	100
6.4	Determine which quadrant the circle belongs to	101
6.5	The left object hits the right one	103
6.6	To determine if need to compute virtual force	105
6.7	The main-loop of the simulation	107
6.8	The class diagram of rigid body management	108
6.9	The procedure of poll event	112
7.1	The original A* Search	115
7.2	The APF-A* Search	115
7.3	The original A* Search with one dot in the middle	116
7.4	The APF-A* Search with one dot in the middle	116
7.5	The magnitude distribution when the obstalce is placed in the middle .	116

7.6	A big obstacle in the centre of the field	117
7.7	Path found by APF-A* Search without SF	117
7.8	Path found by APF-A* Search with SF	117
7.9	Path found by APF-Theta* Search without SF	117
7.10	Path found by APF-Theta* Search with SF	117
7.11	Four medium obstacles	119
7.12	Path found by APF-A* Search without SF	119
7.13	Path found by APF-A* Search with SF	119
7.14	Path found by APF-Theta* Search without SF	119
7.15	Path found by APF-Theta* Search with SF	119
7.16	Five small obstacles	121
7.17	Path found by APF-A* Search without operator factor	121
7.18	Path found by APF-A* Search with operator factor	121
7.19	Path found by APF-Theta* Search without operator factor	121
7.20	Path found by APF-Theta* Search with operator factor	121
7.21	A set of dots placed randomly	123
7.22	Path found by APF-A* Search without SF	123
7.23	Path found by APF-A* Search with SF	123
7.24	Path found by APF-Theta* Search without SF	123
7.25	Path found by APF-Theta* Search with SF	123
7.26	Four walls and a wiggled path	125
7.27	Path found by APF-A* Search without SF	125
7.28	Path found by APF-A* Search with SF	125
7.29	Path found by APF-Theta* Search without SF	125
7.30	Path found by APF-Theta* Search with SF	125
7.31	A series of walls with a lane in the middle	127
7.32	Path found by APF-A* Search without SF	127
7.33	Path found by APF-A* Search with SF	127
7.34	Path found by APF-Theta* Search without SF	127
7.35	Path found by APF-Theta* Search with SF	127
7.36	The maze which is the most complex terrian	129
7.37	Path found by APF-A* Search without SF	129
7.38	Path found by APF-A* Search with SF	129
7.39	Path found by APF-Theta* Search without SF	129
7.40	Path found by APF-Theta* Search with SF	129
7.41	The potential well that the agent will be trapped in the middle	131
7.42	Path found by APF-A* Search without SF	131
7.43	Path found by APF-A* Search with SF	131
7.44	Path found by APF-Theta* Search without SF	131
7.45	Path found by APF-Theta* Search with SF	131
7.46	The trap where the goal is at the opposite direction of the exit	133

7.47	Path found by APF-A* Search without SF	133
7.48	Path found by APF-A* Search with SF	133
7.49	Path found by APF-Theta* Search without SF	133
7.50	Path found by APF-Theta* Search with SF	133
7.51	Path found by APF-A* Search with SF	135
7.52	Path found by Original A* Search	135
7.53	Magnitude distribution	136
7.54	Distribution of the linear function	138
7.55	Distribution of the hyperbola function to the power of -2	139
7.56	Distribution of the hyperbola function to the power of -1	139
7.57	Distribution of the sigmoid function	140
7.58	Comparison between the functions	141
7.59	The scenario of GNRON problem from the study in [51]	142
7.60	The performance of the APF-A* Search without SF	143
7.61	The performance of the APF-A* Search with SF	143
7.62	The performance of the Theta* Search without SF	143
7.63	The performance of the APF-Theta* Search with SF	143
7.64	The path planning demonstration from the study in [5]	144
7.65	The performance of the APF-A* Search without SF	144
7.66	The performance of the APF-A* Search with SF	144
7.67	The performance of the APF-Theta* Search without SF	145
7.68	The performance of the APF-Theta* Search with SF	145
7.69	Sample performance of the Robots	147
7.70	The path found by the APF-A* Search with SF	149
7.71	The path found by the APF-Theta* Search with SF	150
7.72	The attacking and the defense	151
7.73	The path found by the APF-A* Search with SF	152
7.74	The path found by the APF-Theta* Search with SF	152
7.75	The path found by the APF-A* Search with SF	154
7.76	The path found by the APF-Theta* Search with SF	154
7.77	The defense of the keeper	155
7.78	The path found by the APF-A* Search with SF	156
7.79	The path found by the APF-Theta* Search with SF	156
7.80	The re-planning of the attacker after the keeper push the ball away	157
7.81	50 Random obstacles and the path planned by the APF-Theta* Search	158
7.82	The average running time and the obstacle count	159

List of Tables

2.1	gcost and heuristic of new search nodes	27
3.1	Statistics in the case of Big Obstacles	40
3.2	Statistics in the case of Four Medium Obstacles	42
3.3	Statistics in the case of Five Small Obstacles	44
3.4	Statistics in the case of Random Dots	46
3.5	Statistics in the case of Four Walls with a Wiggled Lane	48
3.6	Statistics in the case of Walls with a Lane in the Middle	50
3.7	Statistics in the case of Maze	52
3.8	Statistics in the case of Potential Well	54
3.9	Statistics of the Performance Ratios between the Optimal Searches	56
4.1	Search nodes at first step	77
4.2	Angular Factors	77
4.3	gcost and heuristic of all the nodes	77
7.1	Statistics of the Performances of APF-Optimal Searches with or without SF, in the case of Big Obstacle	118
7.2	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Four Medium Obstacles	120
7.3	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Five Small Obstacles	122
7.4	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Random Dots	124
7.5	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Four Walls with a Wiggled Lane	126
7.6	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Walls with a Lane in the Middle	128
7.7	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Maze	130
7.8	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Potential Field	132

7.9	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Trap	134
7.10	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of Replicated Environment from the Study by Ge & Cui in 2000	143
7.11	Statistics of the Performances of APF-Optimal Searches with and without SF, in the case of the Replicated Environment from the Study by Li, Yamashita, Asama & Tamura in 2012	145
7.12	Statistics of the Performances of APF-Optimal Searches with SF, in the case of the Potential Well running on the Simulator	150
7.13	Statistics of the Performances of APF-Optimal Searches with SF, in the case of the Multiple Walls with a Lane in the Middle running on the Simulator	153
7.14	Statistics of the Performances of APF-Optimal Searches with SF, in the case of the GNRON problem running on the Simulator	155
7.15	Statistics of the Performances of APF-Optimal Searches with SF, in the case of the Cross Blocks running on the Simulator	157
7.16	Statistics of the Performances of APF-Optimal Searches with SF, in the case of the Random Obstacles running on the Simulator	158
7.17	Comparison Between Original A* and APF-A* SF, in the case of One Big Obstacle, according to the figures 3.4 and 7.8	161
7.18	Comparison Between Original A* and APF-A* SF, in the case of Four Medium Obstacles, according to the figures 3.7 and 7.13	162
7.19	Comparison Between Original A* and APF-A* SF, in the case of Five Small Obstacle, according to the figures 3.10 and 7.18	162
7.20	Comparison Between Original A* and APF-A* SF, in the case of Random Dots, according to the figures 3.13 and 7.23	162
7.21	Comparison Between Original A* and APF-A* SF, in the case of Four Walls with a Wiggle Lane, according to the figures 3.16 and 7.28	163
7.22	Comparison Between Original A* and APF-A* SF, in the case of Walls with a Lane in the Middle, according to the figures 3.19 and 7.33	163
7.23	Comparison Between Original A* and APF-A* SF, in the case of Maze, according to the figures 3.22 and 7.38	163
7.24	Comparison Between Original A* and APF-A* SF, in the case of Potential Well, according to the figures 3.25 and 7.43	164
8.1	Summary of the Algorithms Tested and Developed	167

Abstract

The artificial potential field (APF) is a popular method of choice for robot navigation, as it offers an intuitive model clearly defining all attractive and repulsive forces acting on the robot [3] [25] [29] [43] [50]. However, there are drawbacks that limit the usage of this method. For instance, the local minima problem that gets a robot trapped, and the Goal-Non-Reachable-with-Obstacle-Nearby (GNRON) problem, as reported in [51] [5] [23] [2] and [3]. In order to avoid these limitations, this research focuses on devising a methodology of combining the artificial potential field with a selection of optimal search algorithms. This work investigates the performance of the method when using different optimal search algorithms such as the A* algorithm and the any-angle path-planning Theta* Search, in combination with different types of artificial potential field generators. We also present a novel integration technique, whereby the Potential Field approach is utilized as an internal component of an optimal search algorithm, considering the safeness of the calculated paths. Furthermore, this study also explores the optimization of several auxiliary algorithms used in conjunction with the APF-Optimal search integration: There are three different methods proposed for implementing the line-of-sight (LOS) component of the Theta* search, namely the simple line-of-sight checking algorithm, the modified Bresenham's line algorithm and the modified Cohen-Sutherland algorithm. Contrary to the studies presented in [5], [42], [48] and [40] where the APF and the optimal search algorithms were used separately, in this research, an integrative methodology involving the APF inside the optimal search with a newly proposed Safety Factor (SF) is explored. Experiment results indicate that the APF-A* Search with the SF can reduce the number of state expansions and therefore also the running time up to 19.61%, while maintaining the safeness of the path, as compared to APF-A* when not using the SF. Furthermore, this research also explores how the proposed hybrid algorithms can be used in developing multi-objective behaviours of single robot. In this regard, a robot soccer simulation platform with a physics engine is developed as well to support the exploration. Lastly, the performance of the proposed algorithms is examined under varying environment conditions. Evidences are provided showing that the method can be used in constructing the intelligence for a robot goal keeper and a robot attacker (ball shooter). A multitude of AI robot behaviours using the proposed methods are integrated via a finite state machine including: defensive positioning/parking, ball kicking/shooting, and target pursuing behaviours.

Keywords : Artificial Potential Field, Optimal Searches, Robot Navigation, Multi-objective Behaviours.

