



THE UNIVERSITY  
*of* ADELAIDE

Evolution of High Level Motion Control for Autonomous  
Ground Vehicles

Mohd Faisal Ibrahim

A Thesis submitted for the degree of Doctor of Philosophy

School of Computer Science

The University of Adelaide

2015

# Contents

<b>Abstract</b>	<b>viii</b>
<b>Thesis Declaration</b>	<b>x</b>
<b>Acknowledgement</b>	<b>xi</b>
<b>Abbreviations</b>	<b>xii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Overview . . . . .	2
1.2 Motivation . . . . .	3
1.3 Research Goal . . . . .	6
1.4 Method Overview . . . . .	7
1.5 Contributions . . . . .	9
1.6 Thesis Outline . . . . .	10
1.6.1 Related Publications of the Described Contributions . . .	12
<b>2 Literature Review</b>	<b>13</b>
2.1 Autonomous Robotic Exploration . . . . .	14
2.2 State-of-the-Art of Autonomous Robotic Exploration System . .	17
2.2.1 State-of-the-Art of Exploration Strategies . . . . .	17
2.2.2 State-of-the-Art of Motion Control . . . . .	25
2.3 Evolutionary Computation based Motion Control . . . . .	30
2.4 Introduction to EC Techniques - GA, GE and CMA-ES . . . . .	38
2.4.1 Genetic Algorithms . . . . .	38
2.4.2 Grammatical Evolution . . . . .	46
2.4.3 Covariance Matrix Adaptation Evolutionary Strategy . .	49
2.5 Conclusion . . . . .	52
<b>3 Theoretical Frameworks and Descriptions of Experiments</b>	<b>53</b>
3.1 Frontier-based Exploration Strategies . . . . .	54
3.1.1 Updating Occupancy Grid Maps . . . . .	56
3.1.2 Detecting Frontiers . . . . .	59
3.1.3 Assigning Frontiers to Robots . . . . .	60

3.2	Navigating Robots to Frontiers using Motion Control . . . . .	64
3.2.1	General Framework of DTC . . . . .	65
3.2.2	Decision Theory in DTC . . . . .	66
3.3	Using Grammatical Evolution to Optimise Motion Control . . .	68
3.3.1	Genetic Algorithms as the Search Engine . . . . .	74
3.3.2	Formulating the Language-specification . . . . .	77
3.3.3	Formulating the Problem-specification . . . . .	79
3.3.4	GE Mapper . . . . .	80
3.4	Descriptions of the Experimental Set Up . . . . .	85
3.4.1	Software . . . . .	86
<b>4</b>	<b>Autonomous Exploration with Multiple Robots</b>	<b>87</b>
4.1	Objectives . . . . .	88
4.2	The Exploration Framework . . . . .	88
4.2.1	UGV Model . . . . .	90
4.2.2	Exploration Strategy . . . . .	91
4.2.3	Motion Control Structure . . . . .	95
4.3	The Evolutionary Process . . . . .	109
4.4	Experimental Results . . . . .	114
4.4.1	The Derivation of Handwritten Scoring Function . . . . .	114
4.4.2	The Results . . . . .	115
4.5	Discussion . . . . .	119
4.6	Conclusion . . . . .	119
<b>5</b>	<b>Single Robot Multi-Objective Exploration</b>	<b>121</b>
5.1	Objectives . . . . .	121
5.2	The Exploration Framework . . . . .	122
5.2.1	Motion Control Structure . . . . .	123
5.3	The Evolutionary Process . . . . .	128
5.3.1	Objective Function . . . . .	129
5.3.2	Grammatical Evolution . . . . .	130
5.3.3	Covariance Matrix Adaptation Evolutionary Strategy . .	132
5.4	Experimental Results . . . . .	133
5.5	Discussion . . . . .	141
5.6	Conclusion . . . . .	143
<b>6</b>	<b>Evolving the Structure of a Policy</b>	<b>144</b>
6.1	Objectives . . . . .	144
6.2	Formulating a Policy . . . . .	145
6.3	Setting Up the Grammars . . . . .	146
6.3.1	Grammar 1: Evolving Constants . . . . .	149
6.3.2	Grammar 2: Evolving Functions . . . . .	151
6.3.3	Grammar 3: Evolving Structure . . . . .	152

6.4	Experiments . . . . .	156
6.4.1	Experimental Set Up . . . . .	156
6.4.2	Experimental Results . . . . .	160
6.5	Discussion . . . . .	168
6.6	Conclusion . . . . .	176
<b>7</b>	<b>Evolving Input Factor Choice of a Policy</b>	<b>177</b>
7.1	Objectives . . . . .	178
7.2	Related Work on DWA Policies . . . . .	178
7.3	Turtlebot - A Mobile Indoor Robot . . . . .	180
7.4	The Exploration Framework on ROS . . . . .	183
7.4.1	Mapping Module . . . . .	184
7.4.2	Frontier Module . . . . .	185
7.4.3	Navigation Module . . . . .	186
7.5	The Dynamic Window Approach . . . . .	188
7.6	Setting the BNF Grammar for Input Factor Selection . . . . .	191
7.7	Experiments . . . . .	193
7.7.1	Learning Phase . . . . .	193
7.7.2	Application Phase . . . . .	198
7.8	Results and Discussion . . . . .	201
7.8.1	Learning Phase Results . . . . .	201
7.8.2	Application Phase Results . . . . .	209
7.9	Conclusion . . . . .	226
<b>8</b>	<b>Conclusions</b>	<b>228</b>
8.1	Summary of Findings . . . . .	228
8.2	Future Work . . . . .	232
	<b>Bibliography</b>	<b>234</b>

# List of Figures

2.1	A general framework of autonomous robotic exploration. . . . .	15
2.2	Steps for finding optimal solution of a problem. . . . .	39
2.3	An example of a chromosome represents parameter $x$ and $y$ of equation 2.1. . . . .	41
2.4	An example of a pool of parent chromosomes. . . . .	44
2.5	An example of one-point crossover operation. . . . .	45
2.6	An example of point mutation operation. . . . .	45
2.7	GE steps for finding optimal solution of a problem. . . . .	46
2.8	The BNF grammar for parameters $x$ and $y$ of equations 2.1 and 2.2. . . . .	47
2.9	An example of single mapping in GE using equation 2.5. . . . .	49
2.10	An example of CMA-ES candidate solutions distribution on certain generations. . . . .	51
3.1	Algorithm for Frontier-based exploration strategy. . . . .	55
3.2	Algorithm for occupancy grid update. . . . .	57
3.3	One-dimensional sensor model profile. . . . .	58
3.4	Example of frontiers detection process. . . . .	60
3.5	General framework of a DTC. . . . .	66
3.6	The framework of GE. . . . .	73
3.7	An example of the <i>effective</i> crossover. . . . .	77
3.8	An example of production rules to generate mathematical expressions. . . . .	78
3.9	An example of binary-to-decimal genotype conversion. . . . .	82
3.10	An example of mapping codon integers with production rules into a valid mathematical expression. . . . .	83
4.1	A multi-UGV system with one ground station (GS). . . . .	89
4.2	A prototype UGV developed at the University of Adelaide for Multi-Autonomous Ground-robotic International Challenge (MAGIC) 2010. . . . .	91
4.3	Motion model of the UGV. . . . .	92
4.4	The framework of the Frontier-based exploration strategy. . . . .	93
4.5	The motion control structure of a $UGV_i$ . . . . .	96
4.6	UGV state schema in the global frame. . . . .	97

4.7	An example of static hazard costmap generation. . . . .	98
4.8	An example of static hazard state signal calculation. . . . .	100
4.9	An example of dynamic hazard costmap generation. . . . .	104
4.10	An example of goal strength costmap generation. . . . .	106
4.11	An example of the selection process of a UGV's steering direction. . . . .	108
4.12	Snippet of the scoring function source code. . . . .	111
4.13	The BNF grammar. . . . .	111
4.14	Map 1: 15m x 15m indoor-like area. . . . .	113
4.15	Map 2: 15m x 15m indoor-like area. . . . .	116
4.16	Boxplots of explored area between handwritten motion control and evolved motion control. . . . .	117
4.17	Evolution run performance at each generation. . . . .	120
5.1	The motion control structure of a UGV in single robot multi-objective exploration set up. . . . .	124
5.2	An example of 16 candidate short-range target locations surrounding the robot. . . . .	125
5.3	Outline of the evaluative process of a candidate $\pi$ . . . . .	129
5.4	Learning environments. . . . .	130
5.5	The BNF grammar. . . . .	132
5.6	Snippet of the scoring function source code. . . . .	132
5.7	Validation map and explored area. . . . .	135
5.8	Box plots presenting exploration coverage of scoring functions. . . . .	136
5.9	Box plots of power efficiency vs. controllers. . . . .	140
5.10	Contour maps of two-dimensional state signals. . . . .	142
6.1	A generative $\pi$ model for the DTC. . . . .	147
6.2	A production rule to produce single digit number using the digit concatenation technique. . . . .	149
6.3	A production rule to produce a numerical constant. . . . .	149
6.4	Grammar 1 for evolving numerical constants. . . . .	150
6.5	Grammar 2 for evolving unary functions and constants. . . . .	152
6.6	Example of $\pi$ production using grammar 2. . . . .	153
6.7	Grammar design heuristic. . . . .	154
6.8	Grammar 3 for evolving the $\pi$ structure. . . . .	155
6.9	Example of $\pi$ production using grammar 3. . . . .	157
6.10	Experimental set up. . . . .	158
6.11	A learning environment. . . . .	160
6.12	CONST_EVO evolution run performance. . . . .	161
6.13	FUNC_EVO evolution run performance. . . . .	162
6.14	STRUC_EVO evolution run performance. . . . .	163
6.15	Boxplots of evolutionary performance. . . . .	164
6.16	Application maps with footprints of the robot. . . . .	167
6.17	A description of state signals. . . . .	169

6.18	CONST_EVO: Unary functions of state signals. . . . .	171
6.19	FUNC_EVO: Unary functions of state signals. . . . .	172
6.20	STRUC_EVO: Unary functions of state signals. . . . .	174
6.21	Scoring function ranking surfaces. . . . .	175
7.1	Turtlebot 2 platform. . . . .	180
7.2	Turtlebot 2 base. . . . .	181
7.3	Hokuyo URG-04LX-UG01 laser scanner. . . . .	183
7.4	Turtlebot 2 equipped with a Hokuyo LIDAR sensor. . . . .	183
7.5	ROS-based robotic exploration system. . . . .	184
7.6	Block diagram of the mapping module. . . . .	185
7.7	Block diagram of the frontier module. . . . .	185
7.8	Block diagram of the navigation module. . . . .	186
7.9	Example of an obstacle cost map generated by DWA. . . . .	188
7.10	A set of candidate circular trajectories. . . . .	189
7.11	An example of velocity pairs distribution. . . . .	190
7.12	A production rules template for evolution of input factors. . . . .	192
7.13	A BNF grammar for FACRAN_EVO and FACDET_EVO. . . . .	195
7.14	The <i>Embedding</i> stage of evaluation. . . . .	196
7.15	A set of learning maps. . . . .	197
7.16	Maps used in the application phase. . . . .	200
7.17	Evolution performance of CONST_EVO scheme. . . . .	203
7.18	Evolution performance of FACRAN_EVO scheme. . . . .	204
7.19	Evolution performance of FACDET_EVO scheme. . . . .	205
7.20	Medians of CONST_EVO scheme performance. . . . .	206
7.21	Medians of FACRAN_EVO scheme performance. . . . .	207
7.22	Medians of FACDET_EVO scheme performance. . . . .	208
7.23	Map transition: Accumulated collision status. . . . .	212
7.24	Map transition: Accumulated completion status. . . . .	213
7.25	Map transition: Boxplot of explored area of figure 7.16(a). . . . .	214
7.26	Map transition: Boxplot of explored area of figure 7.16(b). . . . .	215
7.27	Map transition: Boxplot of explored area of figure 7.16(c). . . . .	216
7.28	Localisation transition: Accumulated collision status. . . . .	218
7.29	Localisation transition: Accumulated completion status. . . . .	219
7.30	Localisation transition: Boxplot of explored area of figure 7.16(a). . . . .	219
7.31	Localisation transition: Boxplot of explored area of figure 7.16(b). . . . .	220
7.32	Localisation transition: Boxplot of explored area of figure 7.16(c). . . . .	221
7.33	Platform transition: A testing environment to be explored by the robot. . . . .	224
7.34	Platform transition: Seven frontier points plotted on the ground-truth map of the testing environment. . . . .	225
7.35	Platform transition: Accumulated collision status. . . . .	226

# List of Tables

2.1	An example of four chromosomes and their fitness value. . . . .	42
2.2	An example of four chromosomes with their accumulated fitness. . . . .	44
4.1	Evolutionary process set up. . . . .	114
4.2	Performance indicator for controllers. . . . .	117
4.3	Main statistical data comparison between controllers. . . . .	118
5.1	Evolutionary process set up. . . . .	131
5.2	Data of explored area. . . . .	138
5.3	One-way ANOVA test and post hoc multiple comparisons. . . . .	139
6.1	GE configurations for three evolution groups. . . . .	159
6.2	Data of area explored by all controllers. . . . .	166
6.3	T-test among three evolution groups. . . . .	166
6.4	Application phase performance. . . . .	167
7.1	GE configuration for all runs. . . . .	196
7.2	Selected input factors by FACRAN_EVO schemes. . . . .	209
7.3	Selected input factors by FACDET_EVO schemes. . . . .	210
7.4	Table of scoring functions of selected DWA policies. . . . .	211
7.5	Map transition: T-test results to analyse the variation of the exploration coverage performance. . . . .	216
7.6	Localisation transition: T-test results to analyse the variation of the exploration coverage performance. . . . .	222
7.7	Platform transition: Average exploration time and its standard deviation for all evolved DWA policies. . . . .	225



# Abstract

Autonomous robotic exploration is the task of building models of an environment. This task requires robots to rapidly plan, re-plan and execute their motion trajectories using sensory data that is provisional, uncertain and noisy. To navigate successfully under these conditions, robots require carefully designed motion controller software to guide the robot safely, quickly, reliably and efficiently to intermediate exploration objectives. Conventionally, the basic design of a motion controller is derived from first principles using simplified models of motion control and then refined by hand in response to observed performance. While this approach works in simpler applications, it becomes more challenging and less effective as applications become more complex and the number of variables to consider increases. Moreover, changes in robot configuration and environment can entail costly redesign of the controller. As such, we argue that this manual approach will become increasingly impractical as our exploration tasks become more ambitious. In this thesis, we address the development of motion control using techniques from Evolutionary Computation (EC). Our approach is to view the motion control design as a search problem, that can be subject to automation. In this work we present a novel framework for evolving the core component of motion control based on a form of EC called Grammatical Evolution (GE). GE systematically refines populations of potential programmatic solutions for a given problem, until an effective solution is found. In our approach, we use GE to search automatically for the best motion control for a given set of exploration tasks. GE allows the user to constrain the search space for programs using Backus-Naur Form (BNF) grammar specifications. We use these grammars to define the search space for controllers for each exploration application. We conducted four experiments to evaluate our proposed approach. Each experiment demonstrates the framework in different exploration configurations and different requirements. All of our experiments evolved controller code for unmanned ground vehicles (UGV's). Our first experiment evolved numerical parameters for the control of small teams of UGV's. Our second experiment evolved control for a single UGV to optimise exploration performance and energy consumption. Our third experiment evolved both the structure and parameters of the core control function. Our fourth experiment evolved the input factor selection and numerical constants for well-established navigation approach in progressively

more realistic situations - culminating in deployment on real platforms. In each of our experiments we found that the automated search approach outperformed carefully designed handwritten control. Moreover the structure of the evolved equation helped to reveal the nature of the trade-offs inherent in the exploration task and what factors appear to be most relevant to informing effective control.

# Thesis Declaration

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree. I give consent to this copy of my thesis when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

The author acknowledges that copyright of published works contained within this thesis resides with the copyright holder(s) of those works.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

MOHD FAISAL IBRAHIM  
Date: 8<sup>th</sup> DECEMBER 2015

# Acknowledgement

First of all, I would like to express my sincere gratitude and appreciation to my supervisors Prof. Zbigniew Michalewicz and Dr. Bradley Alexander for your countless guidance and supports during my Ph.D study. Both of you are great teachers teaching me a lot about research. I could not think about completing my study without their encouragements, supports and motivation.

Next, my appreciation to the members of Optlog group and my lab's colleagues for your generous thoughts and useful discussions that spark ideas most of the time.

My sincere thanks also to all academic and supporting staffs at the School of Computer Science, International Student Centre (ISC) and Adelaide Graduate Centre (AGC) at The University of Adelaide for helping me on administration stuffs.

I would also like to thank the Ministry of Education Malaysia and Universiti Kebangsaan Malaysia (UKM) for the full sponsorship during my Ph.D study. Not to forget to all my friends and relatives in Adelaide and Malaysia, many thanks for your warm supports and encouragements.

A special thank to my family for their support and sacrifices. To my father Hj. Ibrahim, my mother Hj. Jahani, my father-in-law Dr. Hj. Baseri Huddin, my mother-in-law Hj. Zaharah and all my siblings, words cannot describe my gratitude for your prayers and supports. To my beloved wife Aqilah who is always at my side, special appreciation for your endless sacrifices and understandings along the moments of pursuing our study. To my daughter Aliya, you always cheer up my life. Love you all.

Last but not least, I would also like to thank the members of my examination committee. Their comments have improved the final presentation of the thesis significantly.

# Abbreviations

APF	Artificial Potential Field
BNF	Backus-Naur Form
CFG	Context-Free Grammar
CMA-ES	Covariance Matrix Adaptation Evolutionary Strategy
DTC	Decision-Theoretic based Motion Control
DWA	Dynamic Window Approach
EC	Evolutionary Computation
ER	Evolutionary Robotics
ES	Evolutionary Strategy
FLC	Fuzzy Logic Controller
GA	Genetic Algorithm
GE	Grammatical Evolution
GP	Genetic Programming
GS	Ground Station
GPS	Global Positioning System
HRL	Hierarchical Reinforcement Learning
IMU	Inertia Measurement Unit
LE	Layered Evolution
LIDAR	Light Detection and Ranging Sensor
MPC	Model Predictive Control
ND	Nearness Diagram
NN	Neural Network
POMDM	Partially Observable Markov Decision Model
PSDP	Policy Search Dynamic Programming
RL	Reinforcement Learning
SGHC	Simple Genetic Hill Climbing
SLAM	Simultaneous Localisation and Mapping
SVM	Support Vector Machine
UGV	Unmanned Ground Vehicle
VFH	Vector Field Histogram