

# Immune Systems Inspired Multi-Robot Cooperative Shepherding

by

Sazalinsyah Razali

A Doctoral Thesis

Submitted in partial fulfilment  
of the requirements for the award of

Doctor of Philosophy

of

Loughborough University

24th November 2014

© by Sazalinsyah Razali 2014



# Dedication

To my family,  
for the sacrifices made...  
Thank you...  
for everything...

*“So, verily, with every difficulty, there is relief.”*

*“Verily, with every difficulty there is relief.”*

# Abstract

Certain tasks require multiple robots to cooperate in order to solve them. The main problem with multi-robot systems is that they are inherently complex and usually situated in a dynamic environment. Now, biological immune systems possess a natural distributed control and exhibit real-time adaptivity, properties that are required to solve problems in multi-robot systems. In this thesis, biological immune systems and their response to external elements to maintain an organism's health state are researched. The objective of this research is to propose immune-inspired approaches to cooperation, to establish an adaptive cooperation algorithm, and to determine the refinements that can be applied in relation to cooperation. Two immune-inspired models that are based on the immune network theory are proposed, namely the *Immune Network T-cell-regulated—with Memory* (INT-M) and the *Immune Network T-cell-regulated—Cross-Reactive* (INT-X) models. The INT-M model is further studied where the results have suggested that the model is feasible and suitable to be used, especially in the multi-robot cooperative shepherding domain. The Collecting task in the RoboShepherd scenario and the application of the INT-M algorithm for multi-robot cooperation are discussed. This scenario provides a highly dynamic and complex situation that has wide applicability in real-world problems. The underlying 'mechanism of cooperation' in the immune inspired model (INT-M) is verified to be adaptive in this chosen scenario. Several multi-robot cooperative shepherding factors are studied and refinements proposed, notably methods used for Shepherds' Approach, Shepherds' Formation and Steering Points' Distance. This study also recognises the importance of flock identification in relation to cooperative shepherding, and the Connected Components Labelling method to overcome the related problem is presented. Further work is suggested on the proposed INT-X model that was not implemented in this study, since it builds on top of the INT-M algorithm and its refinements. This study can also be extended to include other shepherding behaviours, further investigation of other useful features of biological immune systems, and the application of the proposed models to other cooperative tasks.

# Acknowledgements

My deepest gratitude goes to my lovely wife, Mashanum Osman for her patience and companionship and to my three beautiful children; Muhammad Marwan, Muhammad Saifullah & Nur Sakinah; for their love and laughter all these years. To my parents, Hj. Razali Endun & Hj. Selemah Manan, to whom I am most indebted. Thank you to my siblings, Risalinsyah, Seri Diana, & especially Dewi Ratna Seri for being there during our early days at Loughborough.

I sincerely thank Dr. Qinggang Meng & Prof. Shuang-Hua Yang for their guidance and assistance in supervising me through the difficult times. Thank you also to my initial Director of Research, Prof. Paul W. H. Chung and later Director of Research Degree Programme, Dr. Ana M. Sălăgean in managing the research direction. My thanks also goes to my yearly progress examiner, Prof. Chris Hinde for his insights and feedback. Thank you also to the ex-Head of Department, Prof. Eran Edirisinghe. I am fortunate to have been involved in various research groups, namely the Digital Imaging Research Group (DIRG); which is a part of the Vision, Imaging & Autonomous Systems Group (VIAS), the Networks, Communications & Control Systems (NCCS) Research Group, & the Intelligent & Interactive Systems Group (IIS). I thank all the academicians & student members of those groups.

Thank you to Md Zaid Ahmad, Amna Saad, Aslina Saad, Afizan Azman, Muhammad Athar Ali, Mohamad Saada, Muhammad Akramshah Ismail, Ibrahim Tholley, Amelia Ritahani Ismail, Siti Nurhaida Khalil, Aede Hatib Musta'amal, Norzanah Rosmin, Shahrol Mohamaddan, Maswida Mustafa, Zainal Abidin Adlin, Muhammed Nafis Osman Zahid, all my friends at the CS Department, Loughborough, Leicester, Nottingham & FTMK.

My utmost appreciation goes to Prof. Honghai Liu & Dr. Walter Hussak for examining my thesis. I gratefully acknowledge Malaysian Ministry of Education & Universiti Teknikal Malaysia Melaka (UTeM) for sponsoring this Ph.D. study.

**Sazalinsyah Razali**

# Declaration

Parts of the research reported within this thesis are based on the author's previous presented publications: Razali et al. [78, 79, 80, 81, 82, 83, 84]. These publications are also listed in [Appendix A](#).

# Contents

<b>Abstract</b>	<b>iv</b>
<b>Acknowledgements</b>	<b>v</b>
<b>Declaration</b>	<b>vi</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background . . . . .	1
1.2 Problem Formulation . . . . .	2
1.3 Research Objectives . . . . .	3
1.4 Motivations . . . . .	4
1.5 Contributions . . . . .	5
1.6 Thesis Structure . . . . .	6
<b>2 Literature Review</b>	<b>8</b>
2.1 Introduction . . . . .	8
2.1.1 Cooperation . . . . .	9
2.2 Computational Intelligence Techniques . . . . .	12
2.2.1 Non Bio-inspired Cooperation Approaches . . . . .	12
2.2.2 Bio-inspired Cooperation Approaches . . . . .	13
2.2.3 Why Immune Systems . . . . .	14
2.3 Biological Immune Systems . . . . .	14
2.3.1 Immune Systems . . . . .	14
2.3.2 Immune Network Model of B-cell . . . . .	16
2.4 Multi-Robot Cooperation . . . . .	17
2.4.1 Swarm-Immune Algorithm . . . . .	19
2.4.2 Immune Network Model of B-cell and T-cell . . . . .	19
2.4.3 Immune Network and Potential Field . . . . .	20
2.5 Conclusion . . . . .	22
<b>3 Immune Inspired Model for Cooperation</b>	<b>23</b>
3.1 Introduction . . . . .	23
3.2 Immune Systems Approach . . . . .	23

3.2.1	Immunoid: the Immune Network based Robot . . . . .	24
3.3	Immune Network for Group Behaviour . . . . .	25
3.3.1	Definition of Task . . . . .	25
3.3.2	Definition of Antigen . . . . .	26
3.3.3	Definition of Antibody . . . . .	28
3.3.4	Group Control Algorithm . . . . .	29
3.4	Immune Systems Inspired Cooperation Model . . . . .	32
3.4.1	The INT-M Model: Immune Network with Memory . . . . .	33
3.5	Cooperative Tasks . . . . .	35
3.5.1	RoboShepherd . . . . .	35
3.5.2	Cooperative Robots for Perimeter Detection and Tracking . . . . .	38
3.6	Conclusion . . . . .	39
<b>4</b>	<b>Experiments &amp; Results of Immune Inspired Models in Multi-Robot Cooperation Tasks</b>	<b>41</b>
4.1	Introduction . . . . .	41
4.2	Simulation Setup . . . . .	41
4.3	RoboShepherd Test Scenario . . . . .	44
4.3.1	Scenario Setup . . . . .	45
4.4	Simulation Results . . . . .	46
4.4.1	Average Time for Completion . . . . .	46
4.4.2	Average Number of Incomplete Task . . . . .	47
4.4.3	Discussions . . . . .	48
4.5	Verification of the Immune Inspired Cooperative Mechanism . . . . .	49
4.5.1	Response to Environmental Changes . . . . .	50
4.5.2	Propagation of Stimulation and Suppression of Antibodies . . . . .	53
4.6	Conclusion . . . . .	56
<b>5</b>	<b>Cooperative Shepherding Refinements</b>	<b>57</b>
5.1	Introduction . . . . .	57
5.2	Shepherding Behaviour's Refinements . . . . .	58
5.2.1	Shepherds' Approach: Safe Zone . . . . .	59
5.2.2	Shepherds' Formation: Lining-up . . . . .	59
5.2.3	Steering Points' Distance . . . . .	64
5.3	Simulation Experiments . . . . .	66
5.3.1	Simulation Setup . . . . .	67
5.3.2	Performance Criteria . . . . .	68
5.3.3	Results . . . . .	68
5.4	Flock Identification Refinements . . . . .	70



5.4.1	Other Approaches . . . . .	71
5.4.2	Motivation . . . . .	72
5.4.3	Proposed Method . . . . .	73
5.4.4	Performance Measures . . . . .	76
5.4.5	Results . . . . .	77
5.5	The INT-X Model: Cooperation with Immune Learning . . . . .	82
5.6	Conclusion . . . . .	85
<b>6</b>	<b>Conclusion</b>	<b>86</b>
6.1	Summary . . . . .	86
6.2	Main Contributions . . . . .	88
6.3	Suggestions and Future Work . . . . .	89
	<b>References</b>	<b>90</b>
<b>A</b>	<b>List of Publications</b>	<b>101</b>
<b>B</b>	<b>List of Activities</b>	<b>103</b>

# List of Figures

1.1	The research areas identified: interest is on the central overlapping area . . . . .	3
1.2	Hierarchical view of the research focus . . . . .	4
2.1	An MRS taxonomy focused on coordination [24] . . . . .	11
2.2	Antigen-antibody binding and Jerne’s Idiotypic Network Theory . .	15
2.3	Basic biological immune systems response [20] . . . . .	16
2.4	An example of how the combinatin of immune network and potential field exhibits cooperative behaviour [50] . . . . .	21
2.5	Architecture of the Reactive Immune Network (RIN) system for mobile robot reactive navigation [60] . . . . .	22
3.1	Immunoid: a robot with an action selection mechanism [35] . . . . .	24
3.2	State diagram of group behaviour . . . . .	25
3.3	Stimulus function of antigen to antibody, $g_i$ [95] . . . . .	27
3.4	Immune Network which includes T-cell and B-cell models [95] . . .	29
3.5	Immune Network-based Cooperative Robots [48] . . . . .	30
3.6	B-cell activation and differentiation into Memory and Plasma cells [20]	33
3.7	The dog-sheep problem scenario, the red ovals are the dogs and the blue circles are the sheep . . . . .	36
3.8	The INT-M states, the greyed states are bypassed when the memory mechanism is triggered resulting in the dashed arrow lines . . . . .	37
3.9	The perimeter detection and tracking problem scenario . . . . .	39
4.1	Player/Stage: the simulation environment being used . . . . .	43
4.2	Player/Stage/Gazebo: the big picture [46] . . . . .	44
4.3	The simulation experiment: involving 2 sheep (red) . . . . .	46
4.4	Average Time for Completion . . . . .	47
4.5	Average Number of Incomplete Tasks (Time limit: 300 seconds) . .	48
4.6	$Ab_0$ to $Ab_3$ are the average of robot 1–3, which start with high task density then changed to 0 density at $t = 50$ . . . . .	51

4.7	$Ab_0$ to $Ab_3$ are the average of robot 1–3, robot 4 starts with 0 task density then changed to detect 75.0% of the task (like the other robots) at $t = 50$ . . . . .	52
4.8	Robot 4 becomes excellent over time, then changed to be inferior at $t = 50$ thus almost instantaneously receives the strategy (i.e. $Ab$ ) from the other robots; which in this case is $Ab_0$ (Aggregation); but later changed back as it returns to be excellent once again . . . . .	54
4.9	Over time, robot 4 becomes excellent and continues to do so as the environment has not changed; thus it maintains its strategy of $Ab_2$ (Dispersion). At the same time, its $Ab_0$ strategy is highly stimulated via propagation by other robots . . . . .	55
5.1	An example of the refinement of low-level shepherding behaviour: robot dogs lining-up (the grazing site is located at the top-right corner) . . . . .	58
5.2	An example of the robot dogs lining-up; the red marker is the imaginary flock centre (the grazing site is located at the bottom-right corner) . . . . .	59
5.3	Illustration of the use of safe zone in shepherds' approach and the occurrence of flock separation when it is not used . . . . .	60
5.4	An example of Global Distance Minimisation lining-up method . . . . .	60
5.5	An example of Vector Projection lining-up method . . . . .	61
5.6	An example of Greedy Distance Minimisation lining-up method . . . . .	61
5.7	A graphical depiction at the start of the lining-up experiment; red markers are steering points replacing sheep, and there are blue, yellow and maroon robots to distinguish each team (both the shepherds and steering points are randomly positioned; the grazing site at the centre of the field) . . . . .	62
5.8	A graphical depiction after the lining-up experiment . . . . .	63
5.9	The steering points' distances in an example line-up formation . . . . .	65
5.10	Comparison of the steering points' distances using the three different methods in terms of timeStep . . . . .	65
5.11	Comparison of the steering points' distances using the three different methods in terms of total distance travelled . . . . .	66
5.12	An example of the simulation setup with 4 sheep (red) and 2 dogs (blue; another two dogs are not in the current view) . . . . .	67
5.13	Average Distance to Origin . . . . .	68
5.14	Average Incomplete Tasks . . . . .	69
5.15	Flock identification in a single shepherding scenario . . . . .	70

5.16	The current problem of perceived flock centre; the right-most & top-most sheep are not detected to be in the same flock . . . . .	72
5.17	The problem of flock identification; the red blob is the shepherd and the white-coloured sheep is an actual flock and the separated blue-coloured sheep should be dismissed, which can be identified using the connected components labelling method proposed. . . . .	73
5.18	First Pass in the Connected Components Algorithm . . . . .	75
5.19	Second Pass in the Connected Components Algorithm . . . . .	75
5.20	Example of Player/Stage simulation run . . . . .	78
5.21	Flock Identification accuracy for different workspaces . . . . .	79
5.22	Flock Member Identification accuracy for different workspaces . . . . .	79
5.23	Average Time Taken for different workspaces . . . . .	80
5.24	Immune Learning after a few immune response have been mounted [20]	82

# List of Tables

3.1	Relationship between Immune Systems and MRS . . . . .	24
3.2	Basic task density and $g_i$ relationship . . . . .	27
3.3	Antigen-antibody affinity stimulus function, $g_i$ (other index values remain as 0.0) . . . . .	27
3.4	Mutual stimulus coefficient, $m_{ij}$ . . . . .	28
4.1	Simulation setup for RoboShepherd scenario . . . . .	45
5.1	Summary of result for distance travelled, in metres (4 shepherds, 1000 iterations) . . . . .	63
5.2	Summary of result for time taken, in seconds (4 shepherds, 1000 iterations) . . . . .	64
5.3	Comparison of terms . . . . .	76
5.4	Comparison between with and without using Player/Stage, with an average of two actual flocks in each 10 different scenarios . . . . .	80
5.5	Frequencies of flock identification in a $40 \times 40$ workspace (without using Player/Stage) . . . . .	81
5.6	Frequencies of flock identification in a $40 \times 40$ workspace (using Player/Stage) . . . . .	81
A.1	List of publications in refereed academic journals . . . . .	101
A.2	List of publications in scientific community periodicals . . . . .	101
A.3	List of publications in refereed conference proceedings . . . . .	102
B.1	List of related activities achieved . . . . .	103

# List of Equations

2.1 Stimulus for antibody $i$ , $S_i(t)$ - Farmer et al. . . . .	17
2.2 Concentration of antibody $i$ , $s_i(t)$ - Farmer et al. . . . .	17
2.3 Stimulus for antibody $i$ , $S_i(t)$ - Sun et al. . . . .	19
2.4 Antibody $i$ concentration, $s_i(t)$ - Sun et al. . . . .	20
2.5 T-cell concentration, $c_i(t)$ - Sun et al. . . . .	20
3.1 Simplified stimulus formula for antibody $i$ , $S_i(t)$ . . . . .	31
3.2 Upper threshold for determining excellent immunoid, $\bar{\tau}$ . . . . .	32
3.3 Lower threshold for determining inferior immunoid, $\underline{\tau}$ . . . . .	32
5.1 Accuracy of the proposed method in flock identification . . . . .	76
5.2 Accuracy of the proposed method in flock member identification . . . . .	77
5.3 The time taken for the proposed method to complete . . . . .	77

# List of Algorithms

3.1	General Immune Network Algorithm—for each immunoid . . . . .	31
3.2	Immune Network T-cell-regulated—with Memory (INT-M) . . . . .	34
3.3	Search strategy with pursuit behaviour . . . . .	38
5.1	Immune Network T-cell-regulated—Cross-Reactive (INT-X) . . . . .	84

# Chapter 1

## Introduction

### 1.1 Background

One of the main issues being studied in Multi-Robot System is cooperation between mobile and autonomous robots in order to achieve a common goal or to maximise the utility for each agent. Robots can also be viewed as agents, specifically embodied agent situated in the physical world. Agents can be defined as a situated computational system which is capable of autonomous action in some environment in order to achieve its design objectives [109]. Thus, Multi-Robot Systems can actually comprise of several homogeneous or heterogeneous self-interested agents. However, this research proposes an application area of dynamically changing environments such that the self-interested agents shall be of homogeneous in nature and the environment is continuous so that processing and decisions must be done in real-time.

There are several reasons why systems consisting of group of agents are of interest, and two of them are as follows [13]:

- tasks may be inherently too complex (or impossible) for a single agent
- building and using several simple agents can be easier, cheaper, more flexible and more fault-tolerant

Cooperation can be defined as a form of interaction, usually based on some form of communication [63]. But this definition is still quite general. Another more specific definition is taken from Robotics study whereby cooperative behaviour is as follows [13]:



*“Given some task specified by the designer, a multiple-robot system displays cooperative behaviours if, due to some underlying mechanism (i.e. the ‘mechanism of cooperation’), there is an increase in the total utility of the system.”*

Effective cooperation entails that the total utility of the system is increased, but at the same time the goal of each single agent is not totally abandoned nor delayed too long. It also requires that competition for resources among agents is minimised.

## 1.2 Problem Formulation

This research is interested in the use of an effective algorithm for cooperation in a team of robots in order to achieve its design objectives. This problem has indeed been studied by many researchers both in the robotics and multi-agent systems areas. The main research problems identified in this study are listed below.

1. Complexities that exist in multi-robot systems

In multi-robot systems, interaction between robots is highly problematic. The robots may be of different types, have different actuators, sensors or just have different capabilities. These differences lead to several complications, such as the inability to detect other robots and communication breakdown. Therefore, the problem within a team of multiple robots is quite difficult to overcome. However, it is still possible to make reasonable attempts at this problem provided that several assumptions and simplifications are introduced.

2. Dynamic environmental changes that are faced by robots

Another problem in multi-robot cooperation is the adaptation to environmental changes. Whenever the situation has changed, the ability to cooperate between robots must not be affected. It is understandable that the interaction of the robots will inevitably be affected by the changing environment, but it is intended that the robots would still be able to cooperate even at the minimal level in order to achieve its design objectives and complete the task at hand. Thus, the *mechanism of cooperation* must be able to withstand and be robust enough to overcome such problem.

### 3. Comprehensive interaction required in a multi-robot cooperation

*Method of interaction* in multi-robot cooperation should be more versatile and inclusive. It should not be too simple, as in a one-to-one interaction, but should involve all related robots that can affect the situation at hand. This research considers local group interactions as important so that emergent group behaviours that are optimal for the local environment can be achieved. Biological immune system manifests emergent cooperative behaviours in the form of the virus-fighting cells in the body. Hence, it is suitable as a method to approach the problem and is discussed in the following chapters.

Other than that, identifying suitable tasks that can be performed is also taken into consideration in studying multi-robot cooperation, since the task selected should be representative and can be scaled to bigger real world problems.

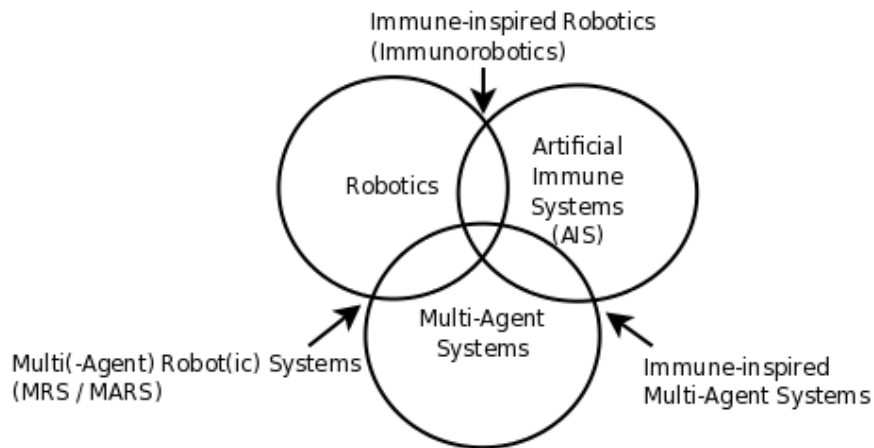


Figure 1.1: The research areas identified: interest is on the central overlapping area

Therefore, this study can be described as the use of immunology-based algorithm in achieving adaptive cooperation in a group of robots. Figure 1.1 shows the main research interest of this study that involves three main research areas. Meanwhile, Figure 1.2 shows the focus of this research.

## 1.3 Research Objectives

This study aims to overcome the problems listed earlier through three objectives. They are listed following the stages of the research whereby firstly an immune

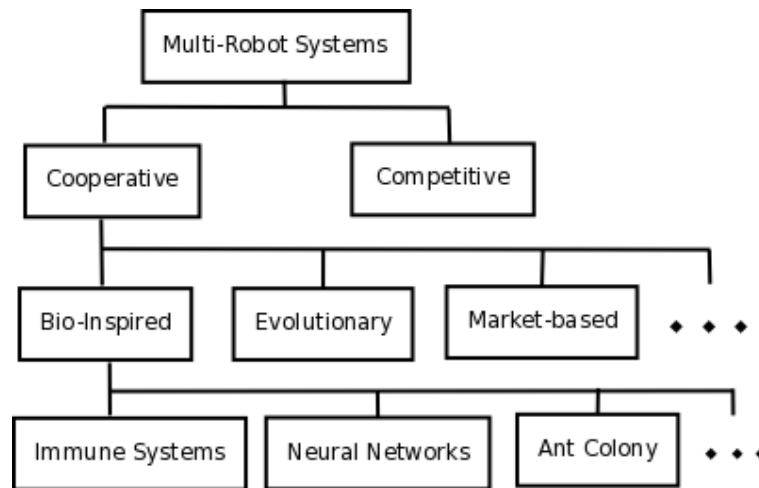


Figure 1.2: Hierarchical view of the research focus

systems approach is defined for robot cooperation, then properties of the algorithm is investigated to establish adaptive cooperation in a specific cooperative task, and finally the refinements of the immune-inspired cooperation behaviour is determined. Listed below are the objectives of this research:

1. To propose immune-inspired approaches to cooperation.
2. To establish an adaptive cooperation algorithm in multi-robot systems.
3. To determine the refinements that can be applied related to cooperation.

In terms of multi-robot cooperation tasks, the intention of this study is to have a representative task scenario that is applicable in other problem domains. Therefore, properties and requirements of cooperative tasks are investigated and the RoboShepherd task scenario is selected. This is presented in the following chapters.

## 1.4 Motivations

This research is mainly interested in the importance of overcoming or at least attempting to overcome the problems pertaining to cooperation in a team of robots. The challenges that motivate this research are described here.

1. The need of robots to cooperate or coordinate their action is vital in advancing their usability to the next level. Moreover, the abundance of robots

that are available today makes it almost inevitable for interactions between robots to occur.

2. The potential use of multi-robot systems that can autonomously cooperate is enormous. For example, such a system is useful in hazardous situations, space explorations, military operations, and even in our homes where several robots can be operating at the same location. These wide potential applications make the study even more important as it is quite possible to have an impact on the socio-economics of the society along with the technological advancements that could be achieved.
3. The biological immune system is a suitable candidate for a cooperation metaphor as it is proved that the task at hand (or rather its design objective) in most circumstances, is well achieved. The immune system cells have the magnificent property to autonomously coordinate their actions to achieve their common objective.

This research proposes that the cooperation among the robots is using approaches that have their roots in biology, specifically the Immune Systems. There indeed exist many models or frameworks proposed by others in the literature relating to cooperation. Some of them are MAPS [101], RETSINA [92], STEAM [97], and CORDA [75]. However, these models do not utilise the adaptive behaviour that can be derived from biology such as; in this case; the immune systems. Furthermore, this research is also driven by the fact that immune systems are not yet widely researched in the multi-robot systems domain.

## 1.5 Contributions

In this research, the use of immune systems inspired algorithms in order to achieve adaptive cooperation is in focus. This provides a new insight in multi-robot systems research, as a perspective that derives from immune systems is studied in order to realise a team of cooperating robots. Furthermore, interactions between multiple robots in such scenarios are also investigated because of its wide applicability in the real world.

The background understanding on the use and application of immune systems in multi-robot systems areas in this research can lead to further study on immune and multi-robot systems research interactions. The main contributions of this research are listed below.

1. Two immune-inspired models are proposed, and one of the model, the INT-M model is implemented and evaluated.
2. The implementation of the cooperative shepherding used in this research is using local ground view; except for the proposed flock identification method which rely on a ‘bird’s eye view’. This sets the study apart from other research, whereby such implementation is indeed difficult but it is more similar to real world situations.
3. The implementation of the immune inspired group behaviour takes into account all the nearby shepherds (i.e. within the communication radius) which is more realistic compared to other works that only uses a one-to-one communication that happens when the shepherds are in contact with one another.
4. The ‘cooperation mechanism’ underlying the immune inspired model (INT-M) is verified to be adaptive in a dynamic multi-robot scenario and supporting experimental data are provided.
5. Refinements related to multi-robot cooperative shepherding are identified and tested.
6. This study recognised the importance of flock identification in relation to cooperative shepherding task and a method to overcome the problem is discussed.
7. The implementation of this study is done on the Player/Stage robotics simulation platform. This means that it can be applied onto real robots with minor changes required.

The findings of the research is significant in the view that immune inspired approach to adaptive cooperation is tested and evaluated. The area of multi-robot systems cooperation now have a new and improved model to use in order to establish the intended interaction in a team of robots. Furthermore, an in-depth study of refinements on the cooperative shepherding behaviour had been conducted and is presented in this thesis.

## 1.6 Thesis Structure

This thesis is structured in the following way. In [chapter 2](#), we will first review the current research in multi-robot systems in general. Then the central theme

of cooperation is defined which is later followed by a general description of the immune systems as the main concern of this study. Then discussions are made on the several multi-robot cooperation techniques available.

In [chapter 3](#), discussions are presented on the proposed immune systems inspired cooperation model that is considered as feasible to be implemented. The model is described in general and will be studied and discussed more deeply in later chapters. Simulation results and verification of the ‘cooperation mechanism’ of the model are presented in [chapter 4](#). In [chapter 5](#), several refinements to the cooperative shepherding behaviour are proposed. The proposed model together with the refinements are again simulated and results are presented. In the latter part of the chapter, another refinement focusing on flock identification is proposed and its results are discussed. A second proposed model based on immune systems inspired cooperation is described at the end of [chapter 5](#).

The final chapter, [chapter 6](#), is where the works done in this study are summarised and the main contributions are listed. The chapter also provides several suggestions for future research works.

# Chapter 2

## Literature Review

### 2.1 Introduction

This chapter discusses other literatures that are related to this study. Two main themes that are crucial in this study are cooperation techniques or approaches, and immune systems literature in the area of multi-robot systems.

There are several overview on multi-robot systems research, as discussed by Cao et al. [13], Arai et al. [3], Wang et al. [106] and Lima and Custódio [55]. These papers are largely concerned with the diversity, usage, and impact of multi-robot systems research.

Multi-robot systems are being studied and applied in a vastly different domains, such as RoboCup [44], Search and Rescue [45, 96], Unmanned Aerial Vehicle (UAV) [14, 71], and military applications for example the DARPA Grand Challenge [6].

The Robot World Cup (RoboCup) is an international competition of soccer playing robots where the main goal is to have a team of autonomous humanoid robots that can beat the winner of FIFA World Cup by the year 2050. There are multitude of challenges and one of it is how the robot teams can cooperate to plan a strategy during game play. There are promising research on robot teams [97, 104], but other challenges remain such as learning and quick adaptation to dynamically changing environments.

There is also a variant competition of RoboCup known as RoboCup Rescue that focuses on humanitarian use of robotics, specifically in disaster mitigation

problem. The goal is to achieve multiple heterogeneous and autonomous robots that can be involved in search and rescue operations. This competition is more challenging since it is based on real-world scenarios and involves other autonomous robots and humans in the rescue operations.

On a related area, multiple Unmanned Aerial Vehicles (UAVs) pose interesting challenges with regard to autonomy and team coordination [47, 49]. In terms of autonomy, most UAVs still have human-in-the-loop operation. However, the goal to achieve operational autonomy or decision autonomy for UAVs is gaining attention [4], especially for military operations.

Multiple autonomous robots can be used in various military operations. Hence, the Defense Advanced Research Projects Agency (DARPA) had initiated the DARPA Grand Challenge competition in the year 2004. The goal was to build autonomous vehicles that can assist humans. The challenge was to manoeuvre in an open and rugged terrain. The team that won the competition in the second year by successfully completing the route was a group from Stanford with its autonomous vehicle named Stanley [102, 103]. This is followed by the DARPA Urban Challenge competition introduced in 2007 with the task for autonomous vehicles to navigate in an urban environment [18]. There is a new competition in 2013 called the DARPA Robotics Challenge (DRC) [17]. It was inspired from the Fukushima nuclear power plant meltdown. It consists of several challenges that are all related with responding to emergency situations in a hazardous location. The first to win this challenge is a robot from Japan named Schaft [100].

### 2.1.1 Cooperation

After reviewing related literatures, it is clear that there are two main terms being used interchangeably to define the concept of multiple robots cooperating together in order to execute a certain task. The first term is *cooperation* and the second term is *coordination*. These two terms are used in various contexts, and subsequently the definitions are not rigid. This leads to a minor confusion regarding which term is appropriate for this research context.

There is another set of multi-robot systems which is obviously not being considered that is the non-cooperative systems. These non-cooperative systems would normally fall into the category of competitive systems, such as soccer playing robots where there are competition between robots in order to achieve their goals. As this research is only looking at robots that are designed to cooperate, the



deep understanding and definition of that concept is discussed in the following paragraphs.

The term cooperation can be loosely defined as [69]:

*“robots that operate together to perform some global task”*

Meanwhile, taking the definition from the field multi-agent systems, the term coordination can be defined as [27]:

*“cooperation in which the actions performed by each robotic agents in such a way that the whole ends up being a coherent and high-performance operation”*

These two definitions are not exhaustive nor are they able to cover all aspects of the concept in various contexts. Nonetheless, these two definitions are suffice enough in differentiating the two terms. So, we can deduce that coordination is cooperation with the specific intention of operating coherently between team members and performing better as a whole group. Hence, we can safely conclude that in our context, coordination is a subset of cooperation. However, this study looks into the general concept of cooperation in its usage and effects, even though the algorithms would mean applying it to achieve a coordinated behaviour of the robots. Furthermore, the field of Cooperative Robotics is already established in the robotics research that encompasses the context applicable in this study.

Next, we should look into the classification of the various types and levels of cooperation in multi-robot systems. This classification gives us an insight on the overall picture of the research being undertaken.

The multi-robot system classification proposed in [24] which focused on coordination, is relevant and useful in describing this study. The taxonomy is divided into two dimensions, namely the *coordination dimension* and the *system dimension*. The coordination dimension aims at characterising the type or form of coordination in multi-robot systems. In other words, this dimension classifies based on ‘*how*’ the coordination is being done. The latter group; system dimension, is the taxonomy based on the features of the system that are relevant to its development. This system dimension cares about ‘*what*’ are being coordinated, that inevitably influence the system development.

Figure is redacted,  
kindly refer [24]  
by Farinelli, A., Iocchi, L., and Nardi, D. (2004)

Figure 2.1: An MRS taxonomy focused on coordination [24]

Referring to [Figure 2.1](#), the top level is regarding the ability of the system to cooperate. This distinguishes cooperative systems from non-cooperative ones (e.g. competitive). In this study, only cooperative systems are being considered. The second level is concerned with the knowledge of each robot about other robots in the group. However, this does not entail communication between robots. This level can also be further detailed as local or global information if the robot is *aware* of its team mates.

The third level is about the ‘mechanism of cooperation’, which distinguishes the system based on the underlying *coordination protocol*. A system that is *Not Coordinated* has no coordination protocol whatsoever, while a *Weakly Coordinated* system may be able to recognise other robots but does not have a model of the team mates. The final level is concerned with the way the decision system is realised in multi-robot systems. In a *Weakly Centralised* system, more than one agent is allowed to take the role of the leader during task execution.

In this taxonomy, coordination is similar to the concept of explicit cooperation as suggested by Mataric [62] while the concept of cooperation is similar to implicit cooperation. Another group of taxonomy which is called the system dimension is a classification based on features that include *communication*, *team composition*, *system architecture* and *team size* that are relevant to system development.

In terms of communication feature, it can be classified into direct and indirect communication. In indirect communication, usually the concept of *stigmergy* is used. In the robotics field, *stigmergy* can be generally defined as communication between two or more robots by sharing of information through inferring from modifications or changes made in the environment. However, communication can also be characterised in more detail with regard to topology, range and bandwidth as suggested by Dudek et al. [23]. The second feature in the system dimension is on team composition, that classify the team based on whether the multi-robot system consists of homogeneous or heterogeneous entities. However, a more precise classification can be achieved by using *Social Entropy* metric values as suggested by Balch [5]. Social Entropy concept is to get the diversity value of the robot society which is inspired by *Information Entropy* theory by Shannon [89].

In terms of system architecture, the system can be categorised into deliberative and reactive categories. Deliberative architecture uses an overall long-term plan for coping with environmental changes while in reactive architecture each robot pursues an individual approach to reorganise its own task in order to accomplish the goal assigned to it. The last feature is team size whereby classification is based on whether many robots are explicitly considered or not during the system design. Team size can also be simply measured quantitatively by the number of robots in the system.

## 2.2 Computational Intelligence Techniques

Several literatures that proposed cooperation approaches using soft computing techniques are reviewed since this study is a cross discipline work between multi-robot system and computational intelligence areas. The approaches in this category can be grouped into biologically and non-biologically inspired, as given in later subsections.

### 2.2.1 Non Bio-inspired Cooperation Approaches

Parker [72, 73] proposed the ALLIANCE approach that models teammate capabilities and performance and use the models to select tasks to execute that is beneficial to the group as a whole. Explicit communication is not required for the task selections. This seems to have a slight overhead since the robot need to observe its team members before executing a task. Another type of approach is the market-

based approach. Kalra et al. [40] provides a comprehensive overview of market-based multi-robot coordination works. Market-based approaches in multi-robot coordination uses the benefits of market economies, such as flexibility, efficiency, responsiveness, robustness, scalability, and generality. This type of approach had been implemented in several application areas such as robot exploration and soccer. However, the mechanism of market-based approach seems to be quite complex and not suitable for this research.

## 2.2.2 Bio-inspired Cooperation Approaches

There are several types of bio-inspired approaches that are related to cooperation. Some of them are discussed in this subsection.

One of the most notable bio-inspired approach is based on the the Ant Colony Optimization (ACO) algorithm [9, 66]. It is used in robotics research such as for path planning or multi-robot cooperation problems [56, 110]. It is based on mimicking a colony of ants and how they interact with each other and produce emerging optimal behaviour. It is advantageous in terms of the similarity of multi-robot problems, but does not seem to have optimal local group behaviour.

Meanwhile, another interesting approach inspired by nature is the Fish Swarm Algorithm (FSA). An overview of the FSA is discussed by Neshat et al. [68]. This approach uses the metaphor of swarms of fish in solving robotics problem such as Multi-Robot Task Allocation [111].

Artificial Bee Colony (ABC) optimisation algorithm is another approach that is used in robotics path planning problems [8]. It mimics the communication behaviour among bees in their colony.

Potter et al. [74] uses Artificial Neural Networks (ANN) to select appropriate behaviours in mobile robots, while Schultz et al. [88] uses Genetic Algorithms (GA) for learning the control methods for herding behaviours. However, these methods only consider herding with a few shepherds and hence cannot be applied to this study.

These bio-inspired approaches have their advantages but immune inspired approach is found to be more suitable for the multi-robot cooperative shepherding task that is being investigated in this research as described in later sections.

### 2.2.3 Why Immune Systems

The interest to study immune inspired approaches stems from the characteristics of the biological immune systems. Prominent characteristics of the immune system is that there is *no central control* of the lymphocytes in fighting antigens that invade the host and the systems *adaptability* in responding to various kind of antigens. The B-cells cooperatively merge at the affected area and produce appropriate antibodies for that particular antigen. This phase of immune response exhibits *cooperative* and *self-organising* behaviours of the related cells. Obviously, in immune network the processing of information is done in *real-time* and in a *distributed* manner; as what a multi-robot system requires. Details about the immune system is discussed in [subsection 2.3.1](#). As for immune systems related approach, several works are done that uses immune systems as metaphor to achieve some level of cooperation [32, 107, 108]. Examples of robotic problems that uses the immune inspired approach are path planning, fault-tolerance and cooperative box-pushing [38, 41, 42, 76]. An overview of robotics related applications of immune-inspired approaches is thoroughly discussed by Raza and Fernandez [77].

However, most of these works are limited either only to a single robot or does not consider a highly dynamic environment such as the RoboShepherd scenario. It is found that the simulations done for most works in the literature uses non-robotics based simulation platform or it is applied onto single robots. Furthermore, none of the immune inspired approaches have looked into the Memory Cells and Learning aspects of the immune system. In one of the proposed approach in this thesis, a specific memory mechanism is used in order to retain the appropriate action for relevant environment condition.

## 2.3 Biological Immune Systems

### 2.3.1 Immune Systems

An immune system is a system that eliminates foreign substances from an organism's body. These foreign substances such as bacteria, fungi or virus cells that can harm the host are called pathogens. When such substance activates an immune response it is called *antigen*, which stimulates the system's antibody generation. Each antigen has a unique set of identification on its surface called *epitope*. These antigenic determinant is where the host's antibodies would attach to by using its

paratope (see Figure 2.2). *Antibodies* are cells in the immune system that kill antigens in order to maintain the host homeostatic state; i.e. balancing the body's health status.

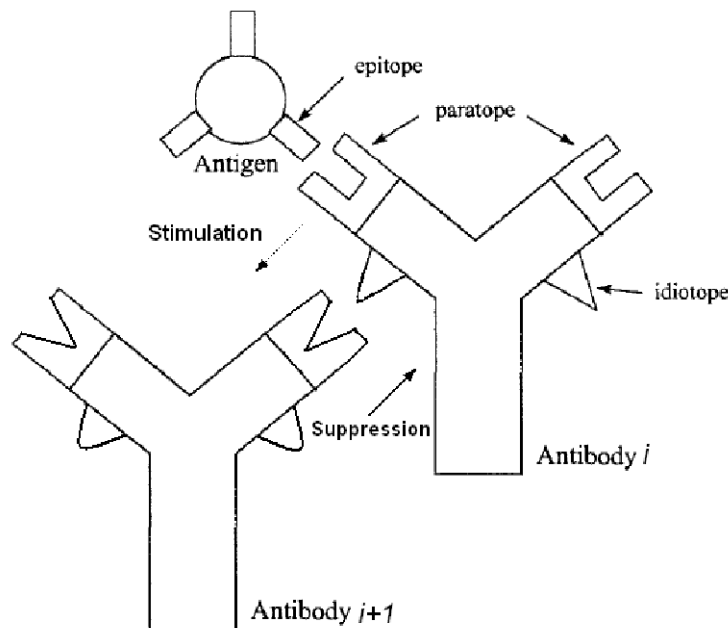


Figure 2.2: Antigen-antibody binding and Jerne's Idiotype Network Theory

The immune system can be divided into two general categories, innate immunity and adaptive immunity. Innate immunity is the first line of defence of the immune system. Generic pathogens that can be recognised and killed by the innate immunity cells would not be able to harm the host further. However, certain disease carrying antigens would bypass this defence mechanism because the innate immunity does not adapt to antigens that originate from various types of illnesses. The adaptive immunity would then play its role through the use of *lymphocytes* which are white blood cells. Lymphocytes have two main types, *T-cells* that mainly help in recognising antigen cells and *B-cells* that mainly produce antibodies to fight specific antigens. In humans, T-cells are primarily produced in the thymus while B-cells in the bone marrow. These two immune responses make up an effective and important defence mechanisms for living organisms.

The immune response basically can be viewed in six phases of recognition and activation (see Figure 2.3). Pathogen is digested by Antigen Presenting Cells (*APCs*) where it is broken down into *peptides* [20]. These peptides will then bind to Major Histocompatibility Complex (*MHC*) molecules, then present on the APC surface. T-cells recognise these different APC receptors and thus become activated. They divide and release *lymphokines* that transmit chemical signals

to stimulate other immune system components to take action. B-cells would then travel to the affected area and be able to recognise the antigen. This would activate the B-cells which then mature into *plasma cells*. Plasma cells are the ones which release specific antibody molecules that neutralise the particular pathogens.

Figure is redacted,  
kindly refer [20]  
by de Castro, L. N. and Timmis, J. (2002)

Figure 2.3: Basic biological immune systems response [20]

This immune response cycle results in the host's immunity against the antigen which triggers it, thus having protection in future attacks [20]. Prominent characteristics of the immune system is that there is no central control of the lymphocytes in fighting antigens that invade the host and the system's adaptability in responding to various kind of antigens. The B-cells cooperatively merge at the affected area and produce appropriate antibodies for that particular situation. This phase of immune response exhibits cooperative behaviour of the related cells.

### 2.3.2 Immune Network Model of B-cell

Studies in immunology have shown that antibodies are not isolated but communicate with each other. Each type of antibody has its specific *idiotope*, an antigen determinant (see Figure 2.2). Jerne, who is an immunologist, proposed the Idiotypic Network Hypothesis which views the immune system as a large-scale closed system consisting of interaction of various lymphocytes (B-cells) [39]. Referring to Figure 2.2, idiotope of antibody  $i$  stimulates antibody  $i + 1$  through its paratope. Antibody  $i + 1$  views that idiotope (belonging to antibody  $i$ ) simultaneously as an antigen. Thus, antibody  $i$  is suppressed by antibody  $i + 1$ . These mutual stimulation and suppression chains between antibodies form a controlling mechanism for

the immune response [20].

Farmer et al. [25] proposed differential equations to Jerne's idiotopic network hypothesis. These equations consist of antibodies' stimulus and suppression terms, antigen-antibody affinity, and cell's natural mortality rate [25]. This large-scale closed system interaction is the main mechanism that can be used for cooperation of multi-robot systems.

$$S_i(t) = S_i(t-1) + \left( \alpha \frac{\sum_{j=1}^N (m_{ij}s_j(t-1))}{N} - \alpha \frac{\sum_{j=1}^N (m_{ji}s_j(t-1))}{N} + \beta g_i(t) - k_i \right) s_i(t-1) \quad (2.1)$$

$$s_i(t) = \frac{1}{1 + \exp(0.5 - S_i(t))} \quad (2.2)$$

Equation 2.1 is the first equation where  $i, j = 1 \dots N$ ,  $N$  is the number of antibody types,  $S_i(t)$  is the stimulus value of antibody  $i$ ,  $s_i(t)$  and  $s_j(t)$  are the concentration of antibodies,  $m_{ij}$  is the mutual stimulus coefficient of antibody  $i$  and  $j$ ,  $g_i$  is the affinity of antibody  $i$  and antigen,  $\alpha, \beta$  are parameters of response rate of other antibody and antigen respectively, while  $k$  is the natural extinction coefficient. The values of  $m_{ij}$  and  $m_{ji}$  are not necessarily the same, as can be seen in the works of Luh and Liu [59], Luh et al. [61]. In Equation 2.2, the concentration of antibody  $i$  at time  $t$  is calculated as  $s_i(t)$ .

This section has discussed the definition and taxonomy of cooperation, and the immune systems approaches to achieve multi-robot cooperation. The Jerne's Idiotypic Network Hypothesis and also its derived equations by Farmer et al. [25] have been described. The next section will further discuss the immune inspired approaches to the problem.

## 2.4 Multi-Robot Cooperation

This study also covers a research area that can be known as Immunorobotics which is considered appropriate. Other immune systems related terms that have been



coined are immunocomputing and immunotronics. The term Immunocomputing was coined by [Tarakanov et al. \[98, 99\]](#) and it is similar to the term Artificial Immune Systems (AIS) that is widely used. The term Immunotronics was appropriately used by [Bradley and Tyrell \[10, 11\]](#) in the electronics hardware research area. This section discusses several cooperation techniques that have been developed by others. Cooperation techniques that are inspired by immune systems are included.

One multi-robot cooperation technique was proposed by Nagao and Miki [\[67\]](#) that uses local communication for a distributed multi-agent system. It uses what is called a state-based cooperation mechanism. The experiments were done using computer simulations for a surrounding task where robots need to surround a static target or beacon. This task is quite similar to a mine detection task by Srividhya and Ferat [\[94\]](#) that will be mentioned later.

In terms of specific cooperation that involves shepherding behaviour, Miki and Nakamura [\[65\]](#) proposed a shepherding method that requires shepherd to follow simple rules. The implemented flocks behaviour exactly follow the boids distributed behavioural model by Reynolds [\[85\]](#). The experiments were done using computer simulations but the scenario involved only one and two shepherd. However, it is interesting that the work was later implemented using a real robot platform [\[64\]](#).

An interesting multi-robot shepherding algorithm which is inspired by the herding commands and techniques used by actual shepherds was proposed by Bennett and Trafankowski [\[7\]](#). Simulations were performed and comparisons were made with the shepherding methods proposed by Lien et al. [\[51, 53\]](#) and Miki and Nakamura [\[65\]](#). There are not that many work on robotics cooperation that utilises the immune systems metaphor.

However, there are several interesting articles on immune inspired cooperation such as the works done by Gao and Wei [\[28\]](#) that proposed the Artificial Immune Network (AIN) model for Dynamic Task Allocation. The proposed model was applied to an emergency handling scenario that requires several robots to diffuse static alarms (targets) which is similar to the surrounding task earlier and the mine detection task mentioned later. The details of several works that specifically uses immune inspired approach in cooperation are described in [subsections 2.4.1, 2.4.2 and 2.4.3](#).

### 2.4.1 Swarm-Immune Algorithm

Lee and Sim [48] have proposed a simple immune network-based algorithm to achieve a swarm-like group behaviour. The algorithm is simple enough, however its main feature is that the decision making process is communicated throughout the local group of robots.

The useful part of this work is that it details all the relevant components of the immune networks and its application in the multi-robot systems domain. The components that are described in detail are antigen, antibody, mutual stimulus coefficient, antigen-antibody stimulus, excellent and inferior robots.

Another main feature of Lee and Sim's work is that the swarm group behaviour is achieved through local information so no global knowledge is required regarding the experimental area. This is advantageous as the robot need less *a priori* information and communication overhead and complexity is low.

However, the approach is limited to a task with the objective that is similar to a grazing behaviour whereby the robot searches for static target location and reacts based on the number of target detected at a particular site. Furthermore, the local group is limited with a one-to-one robot communication. Further discussions on this work with related diagrams are presented in section 3.3.

### 2.4.2 Immune Network Model of B-cell and T-cell

Sun et al. [95] have proposed a model based on Farmer et al. [25, 26] immune network equation as described in subsection 2.3.2; particularly Equations 2.1 and 2.2. The model involves T-cells as a control parameter which provides adaptation ability in group behaviour.

$$S_i(t) = S_i(t-1) + \left( \alpha \frac{\sum_{j=1}^N (m_{ij}s_j(t-1))}{N} - \alpha \frac{\sum_{j=1}^N (m_{ji}s_j(t-1))}{N} + \beta g_i(t) - c_i(t-1) - k_i \right) s_i(t-1) \quad (2.3)$$

$$s_i(t) = \frac{1}{1 + \exp(0.5 - S_i(t))} \quad (2.4)$$

$$c_i(t) = \eta(1 - g_i(t)) S_i(t) \quad (2.5)$$

In Equations 2.3 and 2.4,  $S_i(t)$  is the stimulus value of antibody  $i$  where  $i, j = 1 \cdots N$ ,  $N$  is the number of antibody types.  $m_{ij}$  is the mutual stimulus of antibody  $i$  and  $j$ , that can represent different values [59, 61].  $g_i$  is the affinity of antibody  $i$  and antigen,  $\alpha, \beta$  are parameters of response rate of other antibody and antigen respectively, while  $k$  is the natural extinction coefficient.  $s_i(t)$  is the concentration of antibody  $i$ . The difference with Farmer et al. [25, 26] immune network in Equation 2.1 is that  $s_j(t)$  is not the concentration of self-antibody, but that of other robot's antibody obtained by communication. Equation 2.5 is the added T-cell model whereby  $c_i(t)$  is the concentration of T-cell which control concentration of antibody.  $\alpha, \beta$ , and  $\eta$  are constants. In biological immune system, helper T-cells activate B-cells when antigen invades, and suppressor T-cell prevent the activation of B-cells when the antigen has been eliminated.

The advantage of adding the T-cell model is that the system adapts quickly to the environment by recovery of antibody concentration to the initial state, when antigens have successfully been removed. Thus, the system is more adaptable to environmental changes.

However, the drawback of this approach is that the objective of the task is only to locate and find the target which is static. Furthermore, communication between robots only occurs on a one-to-one basis if they happen to meet each other during execution of the task. The dynamic element is introduced in the task by putting back a set of target in the experiment area when the objective has been completed in the previous cycle.

### 2.4.3 Immune Network and Potential Field

Li et al. [50] have proposed an immune network based decision making for each robot coupled with potential field for the robots' local navigation. The main feature of this work is that the approach is applied to a very interesting problem.

The scenario selected is the dog-sheep problem which is very dynamic and offers a realistic challenge for a multi-robot cooperation approach. The dog-sheep scenario is discussed in detail in [subsection 3.5.1](#).

**Figure is redacted,  
kindly refer [50]  
by Li, J., Xu, H., Wang, S.,  
and Bai, L. (2007)**

Figure 2.4: An example of how the combinatin of immune network and potential field exhibits cooperative behaviour [50]

The deployment of immune network together with potential field in a dog-sheep scenario is beneficial as it is similar to other real world situation such as soccer and military. Furthermore the study looks into both simulation and uses real robot experiments to verify the approach. An example of the real robot experiments is shown in [Figure 2.4](#). This is useful, because it proves that immune inspired approach can be applied on multi-robot systems domain.

However, the article discussion is more focused on potential field approach rather than immune network. Furthermore, there is little information about the details of how the immune network is applied to the group of robots coordination mechanism. Other than that, because it introduces the potential field as a robot's local navigation strategy, the approach seems to need a lot of calculation overhead for each iteration.

Another work by Luh and Liu [58] is also related whereby the *Potential Field Immune Network* (PFIN) approach is proposed for mobile robots motion planning. They later proposed another immune-based method for reactive mobile robot navigation called the *Reactive Immune Network* (RIN) [60]. The general architecture of the system is shown in [Figure 2.5](#). These works are for robot navigation and did not directly study robot cooperation behaviours, but they are valuable nonetheless in understanding the different roles that the immune network can assume.

Figure is redacted,  
kindly refer [60]  
by Luh, G.-C. and Liu, W.-W. (2008b)

Figure 2.5: Architecture of the Reactive Immune Network (RIN) system for mobile robot reactive navigation [60]

## 2.5 Conclusion

This chapter has discussed several computational intelligence techniques that are related to cooperation problem. It then looked into other immune systems inspired cooperation models. In this chapter, it is argued that immune systems based cooperation techniques are applicable in multi-robot systems area bringing with it the advantages that are inherent in the biological immune systems. This also shows that a lot more other techniques inspired from the immune systems can be researched and applied, as only a few features and models of it have been studied.

# Chapter 3

## Immune Inspired Model for Cooperation

### 3.1 Introduction

This chapter discusses immune system inspired model for multi-robot cooperation. Several cooperative tasks that are relevant and suitable to be experimented to test the proposed algorithm are described in general. The new *Immune Network T-cell-regulated—with Memory* (INT-M) algorithm is proposed in this study and it is described in subsequent sections.

### 3.2 Immune Systems Approach

The relationship of the immune systems with multi-robot systems is evident where obstacles, robots and their responses are antigens, B-cells and antibodies respectively. [Table 3.1](#) lists the parallel of MRS and immune systems terminologies.

Immune Network Theory as described in [subsection 2.3.2](#) is suitable as a basis for emulating cooperative behaviour in a multi-robot environment. This is because the immune network uses affinity measures that are dependent on other cells concentration and location in determining the next action. Other than that, multi-robot systems require recognition ability of obstacles and other robots, which is parallel to the immune system recognition and activation phase of an immune response. Obviously in immune network, processing of information is done in

Table 3.1: Relationship between Immune Systems and MRS

Immune Systems	Multi-Robot Systems
B-cell	Robot
Antigen	Robot's Environment
Antibody	Robot's action
T-cell	Control parameter
Plasma cell	Excellent robot
Inactivated cell	Inferior robot
Immune network	Robots interaction
Stimulus	Adequate stimulation among robots
Suppression	Inadequate stimulus from robots

real-time and in a distributed manner, as what a multi-robot system requires.

### 3.2.1 Immunoid: the Immune Network based Robot

One of the earliest works on distributed behavioural model is by Reynolds [85] that focuses on the flocking behaviours of bird-like objects. Reynolds coined the term 'boids' that refers to simulated bird-like or "bird-oid" objects. The study achieved the aggregate motion of a simulated flock that emerges based on interactions of relatively simple behaviours of the individual boids.

In quite the same purpose, the term Immunoid was introduced by Ishiguro et al. [33, 34, 35, 36, 37]. Immunoid is simply defined as an autonomous mobile robot that have an "*immune network-based action selection mechanism*".

Figure is redacted,  
kindly refer [35]  
by Ishiguro, A., Shirai, Y., Kondo, T.,  
and Uchikawa, Y. (1996b)

Figure 3.1: Immunoid: a robot with an action selection mechanism [35]

Although Ishiguro et al. deployed a different approach of immune network in their experiments, the term is very suitable to be used in this study. Each

immunoid acts similar to B-cells in the biological immune systems, but specifically utilising the immune network paradigm in interacting with other immunoids and coping with environmental changes. The use of the term ‘Immunoids’ is suitable to show that the robots are using immune network approach both internally and in interacting with each other.

### 3.3 Immune Network for Group Behaviour

Figure 3.2 shows the state transition of group behaviour in multi-robot systems. The immune network is deployed as the group control algorithm, while each immunoid utilises the Clonal Selection approach for detecting environmental changes, but then communicates and is also affected by other nearby immunoids for action strategies selection phase. The task execution phase is currently simplified; as long as the immunoid is able to find and *carry out* the tasks scattered around them. The overall objective is for the group of immunoids to be able to detect and execute all the tasks in the workspace with appropriate group behaviour selected depending on the changes in local environment. This can be regarded as a general collective search problem. Each phase is covered in more detail in the following sections.

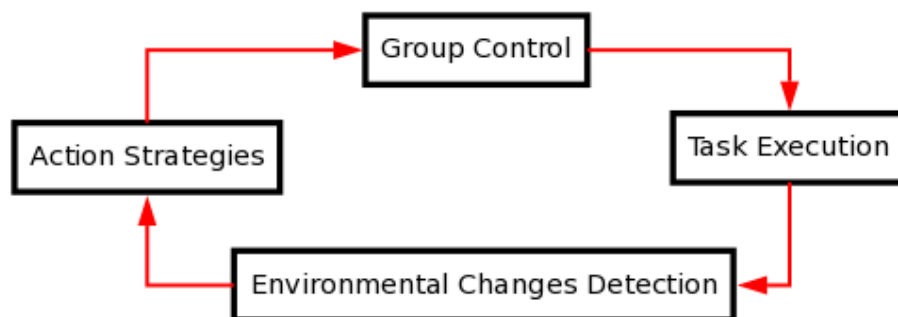


Figure 3.2: State diagram of group behaviour

#### 3.3.1 Definition of Task

For task execution phase, currently the tasks are not detailed out. It can be anything, depending on the application domain. The only requirements are that the objective is to find and *carry out* all the task in the area or workspace. *carry out* is left to as anything, however it obviously needs to be allocated a standard



amount of time to be executed (e.g. an arbitrary value of 15 unit time per task execution).

For the current approach, it is assumed that all immunoids can execute all the task at hand. This can be extended for example only certain immunoids are capable of executing certain tasks or that a few of them need to attend a single task or any variations of these. In the mine detection application using the AISIMAM model [94], each ‘mine’ (i.e. task) needs to be ‘diffused’ (i.e. task execution) by four robots (by simply detecting and going to the task’s location). Furthermore, the allocation of tasks are done instantaneously, therefore there are no planning and scheduling overheads in assigning tasks to robots. Moreover, another assumption for the mine detection problem is that the tasks are static in their location (i.e. not moving about). The element of dynamically changing environment is introduced by placing another set of tasks into the workspace, whereby the immunoids need to adapt to that new situation.

The detail definition of task assignment and execution falls into the research area of Multi-Robot Task Allocation (MRTA) [29], which currently is not the focus of this study. Thus, for this study it is defined as: Single-Task robots and Single-Robot tasks with Instantaneous Assignment of the tasks, or shortened as ST-SR-IA. The total number of task is also global, in the sense that all immunoids have *a priori* knowledge of the total number of task in the experiment (e.g. 500 tasks are spread out in the workspace).

### 3.3.2 Definition of Antigen

As for the antigen, it depends on the environment of the workspace. In this approach, it is considered the density of task distribution that the immunoid have locally detected. This task density is divided into four levels, namely *High*, *Medium*, *Low* and *None*. For each of these environment condition, the immunoid needs to select the appropriate action strategies (i.e. the antibodies). Table 3.2 lists the general relationship of task density being detected and the resulting stimulus value,  $g_i$ .

Therefore, the affinity of antibody  $i$  and antigen (the term  $g_i(t)$  in Equation 2.3 and also Equation 3.1) can be derived by using a stimulus function. An example to get the value of  $g_i(t)$  is by using the stimulus function as shown in Figure 3.3. The simple step function used to assign the antigen to antibody affinity values, i.e.  $g_i$  is as shown in Table 3.3.

Table 3.2: Basic task density and  $g_i$  relationship

Task density	High stimulus value
High	Aggregation, $g_0$
Medium	Searching, $g_1$
Low	Dispersing, $g_2$
None	Homing, $g_3$

Figure is redacted,  
kindly refer [95]  
by Sun, S.-J., Lee, D.-W.,  
and Sim, K.-B. (2001)

Figure 3.3: Stimulus function of antigen to antibody,  $g_i$  [95]Table 3.3: Antigen-antibody affinity stimulus function,  $g_i$  (other index values remain as 0.0)

Task Detected (%)	Task Density	$g_i$ values
(66 – 100]	High	$g_0 = 1.0$
(10 – 66]	Medium	$g_1 = 1.0$
( 0 – 10]	Low	$g_2 = 1.0$
0	None	$g_2 = 1.0, g_3 = 0.5$

The percentage of task detection is calculated from the number of locally detected tasks over the total number of tasks which is known *a priori*. This calculation would need to be done at some standard time interval so that current environment changes are considered in evaluating action strategies. This time interval can arbitrarily be assigned (e.g. every 40 unit times), but should obviously take into account the appropriate interval depending on the scenario at hand.

### 3.3.3 Definition of Antibody

The antibody is defined as the action strategies that are available to the immunoids. After sensing the environment for a specific time-period, the immunoid needs to consider what action strategy is well suited for that current situation. This is when the Clonal Selection approach is executed within the immunoids' internal state, which can use the stimulus function as shown in [Figure 3.3](#).

However, the immunoid needs to consider other local immunoids antibody evaluation. This is the immune network part of the approach. This step is done via communicating the related information with other nearby immunoids. The default antibody which is assigned the highest stimulus value and hence being selected at the beginning is the *Random Search* strategy.

Table 3.4: Mutual stimulus coefficient,  $m_{ij}$

robot $i \setminus$ robot $j$	$\mathbf{Ab}_0$	$\mathbf{Ab}_1$	$\mathbf{Ab}_2$	$\mathbf{Ab}_3$
Aggregation, $\mathbf{Ab}_0$	1	-0.4	-0.2	-0.4
Search, $\mathbf{Ab}_1$	-0.4	1	-0.4	-0.2
Dispersion, $\mathbf{Ab}_2$	-0.2	-0.4	1	-0.4
Homing, $\mathbf{Ab}_3$	-0.4	-0.2	-0.4	1

If two robots have high stimulus values for  $Ab_0$  which is Aggregation mode, then that behaviour is stimulated. The coefficient values of this mutual stimulus,  $m_{ij}$  is shown in [Table 3.4](#) which follows Sun et al. [95]. The coefficients are high (i.e. stimulus), if the antibodies of the two robots,  $i$  and  $j$  are the same. Otherwise, they have low values (i.e. suppressed). These values are assigned arbitrarily and can be changed accordingly depending on the scenario. The degree of stimulation or suppression that is intended can be guided by how we view the similarity or difference of two behaviours. For example, if one robot has Aggregation ( $Ab_0$ ) behaviour and the other robot has Search ( $Ab_1$ ) behaviour, then this should be mutually suppressed (if that is what we intended, such as in a Herding task). If

the scenario is to have the robots cover a wide area (such as in Covering task), then different behaviours from the two robots should be stimulated instead.

The mutual stimulation (or suppression) guarantees that the action being selected is appropriate with the local environment and also an emergent local group behaviour is executed, thus the task execution at that local site is (nearly) optimal. The interaction of antigen-antibody in one immunoid, and antibody-antibody among immunoids is depicted in Figure 3.4. After all the interaction and calculation, the antibody with the highest stimulus value is selected for execution. The values are the same as in subsection 2.4.2, whereby  $m_{ij}$  is the mutual stimulus of antibody  $i$  and  $j$ .  $g_i$  is the affinity of antibody  $i$  and antigen while  $s_i$  is the concentration of antibody  $i$ .  $c_i$  is the concentration of T-cell which control the concentration of antibody and  $\beta$  is the parameter of response rate of other antigens.

Figure is redacted,  
kindly refer [95]  
by Sun, S.-J., Lee, D.-W.,  
and Sim, K.-B. (2001)

Figure 3.4: Immune Network which includes T-cell and B-cell models [95]

### 3.3.4 Group Control Algorithm

The group control or coordination phase is done in a distributed manner via local communication between nearby immunoids. When an immunoid encounters

another immunoid and both have the same or similar strategy, this strategy is stimulated; if not, the strategy is suppressed. This facilitates the group to self-organise towards a common action which is optimal for the local environment. If an immunoid is stimulated beyond a certain threshold which makes it an excellent immunoid, its behaviour is regarded as adequate in the system such that it can transmit its strategy to other inferior immunoids. This is a metaphor of the plasma cell in the biological immune systems. However, there is no central point of control in coordinating the group behaviour.

Figure is redacted,  
kindly refer [48]  
by Lee, D.-W. and Sim, K.-B. (1997)

Figure 3.5: Immune Network-based Cooperative Robots [48]

Figure 3.5 shows a general overview of a possible scenario in an Immune Network-based Cooperative Robots during its execution. The immunoids would detect their own local surroundings for tasks and determine their density, then communicate with other nearby immunoids which can then determine either to stimulate further or suppress the neighbouring immunoids' action selections. This cycle will continue until all the tasks in the workspace are covered.

Algorithm 3.1 shows the general algorithm of immune network that utilise the B-cell and T-cell modelling. The algorithm is for each immunoid. This will then interact with others as and when appropriate.  $S_i(t)$  in the algorithm is the stimulus value of antibody  $i$  at time  $t$  (referring to Equation 3.1), where  $i = 1 \cdots N$ ,  $N$  is the number of antibody types.  $s_i(t)$  is the concentration of antibody  $i$  at time  $t$ .  $c_i(t)$  is the T-cell model that represents the concentration of T-cell at time  $t$ , which control the concentration of antibody.

---

**Algorithm 3.1** General Immune Network Algorithm—for each immunoid

---

**Require:**  $t = 0$ ,  $S_i(0) = s_i(0) = 0.5$  for  $i = 0 \cdots N - 1$ ,  $N$  is number of actions

**Ensure:**  $Ab$  with highest concentration is executed

```

1:  $Ab_{max} \leftarrow Ab_1$  //at start  $Ab_1$  is selected
2: loop
3:   Execute  $Ab_{max}$ 
4:
5:   for  $i \leftarrow 0$  to  $N - 1$  do
6:     Calculate  $S_i(t)$  //refer Eq.(3.1)
7:     Calculate  $s_i(t)$  //refer Eq.(2.4)
8:     Calculate  $c_i(t)$  //refer Eq.(2.5)
9:   end for
10:
11:   if  $S_i(t) > \bar{\tau}$  then //stimulated above upper threshold, refer Eq.(3.2)
12:     immunoid  $\leftarrow$  excellent
           //can transmit  $Ab$  when encounter immunoidinferior
13:   else if  $S_i(t) < \underline{\tau}$  then //below lower threshold, refer Eq.(3.3)
14:     immunoid  $\leftarrow$  inferior
           //receives good  $Ab$  when encounter immunoidexcellent
15:     if immunoid encounter immunoidexcellent then
16:       for all  $i$  do
17:         receive  $Ab_i$  //receives all  $Ab$  from immunoidexcellent
18:         renew  $s_i(t)$  //renews concentration of each  $Ab$ 
19:       end for
20:     end if
21:   end if
22:
23:   if  $Ab_i$  has  $max(s_i(t))$  then //select  $Ab$  with maximum concentration
24:      $Ab_{max} \leftarrow Ab_i$ 
25:   end if
26:
27:    $t \leftarrow t + 1$  //each iteration is standard (e.g. 40 unit time)
28: end loop

```

---

$$S_i(t) = S_i(t-1) + \left( \alpha \frac{\sum_{j=1}^N (m_{ij} - m_{ji}) s_j(t-1)}{N} + \beta g_i(t) - c_i(t-1) - k_i \right) s_i(t-1) \quad (3.1)$$

The stimulus term and suppression term in Equation 2.3 are combined as the second term shown in Equation 3.1, because  $m_{ij}$  is plus (stimulus) or minus (suppression) value.  $m_{ij}$  is referred to in Table 3.4 that adopts the values used

by Sun et al. [95] and can be assigned arbitrarily. As usual,  $i, j = 1 \cdots N$  and  $s_j$  is concentration of other immunoid's antibody.  $\alpha$  and  $\beta$  are parameters of response rate of other immunoid and the environment (antigen) respectively.

Equations 3.2 and 3.3 are the functions and values for the upper ( $\bar{\tau}$ ) and lower ( $\underline{\tau}$ ) thresholds in determining whether an immunoid becomes an excellent (i.e. plasma cell) or an inferior (i.e. inactivated cell) robot. These equations are following Sun et al. [95] that determine if a robot is able to transmit (i.e. influence) other robots (if it is excellent) or be influenced by other robots (if it is inferior). This is because, if a robot has a strategy that is very strong, it should transmit that to others. However, if the robot has low stimulation for all its strategies then it should be following other nearby excellent robots (if there is any). This will ensure that optimal behaviours can emerge for the local situation.

$$\begin{aligned}\bar{\tau} &= \frac{1}{1 + e^{-0.5}} \\ &= 0.622\end{aligned}\tag{3.2}$$

$$\begin{aligned}\underline{\tau} &= \frac{1}{1 + e^{0.5}} \\ &= 0.378\end{aligned}\tag{3.3}$$

### 3.4 Immune Systems Inspired Cooperation Model

This study's proposed approaches are based on Sun et al. [95] algorithm, with the extension of Memory ability so that quick response can be achieved in the future and also Learning in order to provide generalisation.

### 3.4.1 The INT-M Model: Immune Network with Memory

In order to improve the algorithm as described in [algorithm 3.1](#), a specific memory mechanism is proposed in order to retain the appropriate action for a specific environment condition. This mechanism should be introduced after the immunoids have gone through a cycle of action-selection phase since it requires that a previously successful action had been triggered (i.e. the immunoids are either in the activated/normal or excellent state).

[Figure 3.6](#) displays the Clonal Selection process, whereby various B-cells try to identify the antigen. Once the appropriate B-cell is selected, it is activated and multiply (proliferate), so that adequate immune response could be mounted later. The activated B-cells will proliferate and differentiate into Plasma cells that will secrete specific antibodies and Memory cells which will be in the host body for quite a long time [\[20\]](#). These memory cells will act as catalysts in mounting a quick immune response to the same antigen in the future.

Figure is redacted,  
kindly refer [\[20\]](#)  
by de Castro, L. N. and Timmis, J. (2002)

Figure 3.6: B-cell activation and differentiation into Memory and Plasma cells [\[20\]](#)

This approach is termed as *Immune Network T-cell-regulated—with Memory* (INT-M) as it involves modelling the memory part of the biological immune systems. The algorithm is shown in [algorithm 3.2](#) which is an extension of [algorithm 3.1](#) which is to be performed in each immunoid in the group.



---

**Algorithm 3.2** Immune Network T-cell-regulated—with Memory (INT-M)

---

**Require:**  $t = 0$ ,  $S_i(0) = s_i(0) = 0.5$  for  $i = 0 \cdots N - 1$ ,  $N$  is number of actions  
**Ensure:** retain previous  $Ab$  if immunoid is not inferior within similar environment, execute  $Ab_{max}$

- 1:  $Ab_{max} \leftarrow Ab_1$  //at start  $Ab_1$  is selected
- 2: immunoid  $\leftarrow$  inferior //at start immunoid is inferior
- 3: environment  $\leftarrow$  similar //at start environment is similar (i.e. static)
- 4: **loop**
- 5:   Execute  $Ab_{max}$
- 6:
- 7:   //immunoid is activated (normal) or excellent
- 8:   **if** immunoid  $\neq$  inferior **then**
- 9:     //environment sensed is similar to previous
- 10:    **if**  $g_i(t) \approx g_i(t - 1)$  **then** //refer Figure 3.3
- 11:      $S_i(t) \leftarrow S_i(t - 1)$  //use previous Stimulus values
- 12:      $s_i(t) \leftarrow s_i(t - 1)$  //use previous Ab concentration values
- 13:      $c_i(t) \leftarrow c_i(t - 1)$  //use previous T-cell concentration values
- 14:    **else**
- 15:     environment  $\leftarrow$  changed //need to re-evaluate action
- 16:    **end if**
- 17: **end if**
- 18:
- 19:    //immunoid is inferior or environment has changed
- 20:    **if** (immunoid = inferior) || (environment = changed) **then**
- 21:     //use line 5–21 in Algorithm 3.1
- 22:    **end if**
- 23:
- 24:    **if**  $Ab_i$  has  $max(s_i(t))$  **then** //select Ab with maximum concentration
- 25:      $Ab_{max} \leftarrow Ab_i$
- 26:    **end if**
- 27:
- 28:     $t \leftarrow t + 1$  //each iteration is standard (e.g. 40 unit time)
- 29: **end loop**

---

The lines 8–17 in [algorithm 3.2](#) is the added memory part of the algorithm. Its function is to use the previous action for the currently *similar* environment situation. The similarity is evaluated based on the  $g_i(t)$  function whereby [Table 3.3](#) is referred. This extension will enable quicker action-selection process whereby the previous  $S_i(t)$ ,  $s_i(t)$  and  $c_i(t)$  values are used and eliminating the need to recalculate their values. If the current situation is different then the algorithm simply flag the *environment* variable, thus re-evaluating the related equations.

The memory ability is only triggered in immunoids that are activated (i.e. normal) or excellent. Immunoids that are inferior are deemed not suitable to use their previous action as they have low stimulus values or they have used the values

received from other excellent immunoids. Therefore, it is only appropriate that the utilisation of memory is only for those immunoids that are not in the inferior category. An immunoid is considered inferior when it has low stimulation values for all its strategies, as discussed at the end of [subsection 3.3.4](#).

In the biological immune systems, the Clonal Selection process is local whereby although the immune cells are distributed throughout the organism, only cells that are located near the infection site are involved in the process [19]. This is reflected in the proposed INT-M algorithm in which the memory ability is for each immunoid, thus maintaining the appropriate response for its local environment.

Another approach that models the Immune Learning ability in the group behaviour which is named as *Immune Network T-cell-regulated—Cross-Reactive* (INT-X) approach is also proposed in this study. The INT-M algorithm is initially studied and later further refinements to it are proposed as discussed in [chapter 5](#). Therefore, the INT-X approach is deferred since it builds on top of the INT-M algorithm and its refinements. However, the details of the second proposed approach is available in [section 5.5](#).

## 3.5 Cooperative Tasks

The proposed approaches are suited for specific task scenario in order to test the methods. There are various multi-robot cooperative tasks ranging from garbage collecting, formation control, patrolling to shepherding and perimeter detection. The feature that would be needed in a viable and beneficial test scenario are cooperating robots, dynamic elements, no central commands, measurable performance and most importantly can clearly reflect real-world scenarios applicability.

The task scenario that would be related to other real-world scenarios, hence suitable for investigation are the *RoboShepherd* test scenario and the *Perimeter Detection and Tracking* problem. This section discusses how the proposed algorithms are to be implemented in the experimental setup.

### 3.5.1 RoboShepherd

The RoboShepherd task scenario provides a dynamic environment with two types of robots in the scenario, the dogs and the sheep. The dog and sheep problem is

a typical problem in distributed autonomous robotics system. Furthermore, there can be multiple dogs and sheep robots to test cooperative behaviours. The basic shepherding task is described in this subsection.

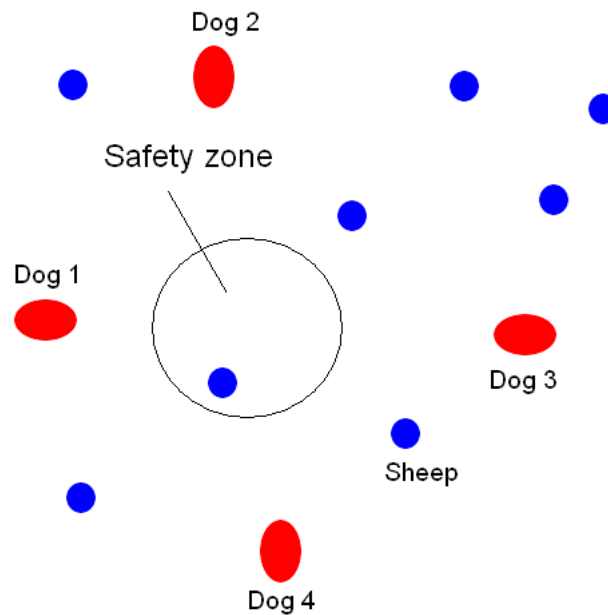


Figure 3.7: The dog-sheep problem scenario, the red ovals are the dogs and the blue circles are the sheep

The objective is for the robot dogs (i.e. the shepherd) to guide the robot sheep into an area called the grazing site or safety zone within a limited amount of time [88]. The robot sheep reacts by performing simple evasion or disperse behaviour to the presence of nearby shepherd providing a dynamic environment. Otherwise, the sheep exhibits random walk behaviour. The robot dogs must control the sheep so that they do not move far away from the grazing site. The number of robot dogs and sheep can be changed.

In this study, the scenario will require multiple robot dogs to perform cooperative behaviour in order to shepherd multiple sheep into the grazing site as shown in Figure 3.7. This is known as the *Collecting* task in the RoboShepherd problem [51]. These shepherds will need to coordinate their action so that optimal and effective group behaviour can be achieved in executing the task. This requires that the robot dogs have the positional information about the sheep in their detecting range, which involves distance and heading or azimuth [50].

This problem is highly dynamic and obviously requires the robots to have real-time processing of partial information of the environment. Our proposed algorithm is based on immune network theories that have many similarities with

the multi-robot systems domain. The research proposes a memory-based immune network that enhances a robot's action-selection process and can obtain an overall a quick group response. The algorithm which is named as Immune Network T-cell-regulated-with Memory (INT-M) is applied to the sheepdog scenario [78, 80].

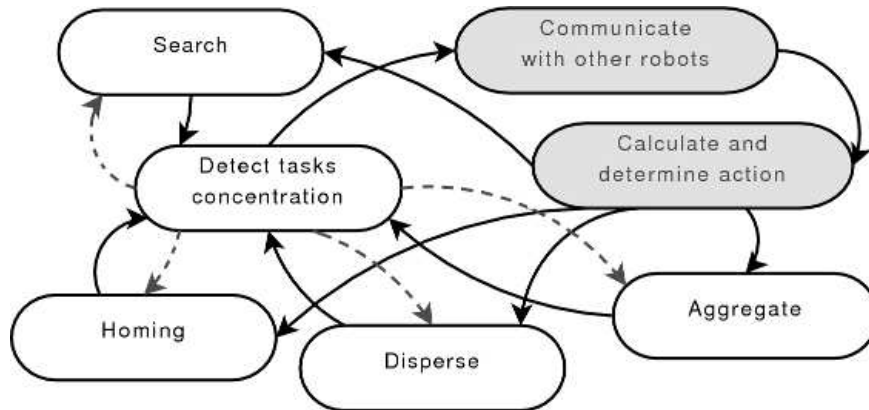


Figure 3.8: The INT-M states, the greyed states are bypassed when the memory mechanism is triggered resulting in the dashed arrow lines

The INT-M model is based upon the work by Sun et al. [95], and it involves modelling the memory part of the biological immune systems. A specific memory mechanism is proposed in order to retain the appropriate action for relevant environment condition. This mechanism is introduced when the newly sensed environment is similar to the previous environment. Thus, a quick action-selection process can be executed without the need of re-evaluating the new situation, as shown in Figure 3.8.

The two proposed models which are Immune Network T-cell-regulated-with Memory (INT-M) and Immune Network T-cell-regulated-Cross-Reactive (INT-X) would require that each robot dogs to use the immune-based algorithms so that they can choose which behaviour to select and communicate with other robot dogs in order to maintain the sheep's progress towards the safety zone. Even though the dog-sheep problem is dynamic in the sense that the target (i.e. sheep) are constantly moving, the environment is sensed at specific time step therefore at each iteration the target would essentially be static. This enables the variables of the INT-M and INT-X algorithm to be evaluated normally. However, the basic behaviour of searching is changed to include pursuit or chasing element. This is shown in algorithm 3.3 which is used in algorithms 3.2 and 5.1, i.e. when the selected antibody is  $Ab_1$ , Seaching behaviour.

---

**Algorithm 3.3** Search strategy with pursuit behaviour

---

**Require:** Search state is currently selected by the immunoid**Ensure:** Exhibit searching & execute pursuit behaviour when sheep is detected

```
1: loop
2:   if no sheep detected then
3:     perform random walking //move in random direction
4:   else
5:     select nearest sheep //determine steering point
6:     perform pursuit behaviour //chase the selected sheep
7:   end if
8: end loop
```

---

### 3.5.2 Cooperative Robots for Perimeter Detection and Tracking

Perimeter detection and tracking is another relevant cooperative task that is suitable to be investigated. It is applicable in several areas, including military (i.e. locating minefields or surrounding a target), nuclear or chemical industries (i.e. tracking radiation or chemical spills), environment (i.e. tracking oil spills), and space (i.e. planetary exploration) [15]. In many cases, humans are used to perform these usually dangerous tasks, but if robotic systems could replace humans, it could be extremely beneficial.

A perimeter is an area enclosing some type of substance. There obviously two types of perimeters, static and dynamic perimeters. A static perimeter does not change over time, an example is a minefield. Dynamic perimeters vary over time that expand or contract over time, like a radiation leak. This task provides quite a challenge for cooperating robots to quickly detect and surround the whole perimeter while it is changing.

In perimeter detection tasks, multiple robots locate and surround a substance, while dynamically coordinating as additional robots locate the perimeter. In this study substances are considered ground-based even though in the real world it can be airborne or underwater. However, there are several limitations to this tasks such as if the perimeter moves with a velocity greater than the robots can move, then the perimeter cannot be tracked. Abrupt perimeter changes is also a limitation because this requires sharp turns that the robots' might not be able to execute as it has limited turning radius [16].

Figure 3.9 shows an example of a perimeter detection and tracking problem scenario. The substance is changing by expanding and contracting at different

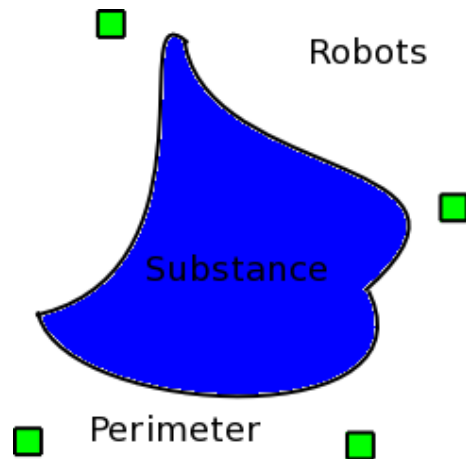


Figure 3.9: The perimeter detection and tracking problem scenario

area. The expansion and contraction rates can be assumed as a constant. Several robots will need to effectively cooperate by spreading out in order to detect the perimeter and then surround the substance. The performance criteria in this scenario is the time limit in detecting and surrounding the whole perimeter; thus ‘*containing*’ it from further expansion (i.e. leak or spill).

The two models, INT-M and INT-X can be deployed in the robots such that the dynamically changing perimeter’s location is considered for evaluating the action-selection phase. The robots will also need to communicate with one another in order to optimally position itself to achieve the objective.

This task provides dynamic challenging environment that requires the robots to be able to quickly detect the environment and concurrently adapt to changes. It is considered that this scenario might be quite complex because quite a lot of changes and adaptation of the algorithm to suit the scenario that need to be done. Therefore, this scenario is not implemented in the study but can be considered in any future works.

## 3.6 Conclusion

This chapter have discussed the immune network based cooperation model introduced by Sun et al. [95] and have proposed a memory extension to it. The approach that utilises memory had been described. Relevant cooperative tasks have been discussed, namely the dog-sheep problem (i.e. RoboShepherd) and the perimeter detection and tracking scenario. The second approach (INT-X) is briefly

mentioned and further discussion is in [section 5.5](#). The INT-M and INT-X model are suited for the relevant tasks but only the INT-M is investigated further. This is because, the research is also focused on investigating any refinements to the behaviours that are related to cooperation. Moreover, the study of a second immune-inspired model would be more appropriate when there is a deep understanding and research done on any proposed refinements that can be identified whether it is from bio-inspired or immune-inspired research.

# Chapter 4

## Experiments & Results of Immune Inspired Models in Multi-Robot Cooperation Tasks

### 4.1 Introduction

This chapter presents the results obtained and discusses the effect of the immune system inspired model applied onto multi-robot cooperative herding. A distinct part of this study is that we are looking into the memory-based immune network cooperation approach by the robots (i.e. dogs) in maintaining the herd (i.e. sheep). This utilises the advantage of memory in the action-selection phase and affects the resulting dynamic behaviour of both the robot dogs and the robot sheep. Further verification of the ‘cooperation mechanism’ is performed to show the ability of the immune inspired approach.

### 4.2 Simulation Setup

A few software tool-kits were tried out, such as the Optimization Algorithm Toolkit (OAT) 1.4 [12]. This is basically a tool-kit for various optimisation algorithms, such as Genetic Algorithm, Ant Colony Optimization, etc. However, the tool-kit primarily focuses on optimisation problems and is not suitable for robotics research.



This research also studied the OpenSteer<sup>1</sup> C++ implementation of Reynolds [86] seminal works in autonomous steering behaviours<sup>2</sup>. The flocking behaviours in this study closely follow the model mentioned in his work. However, the OpenSteer platform is quite outdated hence it is not selected.

Another computational platform that is interesting is the RoboCup simulation software. Robot Soccer World Cup (RoboCup) is a competition of robots playing soccer that started in Japan by the works of Kitano et al. [43, 44]. The competition involves real and simulated robots. For the simulation category, there are 2D and 3D versions of the platform<sup>3</sup>. The 3D simulation platform uses the open sourced SimSpark<sup>4</sup> generic application framework [70]. Another category under RoboCup is the RoboCup Rescue competition which uses simulated robots [45, 96]. However, since this study focuses on cooperation models instead of competitive behaviours thus the RoboCup simulation platforms are not suitable for this research.

This research used the Player/Stage simulation platform on a Fedora 9 Linux operating system [30]. The version being used is Player 2.1.2 and Stage 2.1.1 which are not the latest release of the software but are quite stable releases. The simulation environment is released as an Open Source software. A snapshot of a screen is as shown in Figure 4.1.

The Player/Stage environment is suitable because it supports a wide variety of mobile robots and accessories. Moreover, the Player robot server is probably the most widely used robot control interface in the world<sup>5</sup>. Its simulation back-ends, Stage and Gazebo, are also very widely used. Since they are both Player-compatible, client programs written using one simulator can usually be run on the other with little or no modification. The key difference between these two simulators is that whereas Stage is designed to simulate a very large robot population with low fidelity, Gazebo is designed to simulate a small population with high fidelity. Thus, the two simulator are complementary, and users may switch back and forth between them according to their needs.

Player provides a network interface to a variety of robot and sensor hardware. Player's client/server model allows robot control programs to be written in almost any programming language and to run on any computer with a network connection to the robot. Player supports multiple concurrent client connections to devices,

---

<sup>1</sup><http://opensteer.sourceforge.net/>

<sup>2</sup><http://www.red3d.com/cwr/steer/>

<sup>3</sup><http://sserver.sourceforge.net/>

<sup>4</sup><http://simspark.sourceforge.net/>

<sup>5</sup><http://playerstage.sourceforge.net/wiki/index.php/PlayerUsers>

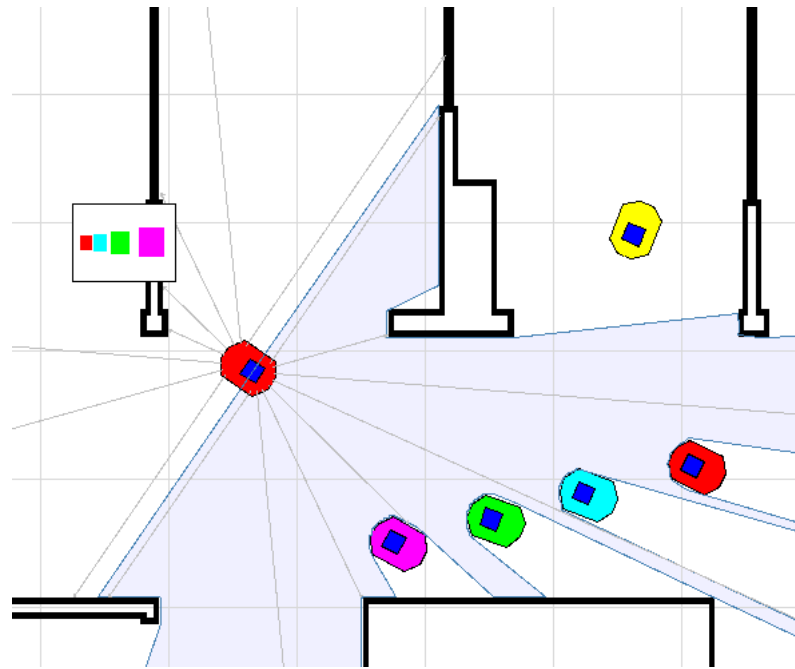


Figure 4.1: Player/Stage: the simulation environment being used

creating new possibilities for distributed and collaborative sensing and control.

Stage simulates a population of mobile robots moving in and sensing a two-dimensional bit-mapped environment. Various sensor models are provided, including sonar, scanning laser rangefinder, pan-tilt-zoom camera with colour blob detection and odometry. Stage devices present a standard Player interface so few or no changes are required to move between simulation and hardware. Many controllers designed in Stage have been demonstrated to work on real robots.

Gazebo is a multi-robot simulator for outdoor environments. Like Stage, it is capable of simulating a population of robots, sensors and objects, but does so in a three-dimensional world. It generates both realistic sensor feedback and physically plausible interactions between objects, which includes an accurate simulation of rigid-body physics.

Furthermore, these platforms are licensed under the General Public Licence (GPL), which means it is free to use, distribute and also modify. This entails that it is possible to include the proposed Immune-based approach in the code repository of the robot control component.

Figure 4.2 depicts the overall interactions between different components; namely the Player server, the Stage and Gazebo simulations platform and robots hardware [46]. The client commands interacts with the Player server via the Transmis-

Figure is redacted,  
kindly refer [46]  
by Koenig, N. (2004)

Figure 4.2: Player/Stage/Gazebo: the big picture [46]

sion Control Protocol (TCP) or User Datagram Protocol (UDP) depending on the usage. The Player server can then interact with either Stage or real robot hardware via TCP; using wired or wireless connections. Simulations can also be executed in three-dimensions on the Gazebo platform using Shared Memory (SHM). The Player server then sends the current data reading of the sensors; real or simulated; back to the client control program for further processing.

In previous sections, we argued that the immune network is a suitable analogy for multi-robot cooperation problems. Experimental data are presented in [sections 4.4](#) and [4.5](#) that validate the applicability and efficiency of the proposed algorithm. As mentioned in the final chapter, the study could be continued in this area, whereby the robots tasks can be appropriately changed to suit other application domains. Other than that, another future work that could be performed is to transfer the simulation experiment onto mobile robots for further investigations.

### 4.3 RoboShepherd Test Scenario

The selected scenario was explained in general in [subsection 3.5.1](#) where only the Collecting task of the RoboShepherd scenario is considered in this study. Several modifications were made from the original RoboShepherd scenario introduced by Schultz et al. [88]. A few assumptions were also made to simplify the simulation, such as the sheep will stop once it arrived in the central grazing site.

Subsection 4.3.1 describe the scenario setup for the simulation experiments that were done.

### 4.3.1 Scenario Setup

The experiments are done for shepherding 2, 5 and 8 numbers of sheep. The shepherding behaviours are the immune-based and the local shepherding behaviour. In local shepherding, the robot dogs will only chase the sheep within its range and do not have any cooperation mechanism. The range for the robot dogs are set to 5 metres for forward sight (i.e. laser) and 20 metres for emulating the sense of hearing (i.e. communication radius).

Table 4.1: Simulation setup for RoboShepherd scenario

Features	Details
Number of sheep	2, 5 & 8 (Colour: red)
Number of dogs	always 4 (Colour: blue)
Dogs' sensor	5 metres forward laser
Dogs' communication	20 metres radius
Area / Field	40x40 metres (walled)
Grazing site	centre with 5 metres radius (sheep will stop)
Time limit	5 minutes
Shepherding behaviours	Immune-based & Local behaviour

The field is constructed of a walled field with the size of 40 metres each side. The grazing site is situated at the centre with a radius of 5 metres and each sheep that have entered it will stop. Each experiment is limited to a limit of 5 minutes and it is done for three times where the average values are then calculated. Positions of both sheep and dogs are random for each trials. Table 4.1 summarises the simulation setup for the RoboShepherd test scenario.

Figure 4.3 is a snapshot of one of the experiment done that shows the limited behaviour of local shepherding. Other robot dogs do not sense the sheep that is outside of the grazing site. The one robot dog that is chasing the particular sheep is doing all the shepherding, which is not optimal as a group.

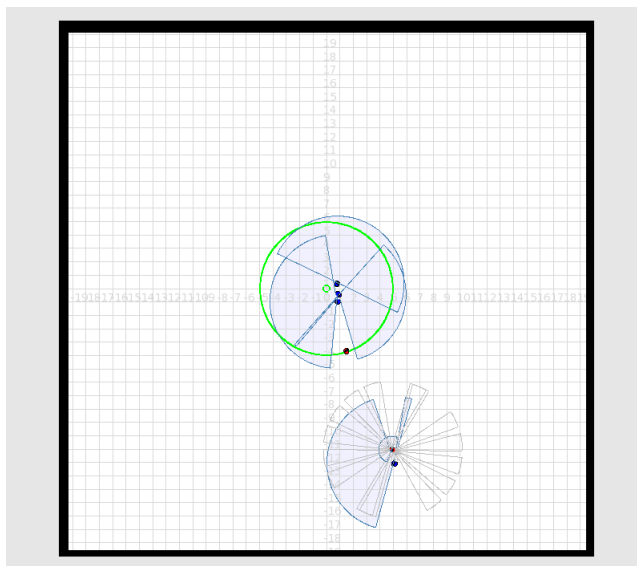


Figure 4.3: The simulation experiment: involving 2 sheep (red)

## 4.4 Simulation Results

The performance is measured using two aspects. The average time of the first sheep that is shepherd into the grazing site (which is known as Time for Completion), and also the number of sheep left in the field (which is known as Incomplete Task) after the maximum time is up. The reason that the first sheep is chosen is because it is anticipated that there might be situation whereby the time it takes to herd all the sheep into the grazing site would be too long. Therefore, the first sheep is used to signify how quick the sheep can start to complete the overall task.

### 4.4.1 Average Time for Completion

The average time for completion is shown in [Figure 4.4](#), where the point for local behaviour with 8 sheep is not plotted because all sheep are unable to be shepherd into the grazing site by using that approach. This result shows that the immune-based approach can scale better compared to the local behaviour. The results also show that too few sheep is not optimal, because they tend not to flock once they are separated (this can be seen during the run of the experiments). Meanwhile in the five sheep scenario, the sheep usually will either be in a big or small flock thereby posing an easier shepherding task to the robot dogs. On the other hand, if the ratio of sheep is too high compared to the available shepherd, the task becomes more complex. But as shown in [Figure 4.4](#), it is still manageable for the

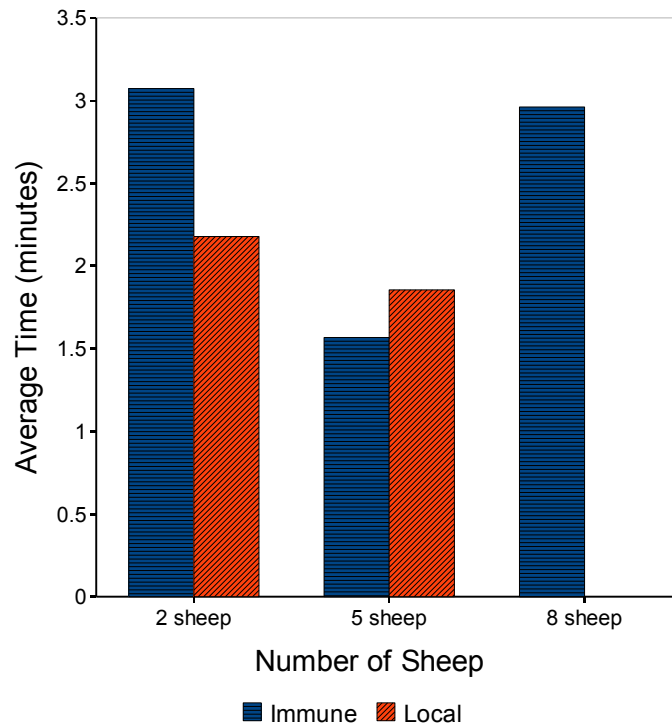


Figure 4.4: Average Time for Completion

immune-based method but not so for the local behaviour approach.

The number of sheep of 2, 5 and 8 were chosen to represent three types of shepherding complexities. Two sheep in a flock to represent a small sheep-to-dog ratio (i.e. 1:2) while five sheep representing an average complexity. Meanwhile, eight sheep is to represent a highly complex dynamic situation where there are more sheep than dogs, i.e. an underactuated scenario. It is considered that these three numbers are suffice to represent the types of complexities that can occur for such a situation.

#### 4.4.2 Average Number of Incomplete Task

Another important performance to consider is the average number of incomplete tasks that signify the ability to maintain the balance of the overall goal of shepherding all the sheep and also completing it in a short time. Figure 4.5 shows that the immune-based approach has lower average incomplete task as the number of sheep gets larger, meanwhile the local shepherding totally fail when the number of sheep is set to eight which represents a highly complex situation. This result together with the result explained in the previous section signifies that the immune network cooperation in shepherding on average can achieve better completion time

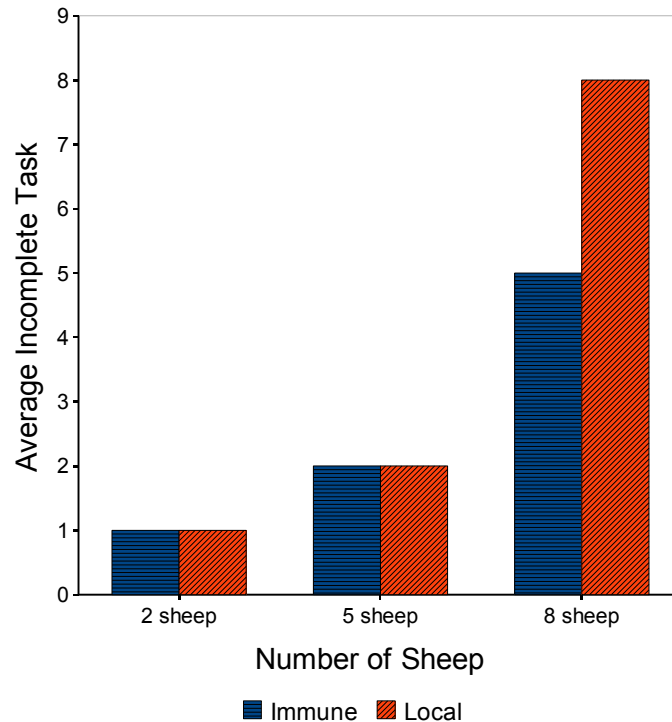


Figure 4.5: Average Number of Incomplete Tasks (Time limit: 300 seconds)

without sacrificing the overall goal (i.e. having low rate of incomplete tasks). The result in Figure 4.5 is based on a time limit of 300 seconds. It is expected that the immune-based approach would have a much lower average incomplete task rate if the time limit is set to be higher.

This subsection shows that the use of immune network cooperation in a shepherding task is quite robust to the sheep-to-dog ratio. The immune-based behaviour can still perform although the number of sheep is twice the number of dogs. It must be noted that the type of shepherding being studied is the *Collecting* task which is inherently complex as mentioned in subsection 3.5.1. The number of sheep selected follows the previous subsection for the same reason, which is to represent the three types of complexities that can occur in such a scenario.

### 4.4.3 Discussions

An observation from these results is that local shepherding behaviour would fail when the number of sheep increases. Also, the Immune-based approach shows that it is quick but without sacrificing overall goal of herding the sheep. A good shepherding performance would always maintain the overall goal of shepherding all of the sheep into the grazing site.

One difficulty of the scenario that have been met is that the sheep distribution in the field is randomly scattered. Therefore, the dogs task had been scattered all over the area, trying to herd the nearest sheep (in relation to the grazing site). This is known as a Collecting task category, similar to a garbage collecting task (with a difference that the ‘garbage’ dynamically moves around) [51, 53].

Finally, the performance criterion can be diversified in order to obtain a more holistic comparison of the approaches. Some of the performance criteria that might be useful are:

1. the average distance of the flock for each time-step
2. the number of communication messages required by the robot dogs
3. the power/energy consumption of the robot dogs to complete the tasks

The average distance of the flock can show that the herding was well maintained by the robot dogs, hence signifying a good herding approach. The communication complexity can show the cost of achieving the cooperative shepherding by the robot dogs. The same goes for the energy criterion, whereby this cost can be taken into consideration for evaluating the performance of a particular shepherding approach. The first performance criteria above is used in the experiment in [section 5.3](#).

## 4.5 Verification of the Immune Inspired Cooperative Mechanism

This section is to test the underlying immune inspired cooperative mechanism, with regard to the stimulation and suppression of antibodies amongst the group of robots. Simulation experiments without using the Player/Stage platform were conducted to verify the proposed cooperation mechanism.

In this test, there are four robot dogs with no sheep involved. The presence of sheep (i.e. the percentage of task detected) are hard coded into the robot dogs. This is because, this test is to verify the underlying immune inspired cooperative mechanism, specifically their response to environmental changes and whether the robot dogs can influence (i.e. transmit their strategies) one another.

The values for the constants are  $\alpha = 0.3$ ,  $\beta = 0.05$   $\eta = 0.05$  and  $k = 0.002$  which follows Sun et al. [95] values, except for  $\eta$  which is our own value. Since



$\alpha$  and  $\beta$  are response rates (of other antibody and antigen), therefore it is set to a low number. It can be increased if we want to mimic a quick stimulation (or suppression) rates, but it would then be not realistic for a real world problem. This is the same for  $\eta$  which is a constant in calculating the concentration of T-cell,  $c_i$ . The value of  $k$  should be very low since it represents the natural extinction coefficient of the immune cells. At the start of simulations, the values for  $g_i$  are set to 0.0 except for  $g_2$  (Dispersion) and  $g_3$  (Homing) are assigned 1.0 and 0.5 respectively. This is because, we want the robot dogs to initially disperse so that they can find more sheep and not group together, hence they can shepherd more sheep as a whole.

Robot 4 starts with not seeing any of the tasks (i.e. percentage of tasks detected is 0.0%), although assumption is made that all robots are within each others' communication range. This may happen for instance when robot 4 is facing another direction from the rest of the group. Meanwhile, the other robots are assumed to have already detected 75.0% of the task at start time. Furthermore, it is assumed that all robots remain geographically static over time. This is in order to prevent the robot dogs from being out of each others' communication range, since the purpose of this test is to verify the workings of the underlying cooperative mechanism.

Figures 4.6–4.9 display the average for each antibodies' concentration value (i.e.  $s_i$ ) over time. The antibody (i.e. strategy) with the highest concentration (i.e. maximum value) of  $s_i$  will be selected by the robot to be executed.

### 4.5.1 Response to Environmental Changes

In order to test the response of robot 1–3 towards changes in its environment, all of the robots' tasks detected values are changed to 0.0% at  $t = 50$ . Figure 4.6 shows the effects of this, whereby slower increase of robot 1–3  $Ab_2$  (Dispersion) value and the gradual decrease of their  $Ab_0$  (Aggregation) value can be seen. This is due to the fact that only robot 4 is influencing this behaviour to the other three robots.

For testing the response of robot 4 to environmental changes, the task detected of all robots are assigned to 75.0% at  $t = 50$ . Figure 4.7 displays a steeper and faster increase of  $Ab_0$  (Aggregation) and decrease of  $Ab_2$  (Dispersion) respectively. This can be seen in Figure 4.7 that from the time of intervention at  $t = 50$  the concentration of antibody,  $Ab_0$  for robot 4 increase to 1 at around  $t = 70$ . On the

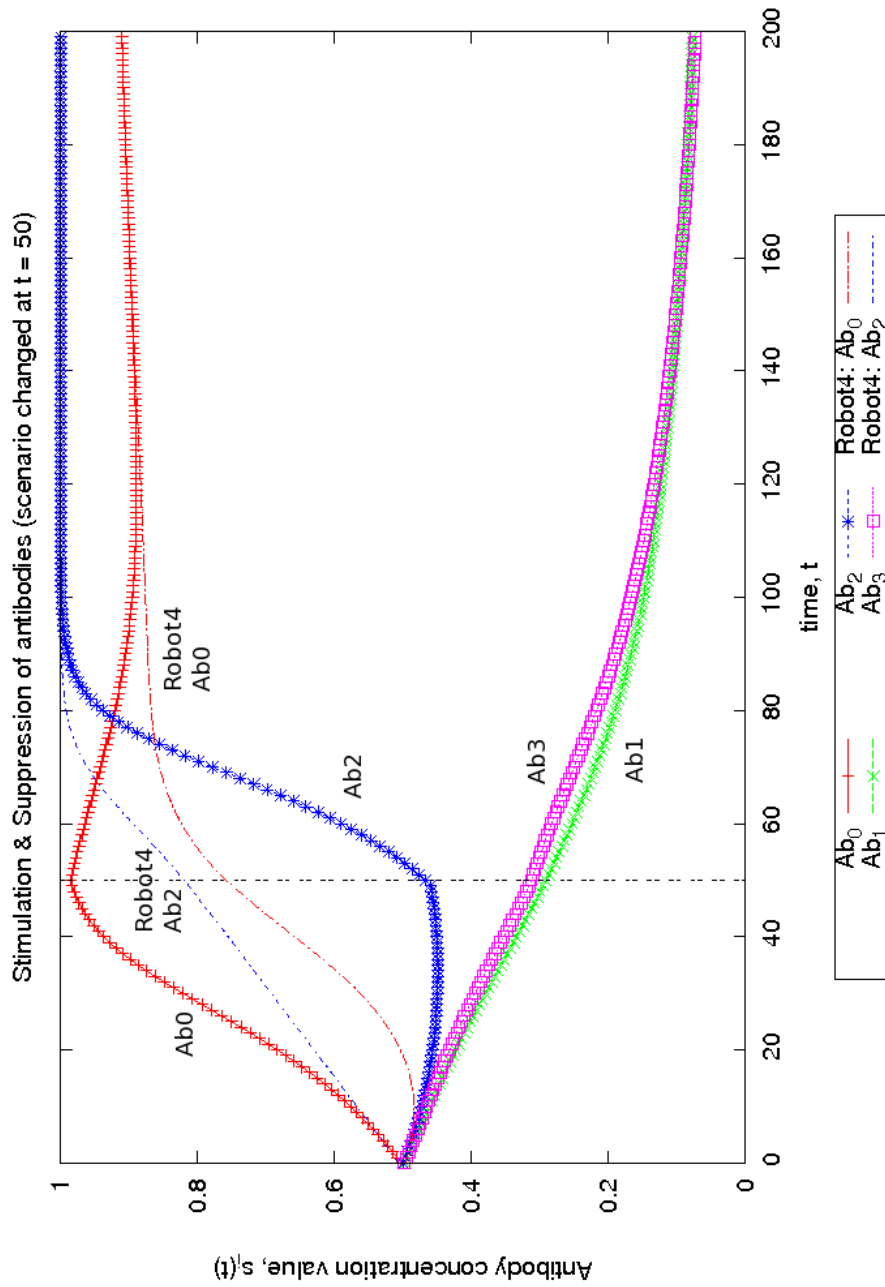


Figure 4.6:  $Ab_0$  to  $Ab_3$  are the average of robot 1-3, which start with high task density then changed to 0 density at  $t = 50$

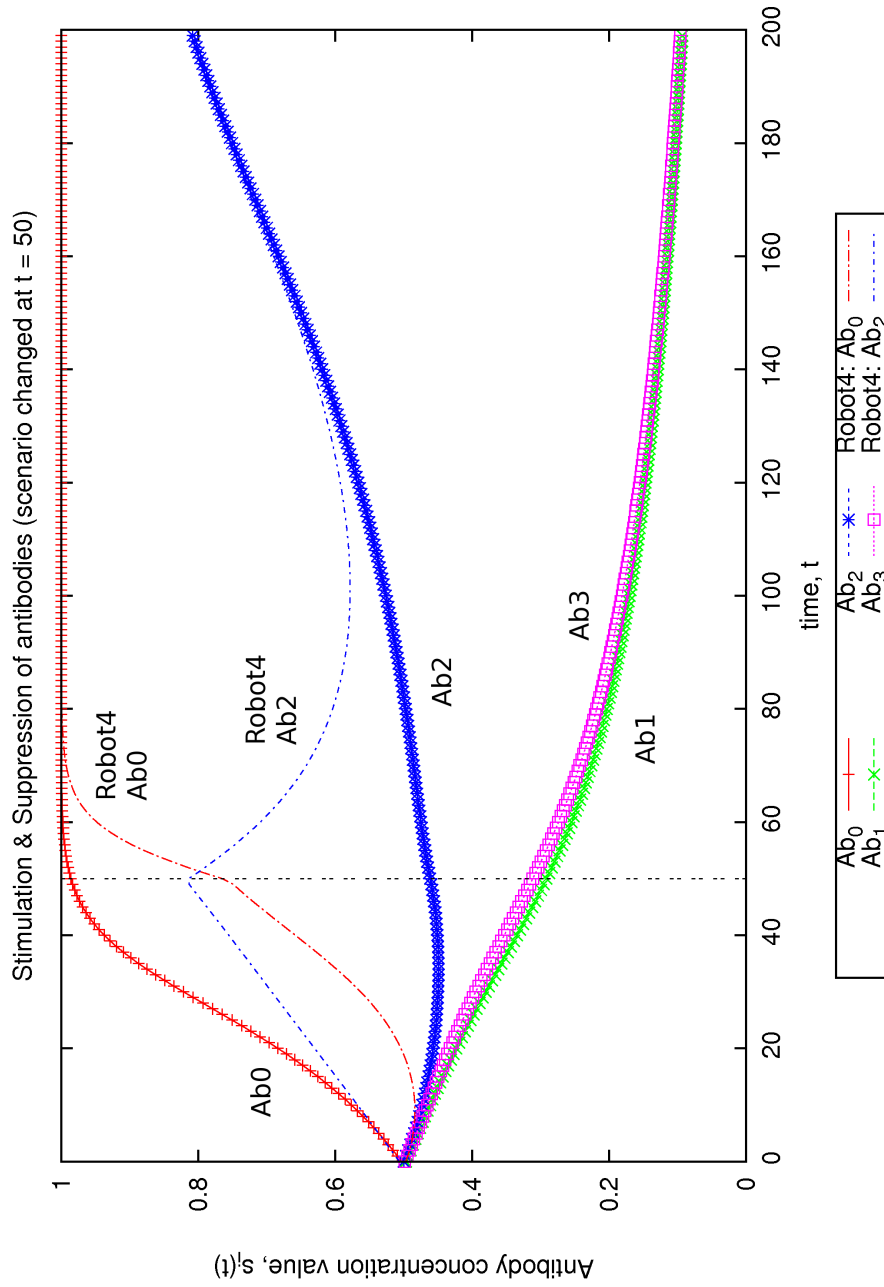


Figure 4.7:  $Ab_0$  to  $Ab_3$  are the average of robot 1–3, robot 4 starts with 0 task density then changed to detect 75.0% of the task (like the other robots) at  $t = 50$

other hand, the concentration of antibody  $Ab_2$  for robot 4 decrease significantly within that time range. This signifies a higher level of influence onto robot 4 by the other three robots. This is known because, the average concentration for antibody  $Ab_0$  is very high and all the robots are within each others' communication range that enables them to influence each other. In real applications, this means that when there are changes in the environment the robot dogs will adapt accordingly to achieve optimal local group behaviour.

## 4.5.2 Propagation of Stimulation and Suppression of Antibodies

Simulations are run to evaluate the propagation of stimulation and suppression of various antibodies among the group of robots. These will show that the local group behaviour is propagated within the neighbourhood. The idea for this test is that, if neighbouring robot dogs have chosen  $Ab_0$  (Aggregation) they can strongly influence robot 4 which currently chooses  $Ab_2$  (Dispersion) if that robot 4 is in an inferior state. The situation for this test is the same as mentioned in the early part of this [section 4.5](#). In [Figure 4.8](#) robot 4 gradually becomes excellent, then at  $t = 50$  it is set to be inferior. The definitions of excellent and inferior states have been discussed at the end of [subsection 3.3.4](#), whereby a robot dog becomes excellent when any one of its strategy (i.e. antibody) stimulus value (i.e.  $S_i$ ) is above 0.622 and it becomes inferior when it has none of its antibody stimulus values are beyond 0.378 (see [Equations 3.2](#) and [3.3](#)). The figures in this section show the antibody concentration value,  $s_i(t)$ . The antibody stimulus values are used to track the state of excellent or inferior robots during the simulation experiment.

The figure shows that in almost instantly robot 4 receives the 'better' strategy (Aggregation,  $Ab_0$ ) from the other robots. The other robots can correctly sense the task. However, since robot 4 local task detected remains 0.0%,  $Ab_2$  is still stimulated. Robot 4 eventually becomes excellent again and thus selects  $Ab_2$  (Dispersion) once more; as it would much more 'believe' what it can sense. This happens at  $t \approx 110$  as shown in [Figure 4.8](#).

[Figure 4.9](#) shows as robot 4 gradually becomes excellent, it continues to choose (i.e. 'believe')  $Ab_2$  (Dispersion) strategy; which is suited to its locally sensed environment (i.e. no task detected). It remains to focus on it's locally sensed environment, however its  $Ab_0$  (Aggregation) is highly stimulated because of the propagation of this strategy from the other robots. The other robots'  $Ab_2$  strategy is

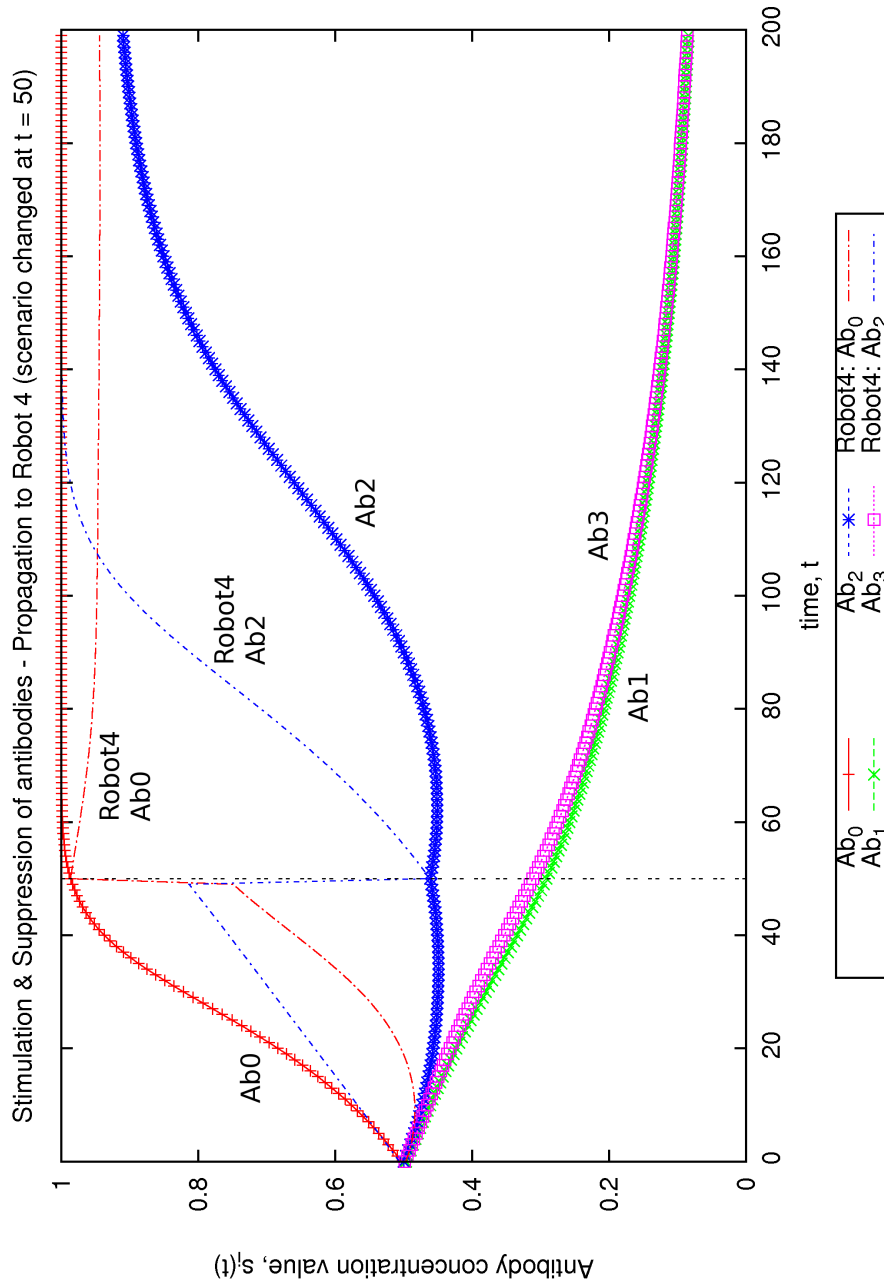


Figure 4.8: Robot 4 becomes excellent over time, then changed to be inferior at  $t = 50$  thus almost instantaneously receives the strategy (i.e.  $Ab$ ) from the other robots; which in this case is  $Ab_0$  (Aggregation); but later changed back as it returns to be excellent once again

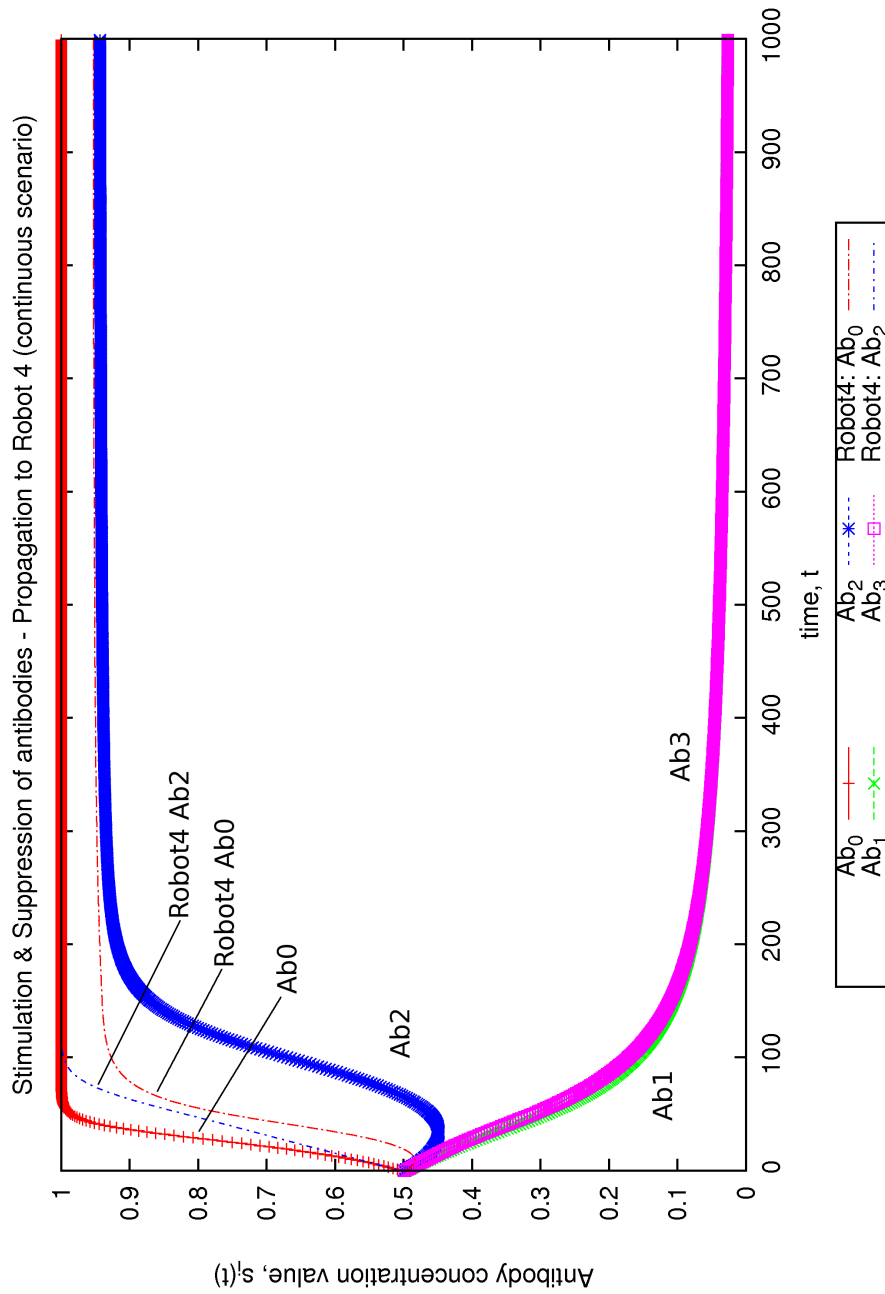


Figure 4.9: Over time, robot 4 becomes excellent and continues to do so as the environment has not changed; thus it maintains it's strategy of  $Ab_2$  (Dispersion). At the same time, its  $Ab_0$  strategy is highly stimulated via propagation by other robots

also stimulated.

## 4.6 Conclusion

This chapter has discussed about the robotics simulation being used, and described the test scenario that is implemented. The simulation results on RoboShepherd test scenario as discussed in [section 4.4](#) have shown that the INT-M model is feasible to be implemented and used in multi-robot systems. Furthermore, the underlying ‘cooperation mechanism’ of the INT-M model has been verified. The next chapter will discuss about the refinements of the cooperative shepherding behaviours in order to have a better performance of multi-robot systems.

# Chapter 5

## Cooperative Shepherding Refinements

### 5.1 Introduction

In [chapter 4](#), the immune inspired model of INT-M had been experimented. The underlying cooperative mechanism had also been discussed. From the simulations done, there are several limitations of multi-robot cooperative behaviour that had been identified. These limitations are related to how the shepherds navigate and determine steering points in order to push the flock towards the safety zone.

These limitations affects the behaviour of the shepherds as a group such as several shepherds competing with each other in order to arrive at the same steering point. It is much better if multiple shepherds cooperate in herding the flock in an organized way, such as forming a line behind the flock. Other than that such limitations also affect the behaviour of the sheep. For example if the shepherds move too near towards the flock of sheep, then there is higher possibility that the flock would get separated thus making the task of herding the flock more difficult.

This chapter discusses about the refinements proposed in the cooperative shepherding behaviour. These refinements are needed in order to achieve better shepherding. There are three refinements proposed in [section 5.2](#) which are Shepherds' Approach, Shepherds' Formation and Steering Points' Distance that are discussed in [subsections 5.2.1](#), [5.2.2](#) and [5.2.3](#) respectively. Other than that, further refinements had been identified afterwards which is regarding flock identification that is discussed in [section 5.4](#). Furthermore, another proposed approach of modelling



Immune Learning ability into the group behaviour is also discussed in the *Immune Network T-cell-regulated—Cross-Reactive* (INT-X) approach in [section 5.5](#).

## 5.2 Shepherd Behaviour's Refinements

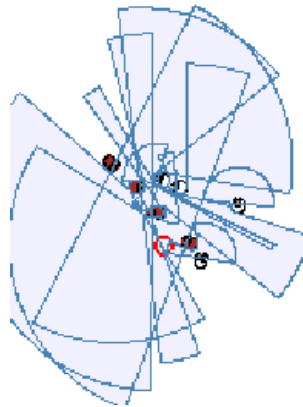


Figure 5.1: An example of the refinement of low-level shepherding behaviour: robot dogs lining-up (the grazing site is located at the top-right corner)

Multiple shepherds pose a few underlying problems regarding the interaction between the shepherds and the flock [53]. For example, flock separation can often occur simultaneously at different parts of the flock when disturbed by several shepherds. This makes it hard to control the flock and achieving the overall goal of herding it. The task of multi-robot shepherding requires inherent cooperation in which to achieve the objective, each robots in the team depends on the actions of one another.

The proposed refinement of the INT-M model discussed in this section includes Shepherds' Approach, but mainly focused on the Shepherds' Formation and Steering Points' Distance aspects. These three refinements are then applied onto the dog and sheep scenario.

The formation involves the robot dogs to line-up behind the group of sheep so that the flock can be better controlled. [Figure 5.1](#) is the depiction of the proposed refinement of the approach by having the robot dogs forming a line behind the group of sheep. The basic lining-up formation is shown in [Figure 5.2](#), where the red marker is the imaginary centre of the flock that needs to be herded.

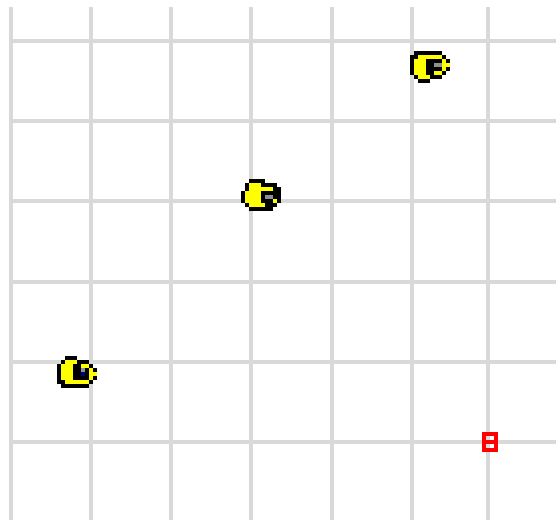


Figure 5.2: An example of the robot dogs lining-up; the red marker is the imaginary flock centre (the grazing site is located at the bottom-right corner)

### 5.2.1 Shepherds' Approach: Safe Zone

The shepherds' approach towards the flock of sheep is also refined by making the robot dog to obey an imaginary safe zone of the sheep. This is in order that the sheep would not be too highly influenced by the incoming dog and resulting in the sheep being separated. This is depicted in Figure 5.3 whereby Dog 2 is trying to go to its steering point, but resulted in separating the flock. However, for Dog 1, it obeys the safe zone of the sheep in the flock thus resulting with a curved path towards its steering point.

This first refinement is achieved by setting a threshold value so that the shepherds do not get too near to the sheep. The safe zone of the sheep has been set to 0.5 metres radius. This results in a lower flock separation occurrences, thereby having better shepherding behaviour.

### 5.2.2 Shepherds' Formation: Lining-up

This section discusses about the experiments done in order to choose between three available lining-up methods as mentioned in Lien et al. [53] that are Global Distance Minimisation, Vector Projection and Greedy Distance Minimisation. These three methods are ways that the shepherds can be assigned to the steering points on the line behind the flock, in order to effectively herd that flock. This is important as to achieve quicker time and shorter distance travelled by the shepherds to reach their designated steering points. Moreover, the method chosen should min-

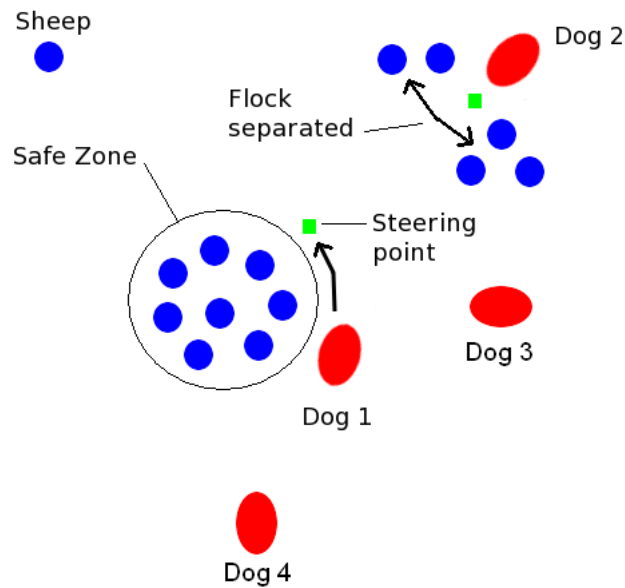


Figure 5.3: Illustration of the use of safe zone in shepherds' approach and the occurrence of flock separation when it is not used

imise interference between the shepherds during their travel to their designated steering points, as this may affect the shepherding behaviour.

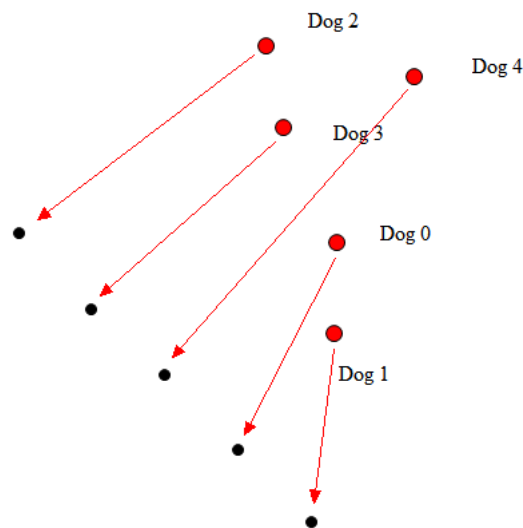


Figure 5.4: An example of Global Distance Minimisation lining-up method

The Global Distance Minimisation is to get the most shortest total distance travelled by all the shepherds. However, this means that there may be a time overhead in doing the calculation. In Figure 5.4, to overall total distance of the shepherds is minimised such that Dog1 is assigned to the bottom-most steering point. In the Vector Projection method, steering points are simply assigned to shepherds according to the matching of position from left to right. This is just

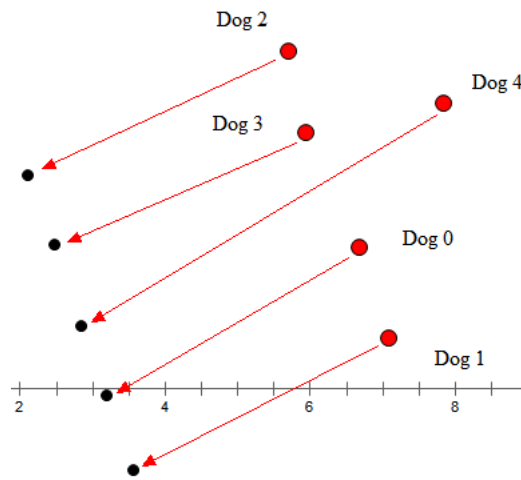


Figure 5.5: An example of Vector Projection lining-up method

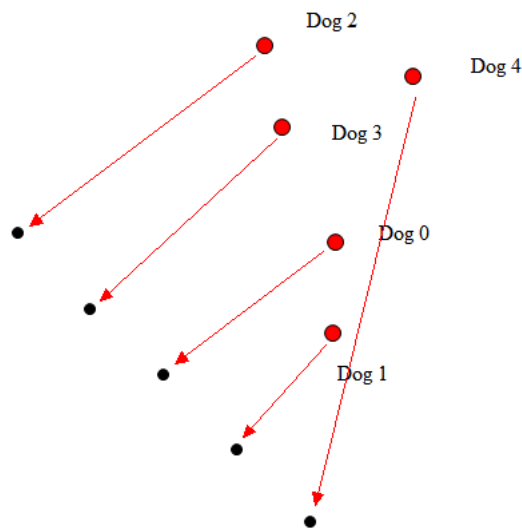


Figure 5.6: An example of Greedy Distance Minimisation lining-up method

like when a group of people going towards a row of seat, where usually one will sit on the chair that aligns to him or her, based on the number of seats and person in the group. This method produces less interference amongst the shepherd during travelling, and can be seen in [Figure 5.5](#). The last method is the Greedy Distance Minimisation whereby each shepherd will go to the nearest available steering point. The method is simple but may cause disturbances to the shepherds during their travel towards their designated points, as the routes may intersect one another [[53](#), [57](#)]. In contrast to [Figure 5.4](#), Dog1 is assigned to the steering point that is nearer to it (i.e. greedy), as can be seen in [Figure 5.6](#).

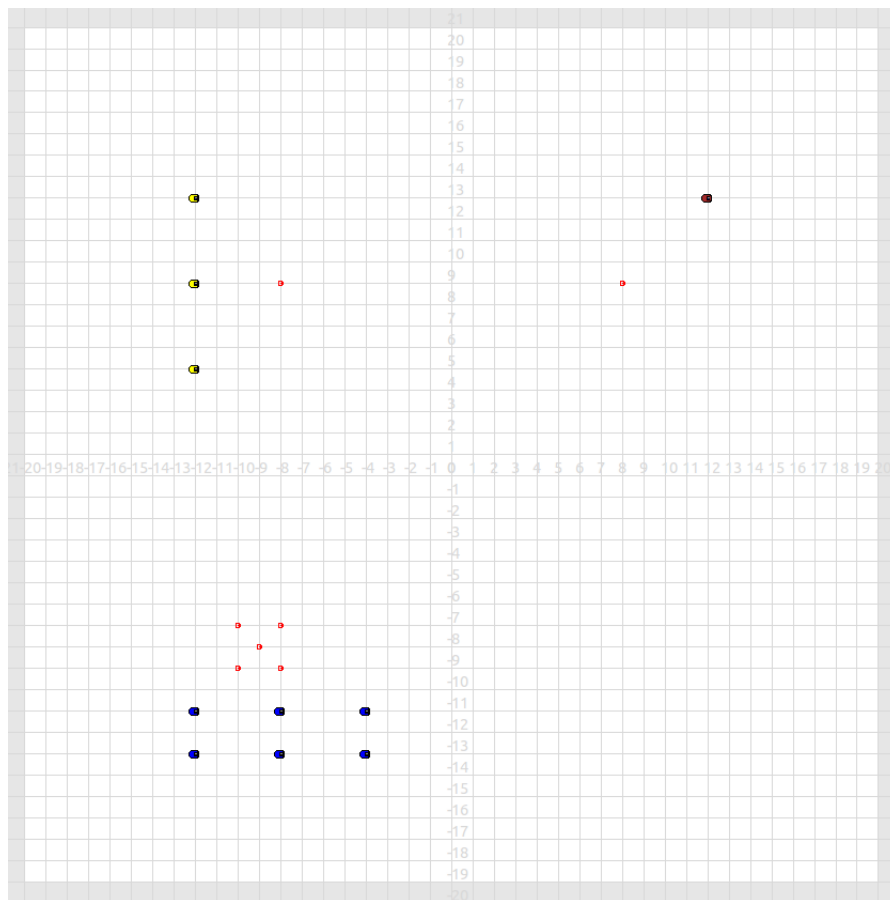


Figure 5.7: A graphical depiction at the start of the lining-up experiment; red markers are steering points replacing sheep, and there are blue, yellow and maroon robots to distinguish each team (both the shepherds and steering points are randomly positioned; the grazing site at the centre of the field)

This experiment is done using C++ on the command line in order to verify the shepherds' formation performance. It is done using four randomly position shepherds and a line where the steering points lies, which is also random. The workspace size is 40x40 metres and each method is repeated for 1,000 times in the simulation. Graphical depictions of this experiment is shown in [Figures 5.7](#) and [5.8](#), which respectively display a before and after example situations.

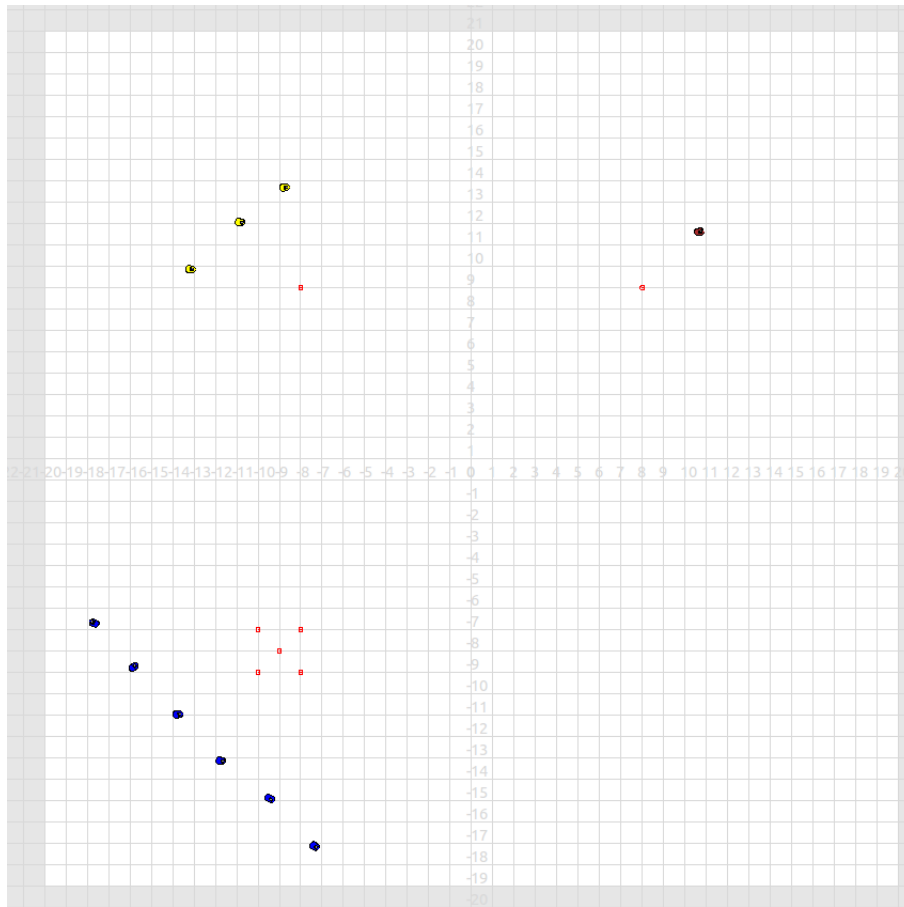


Figure 5.8: A graphical depiction after the lining-up experiment

Table 5.1: Summary of result for distance travelled, in metres (4 shepherds, 1000 iterations)

Method	Mean	Standard deviation
Global	91.41	23.72
Vector	92.50	23.60
Greedy	92.56	23.85

The summary of results in terms of total distance travelled is shown in [Table 5.1](#). The Global method achieved less distance, and the Vector Projection method did quite well. It can be seen that the performance of the three methods did not differ greatly.

However, as shown in [Table 5.2](#) the Global method fared poorly in terms on time taken to calculate the assignment of steering points for the line formation. In contrast, the Vector Projection method achieved quicker time. Therefore, the method that have been chosen for the Shepherds' Formation refinement is Vector Projection because of its low time overhead and a reasonable distance required for the shepherds.

Table 5.2: Summary of result for time taken, in seconds (4 shepherds, 1000 iterations)

Method	Mean	Standard deviation
Global	0.01687	0.004932
Vector	0.00130	0.003394
Greedy	0.00177	0.003819

### 5.2.3 Steering Points' Distance

The final refinement to the shepherding behaviour is in terms determining the best steering point's distance. This is to minimise the interference between the shepherds when they try to arrive at their designated points in order to form the line. This affects the herding efficiency to the shepherds in the long run, as they are moving about to stay clear of each other if the target points are not optimum. This is shown as *Distance Dog To Dog* in [Figure 5.9](#).

This experiment uses the three shepherds' formation methods on the robotics Player/Stage simulation platform. There are four shepherds involved in this experiment with a pre-determined line formation position assigned in the 40x40 metres workspace. There a four steering points distances that are evaluated which are 1.0, 1.5, 2.0 and 2.5 metres. Each lining-up method and steering points' distances are repeated 100 times with a 60 seconds limit for each run. The experiment is to choose an optimal steering points' distance between the shepherds in the line formation so that better group shepherding behaviour can be achieved.

The results for time-step required to achieve the line formation as shown in [Figure 5.10](#) is quite high when the steering points' distances are set to 1.0 and 1.5

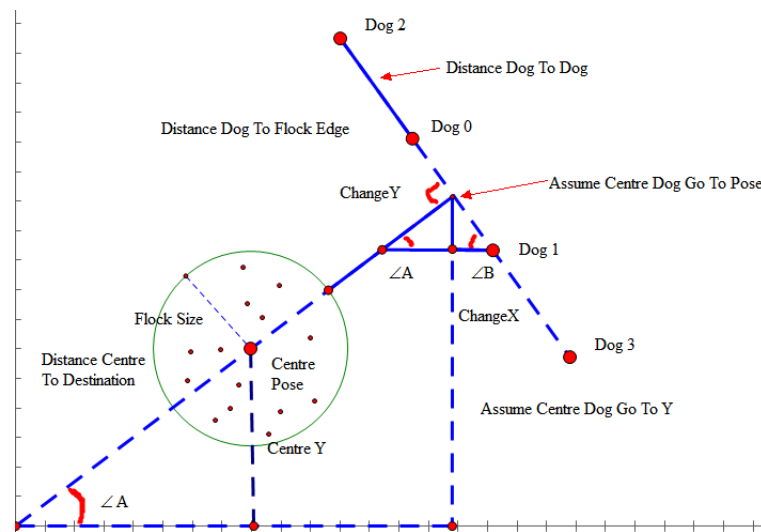


Figure 5.9: The steering points' distances in an example line-up formation

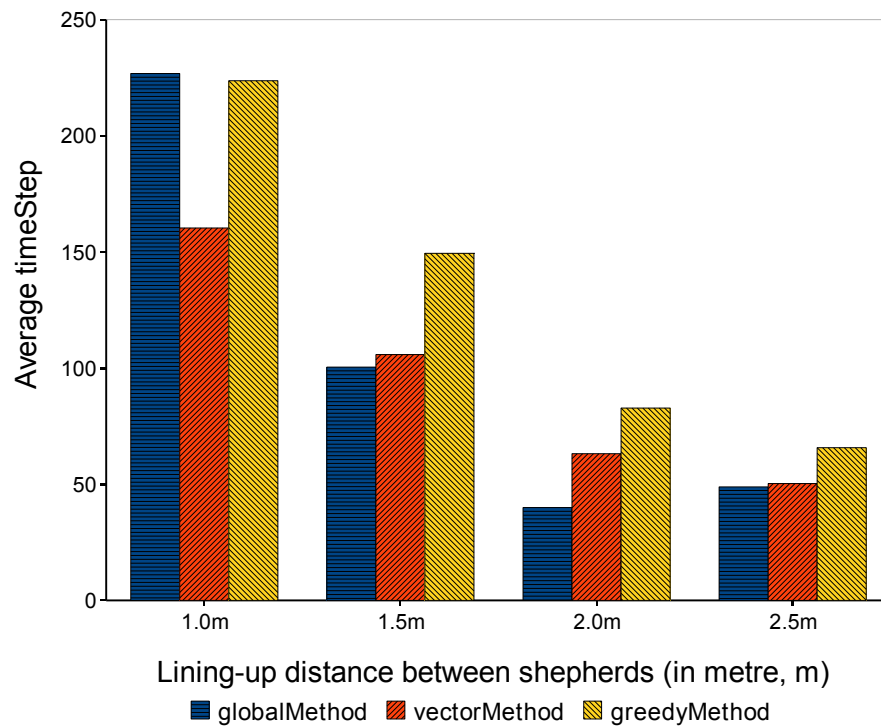


Figure 5.10: Comparison of the steering points' distances using the three different methods in terms of timeStep



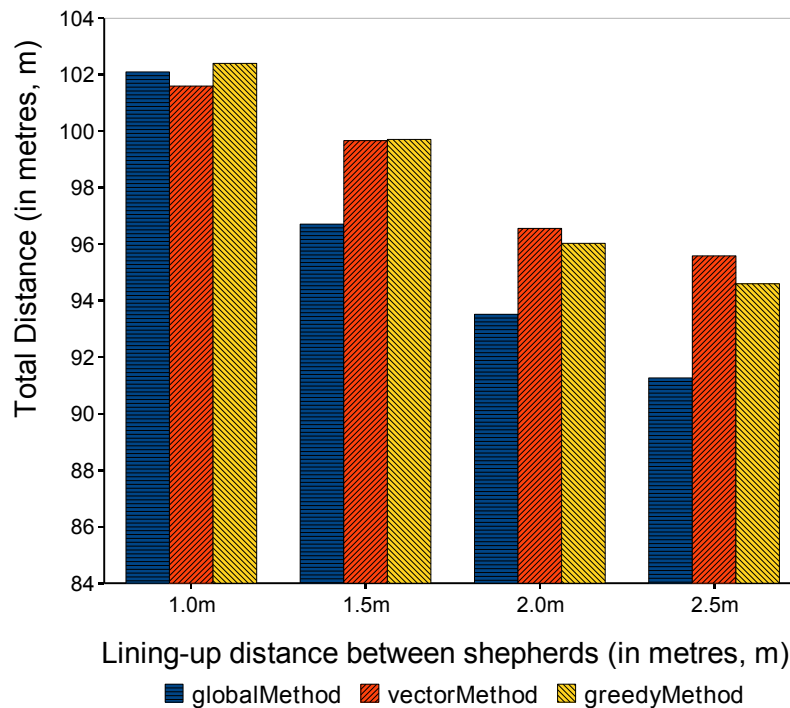


Figure 5.11: Comparison of the steering points' distances using the three different methods in terms of total distance travelled

metres. This may largely be due to the high chance of a situation where the shepherds need to avoid each other during their movement towards their designated points.

Meanwhile in terms of the total distance travelled, the results as shown in [Figure 5.11](#), also suggests that the better option of steering points' distances are either 2.0 or 2.5 metres, since less than 2.0 metres may incur more distance to travel by the group of shepherds. This is true for all the three lining-up methods. However, a 2.5 metres distance may indeed affect the control of the flock (since the shepherds are far between each other). Therefore, a steering points' distance of 2.0 metres is chosen in order to have a better shepherding behaviour.

### 5.3 Simulation Experiments

The proposed approach as described in [algorithm 3.2](#) together with the refinements mentioned in previous sections are applied to the dog and sheep problem and adjusted where necessary. The Player/Stage simulation platform [30] on a Fedora 9 Linux operating system was used to test the refined model. This experiment is to

study the performance of the INT-M cooperative shepherding after applying all of the refinements mentioned earlier, namely the Shepherds' Approach, Shepherds' Formation and Steering Points' Distance. Simulation data were collected and the behaviours of the simulated robots were analysed.

### 5.3.1 Simulation Setup

There are four shepherds using the INT-M model are involved in the simulation, with a sheep flock size from two to four. The range for the robot dogs are set to five metres for forward sight (i.e. laser) and 20 metres for emulating sense of hearing (i.e. communication radius). The field is constructed of a walled field with the size of 40 metres each side. The grazing site is situated at the centre with a radius of five metres and each sheep that have entered it will stop. Each run is limited to a limit of five minutes (i.e. 300 seconds; as used in Figures 5.13 and 5.14) and it is done for six times, then the average values are calculated. An example of the simulation is shown in Figure 5.12 involving four sheep and four robot dogs (but only two of the robot dogs are shown in the figure).

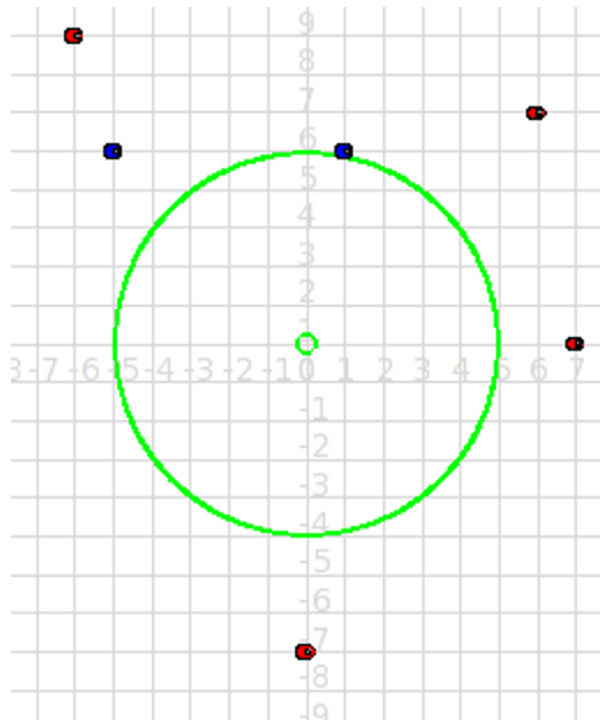


Figure 5.12: An example of the simulation setup with 4 sheep (red) and 2 dogs (blue; another two dogs are not in the current view)

### 5.3.2 Performance Criteria

The performance can mainly be measured on two aspects. The average distance of the flock that is shepherd into the grazing site (which is known as Average Distance to Origin), and also the average percentage of sheep left in the field (which is known as Average Incomplete Tasks) after the maximum time is up. The average percentage of incomplete tasks criterion signifies the ability to maintain the balance of the overall goal of shepherding all the sheep and also completing it within the specified time.

### 5.3.3 Results

Figure 5.13 shows the average distance of the flock (in relation to the origin) over time. There are three flock sizes in the experiment; from two sheep up until four sheep in a herd. The figure indicates that in average the group of sheep is able to be contained within the flock. This reflects on the refinements applied to the dogs' shepherding behaviour. Furthermore, the average distance of flocks with four sheep is quite stable over time. However, flocks of size two do show a relatively smoother transition over time; indicating that the flock is quite manageable.

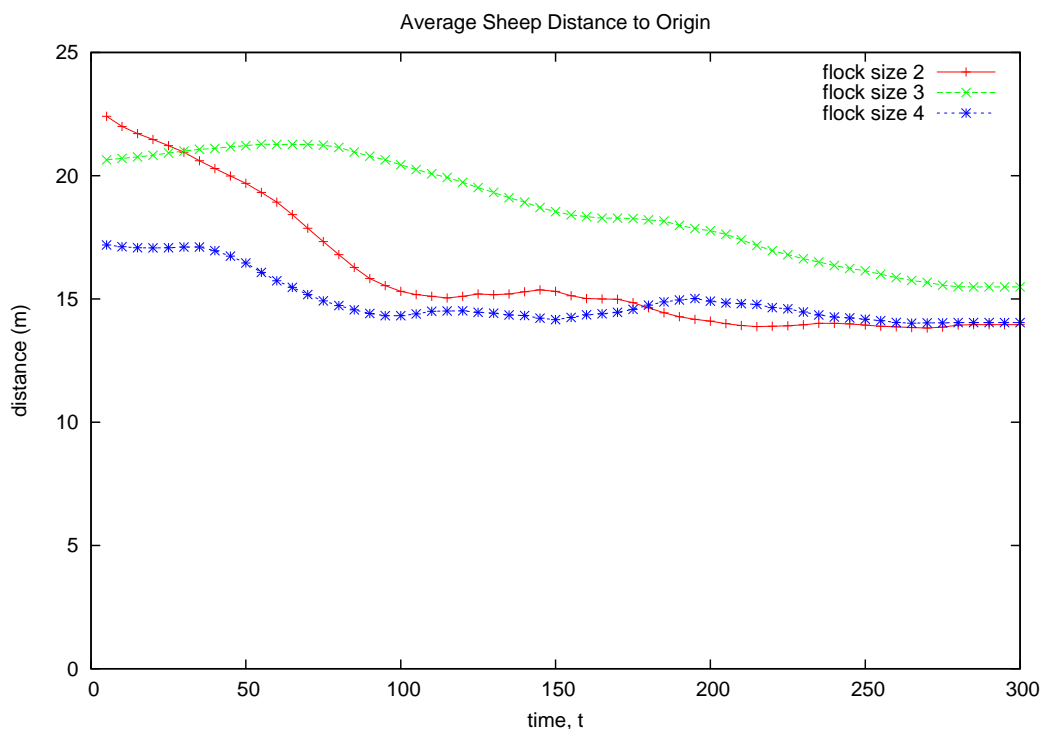


Figure 5.13: Average Distance to Origin

Figure 5.14 shows the average percentage of sheep still outside of the grazing

site over time. The figure suggests that in average there will at least be some sheep that can be shepherd into the grazing site, because after the time is up all of the flock sizes have less than 80% of incomplete tasks remaining. Nonetheless, the average incomplete tasks percentage for all flock sizes are not less than 60%. In general, flocks of size two can achieve lower incomplete task rate within the time limit. On the other hand, flocks with four sheep display quicker response that might indicate a trend.

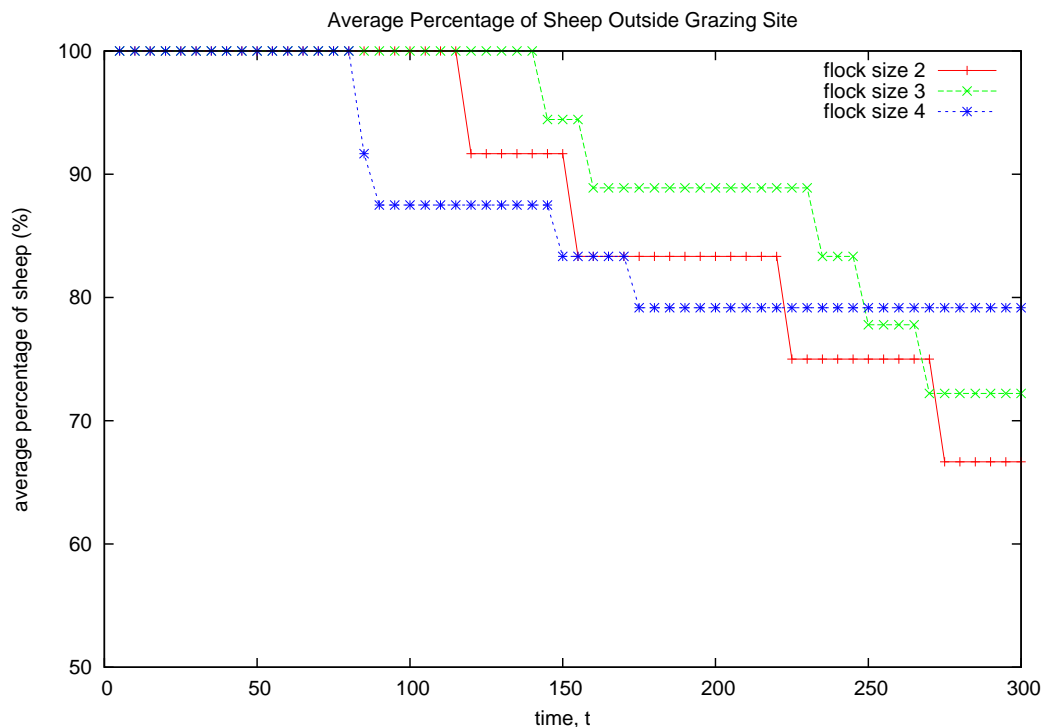


Figure 5.14: Average Incomplete Tasks

This current section discussed the refinements of the INT-M model to include a better low-level shepherding behaviour. Three low-level behaviours that are looked into are: *Shepherds' Approach*, *Shepherds' Formation* and *Steering Points' Distance*. In particular, the formation of multiple shepherds selected is the *Line formation* whereby the robot dogs would line-up behind the group of sheep so that the flock can be better grouped together as mentioned by Lien et al. [53]. The *Vector Projection* method of assigning shepherds to their steering points is chosen. Since the movement towards the (group of) sheep by each robot dogs will influence the flock, the *Safe zone* approach method is selected as the *Shepherds' Approach* refinement. This minimises the inherent problem of flock separation that might happen when a robot dog approaches a group of sheep. The *Steering Points' Distance* has also been studied and the value of 2.0 metres between the shepherds has been chosen. All of these refinements were applied in the simulation experiments in this section. The performance of INT-M with refined cooperative

shepherding behaviour is better compared to the initial performance as discussed in [section 4.4](#).

## 5.4 Flock Identification Refinements

Further refinements are identified based on the experiments done. One of them is described in this section. The shepherd decides on how to move in order to control the movements of the flock. This is known as shepherds locomotion. In order to make the decision, the shepherd needs to identify the flock that he wanted to manage. This task is known as flock identification. The purpose of flock identification is to recognise and determine whether the sheep in the area are in the same flock. It is important because it leads to shepherds locomotion decision.

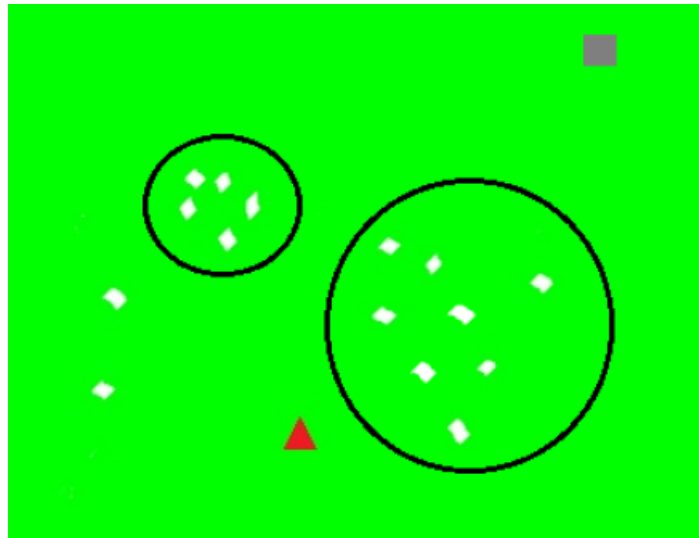


Figure 5.15: Flock identification in a single shepherding scenario

[Figure 5.15](#) shows a scenario of the flock identification phase in single shepherding. The purpose of flock identification is to recognise and determine whether the individuals in the area are in the same flock. The shepherd (shown as a triangle) needs to observe the area and identify which sheep (marked as white) belongs to which flock (shown as black circles). It is important because it leads to shepherds locomotion decision. Once the flock has been identified, then the shepherd can decide which flock needed to be steered first. The flock centre will be calculated in order to determine the steering point and push the flock towards the goal (shown as the grey square near the top-right corner of [Figure 5.15](#)).

This section focuses on investigating on how to adapt the connected-components method in image processing for flock identification in which the idea that each

sheep in the group can be viewed as a pixel in a digital image.

### 5.4.1 Other Approaches

Most of the studies use a bird's eye views in terms of flock identification i.e. seeing all the robot shepherd and robot sheep from the top view such as in studies by Lien et al. [52, 54]. Harrison et al. [31] uses flock blobs which uses occupancy grids which also is based on a bird's eye view of the whole scenario. Lien et al. [52] uses a mathematical model for flock identification called a compact area which is based on the inverse of the packing circles in a circle problem. Meanwhile, Razali et al. [80, 81] uses a different approach whereby the shepherds only have local ground view of the flock, and thus uses a 'perceived flock centre' and the nearest 'flock' member as an anchor to determine the steering points.

#### 5.4.1.1 Flock blobs

Harrison et al. [31] proposed a shepherding strategy, called *DEFORM*. In this algorithm, the flock identification task is done by using flock blobs. Flock blob ( $B_F$ ) is the set of all grid cells occupied by members of the flock. Target blob ( $B_T$ ) is the area to which the shepherds try to guide the flock. Target blob is formed by using 8-connected set around the cell that contains the member of the flock which is closest to the goal ( $f_{closest}$ ). This technique is based on a bird's eye view of the whole scenario.

#### 5.4.1.2 Compact area

Lien et al. [54] uses a mathematical model for flock identification. The shepherds are often not able to keep the flock intact especially for large flocks. Thus, Lien et al. came up with a technique called compact area. The compact area of a group is the smallest circle that could contain all group members. All members outside the compact area are considered as separated, in other words, they are not in the same flock. This technique is based on the inverse version of the packing circles in a circle problem.

### 5.4.1.3 Perceived flock

Razali et al. [80, 81] uses a different approach whereby the shepherds only have local ground view of the flock, and thus uses a ‘perceived flock centre’ and the nearest ‘flock’ member as an anchor to determine the steering points. This is shown in Figure 5.16.

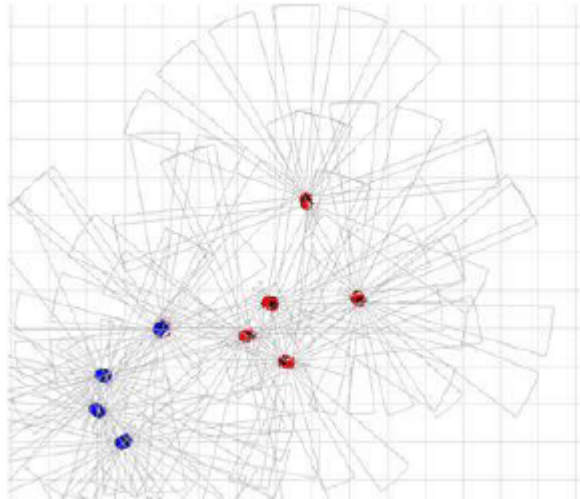


Figure 5.16: The current problem of perceived flock centre; the right-most & top-most sheep are not detected to be in the same flock

## 5.4.2 Motivation

The problem of current approaches is that it is either quite complex or it is not precise enough, such as the ‘perceived flock centre’ approach. This can be exemplified by Figure 5.17. The red blob is the shepherd and will detect all sheep within its radius including the separated blue-coloured sheep and assumes that is the flock. This affects the determination of steering point later. However, by using the connected components labelling method, only the white-coloured sheep is known to be in the same flock, thus attention is given to that identified flock. Therefore, this proposed approach tries to balance between having a simple flock identification technique and obtaining a high degree of accuracy. Although this proposed approach is using the ‘bird’s eye view’, it is limited by the communication radius of the flock members, as discussed in the later sections.

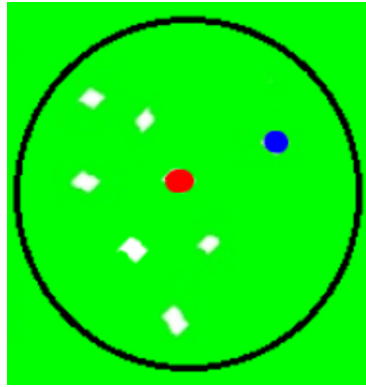


Figure 5.17: The problem of flock identification; the red blob is the shepherd and the white-coloured sheep is an actual flock and the separated blue-coloured sheep should be dismissed, which can be identified using the connected components labelling method proposed.

### 5.4.3 Proposed Method

This section proposes a technique based on an existing method from a different domain. Connected-components methods are well researched in the image processing domain [22]. It is also known as connected components labelling. It is based on graph-theory where the digital image pixels are viewed as vertices and the connected neighbours are the edges. According to Di Stefano and Bulgarelli [21], the definition of connected component relies on that of a pixels neighbourhood. It can be adapted for a more precise identification of flocks by viewing each sheep as a pixel and using the sheep's communication range to find the connected neighbours.

#### 5.4.3.1 Connected Components Labelling Method

In image analysis, specifically in binary images, one of the common problem is to determine which parts of an object is physically connected. Human are gifted with the ability to easily distinguish the differences and notice the similarities, but not computers or robots. The connected components labelling is introduced by Rosenfeld and Pfaltz [87] to solve this problem.

Connected components labelling is defined as a set of pixels that is said to be connected in which each pixel is connected to their neighbouring pixels. A connected components labelling of a binary image,  $B$  is a labelled image  $LB$  in which the value of each pixel is the label of its connected components [93]. An algorithm that takes in a binary image and outputs a new labelled image with distinct labels



for each connected components is called a connected components labelling algorithm [90]. There are two general algorithms for connected components labelling which are recursive algorithm and row-by-row algorithm.

The first one is a straightforward algorithm known as the recursive algorithm. A pixel is chosen from an image and from that pixel, we check its neighbours for connectivity. As the image size grows, the time taken for the algorithm to execute increases rather quickly. This is the disadvantage of the recursive algorithm.

### 5.4.3.2 Classical Connected Components Labelling

The other one is the row-by-row algorithm also known as the classical algorithm [87]. It consists of two passes. During the first pass, the algorithm scans the pixels from left to right, record the equivalences and assign temporary labels. In the second pass, replacement of each temporary label is done by relabelling the label of its equivalent class. Figures 5.18 and 5.19 show the flowchart of first pass and second pass respectively. This classical algorithm uses the union-find data structures which makes this algorithm more efficient [91].

In this section, the classical algorithm of connected components will be used for the task of flock identification. Furthermore, this section only focuses on 8-connectivity neighbourhood definition because 4-connectivity is less precise although it obviously performs faster than the 8-connectivity variant. The connected components algorithm takes place when the shepherd sees a sheep from its current location. It only takes place within the shepherd's vision radius. Once the shepherd has confirmed the number of sheep within its radius of vision, the First Pass is executed.

During the First Pass, the shepherd will perform an 8-connectivity neighbour checking technique using the sheep location as the centre of the 8-connectivity. Whenever a neighbouring sheep is spotted, the current sheep will be assigned to the neighbours label. This phase will continue until all the sheep has been labelled.

After completing the First Pass, the Second Pass will execute. This phase will use the information from the union-find data structures in relabelling the sheep. The shepherd will check each and every sheep in its vision radius. From the sheep's labels, the shepherd could find out whether the label is actually the parent or the child of other labels.

The terms of the classical algorithm in image processing slightly differs from

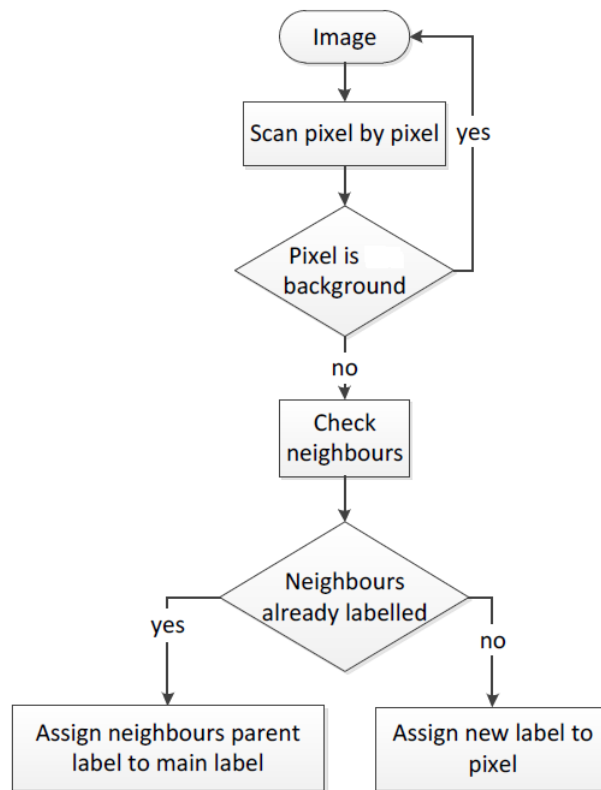


Figure 5.18: First Pass in the Connected Components Algorithm

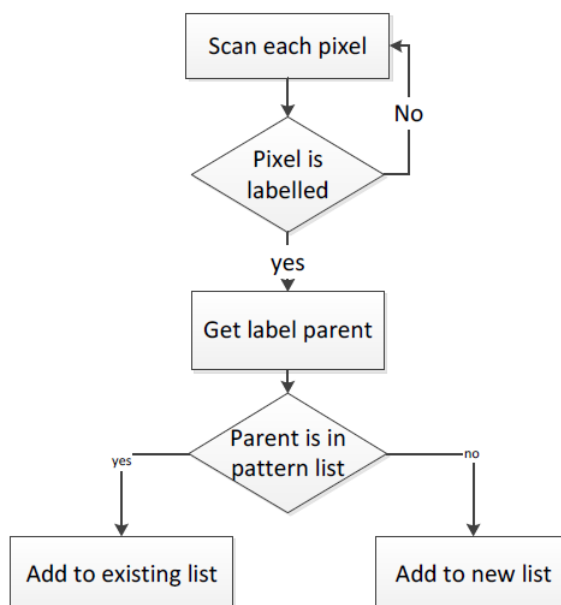


Figure 5.19: Second Pass in the Connected Components Algorithm

the terms of adapted algorithm in multi-robot shepherding but the functionality or the roles of the terms are the same. [Table 5.3](#) shows the comparison of connected components terms between image processing domain and the multi-robot shepherding domain.

Table 5.3: Comparison of terms

Image Processing	Multi-Robot Shepherding
Image	Workspace / Field
Pixels	Sheep
Objects	Flock
Base Pixel	Location of the Shepherd

#### 5.4.4 Performance Measures

In this section, in order to verify the usefulness and the effectiveness of the proposed algorithm, the algorithm needs to undergo a process called performance measurement.

##### 5.4.4.1 Identification

The identification performance test is done to measure the accuracy of the algorithm in the identifying task. The identifying task involves the number of flocks detected by the algorithm and the number of flock members in each flock. The results from the algorithm are compared with the testing data set which is done manually.

The accuracy of the algorithm in flock identification for each run is calculated based on the number of detected flocks. The results were then compared to the actual number of flocks which had been manually identified. The formula for the accuracy of the method in flock identification is shown in [Equation 5.1](#). False positive and false negative values can also be used to know more about flock identification accuracy.

$$flock_{accuracy} = \frac{flock_{detected}}{flock_{actual}} \times 100\% \quad (5.1)$$

Apart from the number of flocks, this test will also involve the flock members in each flock. The accuracy of flock members identification is done by comparing the number of detected flock members by the algorithm and the testing data set. The formula of flock member identification is shown in [Equation 5.2](#).

$$member_{accuracy} = \frac{member_{detected}}{member_{actual}} \times 100\% \quad (5.2)$$

#### 5.4.4.2 Time Taken

The time taken performance test is also done to measure the effectiveness of this method. In this test, it involves the variation of workspace size that the algorithm is working on. The accuracy of the flock and flock members identification is assumed to be 100% since this test focuses on the time taken for the algorithm to complete its task. The time taken for the algorithm to complete its task is calculated as shown in [Equation 5.3](#), where  $n$  scenarios are the number of different sets of sheep positions.

$$time_{workspace} = \frac{time_{total}}{n \text{ scenarios}} \quad (5.3)$$

### 5.4.5 Results

In the first part of the experiment it is done only on C++ to test the connected components algorithm. These experiments are done without integrating with Player/Stage related base codes. The purpose of these tests is to measure the accuracy of the connected components algorithm.

The testing data used for this experiment is obtained by manual identification from the output of the system. The output generated by the system are then compared to the testing data. Each run is measured according to three performance measures which are discussed in [subsection 5.4.4](#). The results from each run for each performance measures are produced and recorded. The results are

then further analysed.

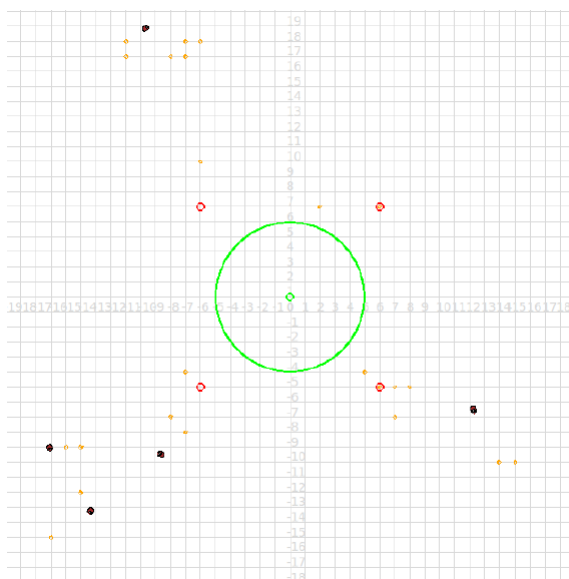


Figure 5.20: Example of Player/Stage simulation run

The latter part of the experiment is performed on the Player/Stage [30, 105] robotics simulation software to test the connected components algorithm together with other robotics behaviours, such as obstacle avoidance, navigation, goal-seeking, and lining-up. A total of 10 runs have been executed on a workspace of size  $40 \times 40$  metres. An example of a Player/Stage simulation run is shown in Figure 5.20.

#### 5.4.5.1 Flock Identification Results: Offline

This experiment involves a total of 40 runs which have been executed with 10 sets of testing data for four different workspace sizes. Figures 5.21, 5.22 and 5.23 show the average accuracy of connected components algorithm in flock identification, flock member identification and average time taken of the algorithm to complete its task respectively.

Based on the results in Figure 5.21, the proposed method has the highest accuracy of flock identification in workspace of size  $20 \times 20$  and has the lowest accuracy of flock identification in workspace of  $6 \times 6$ . There is an increasing pattern from workspace  $6 \times 6$  to  $20 \times 20$  but decreases when it comes to workspace of  $40 \times 40$ . This probably occurred because of the spread of flock members are more dispersed in the workspace of size  $40 \times 40$ .

Based on the results in Figure 5.22, the proposed method has the highest

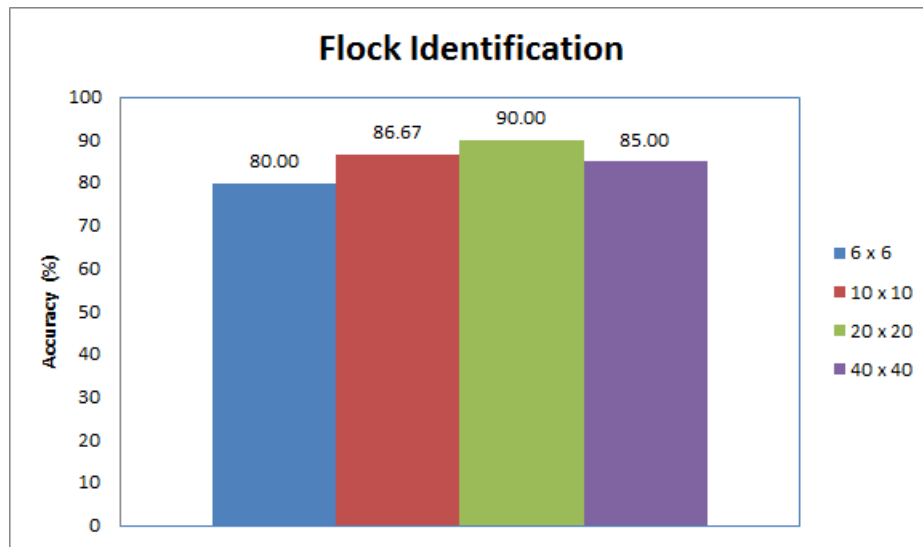


Figure 5.21: Flock Identification accuracy for different workspaces

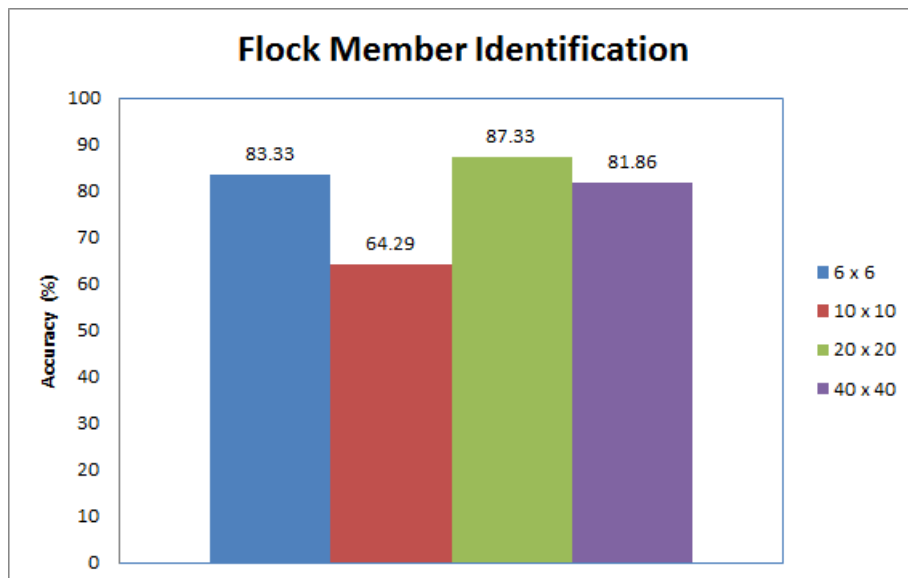


Figure 5.22: Flock Member Identification accuracy for different workspaces

accuracy of flock member identification in workspace  $20 \times 20$  and has the lowest accuracy of flock member identification in the workspace of  $10 \times 10$ . The placement of the flock members are oddly placed which results in irregular pattern.

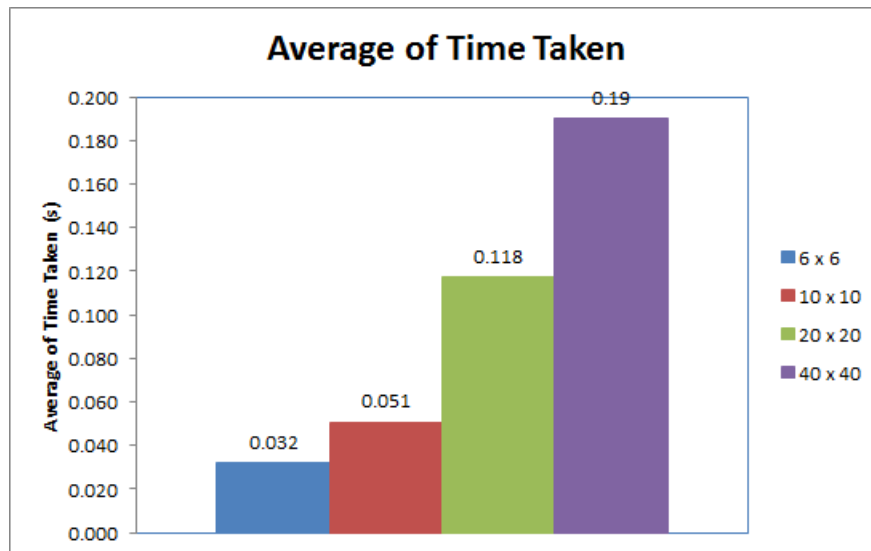


Figure 5.23: Average Time Taken for different workspaces

Based on the results in Figure 5.23, it shows that the proposed algorithm takes the longest time in workspace of  $40 \times 40$  while it takes the shortest time in workspace of  $6 \times 6$ . Obviously, the bigger the workspace size, the more pixels that it needs to process.

#### 5.4.5.2 Flock Identification Results: using Player/Stage

Table 5.4: Comparison between with and without using Player/Stage, with an average of two actual flocks in each 10 different scenarios

	without P/S	with P/S
Flock Identification (%)	85	70.58
Average Correct Flock Detected	2.10	1.00

Based on the results in Table 5.4, it shows that the proposed method without the integration of Player/Stage base code performs better flock identification and managed to detect higher number of correct flocks, on average. This is possibly because of the integrated algorithm involves additional computational works that are needed to be done to perform other navigational behaviours and the line-up of the shepherds. The movements of both the sheep and shepherds might result in the inaccurate identification of flocks.

Table 5.5: Frequencies of flock identification in a  $40 \times 40$  workspace (without using Player/Stage)

<b>Flock</b>	<b>Exist</b>	<b>Does not exist</b>
Detected	22	1
Not detected	4	0
Total	26	
Identification (%)	$\simeq 85$	

Tables 5.5 and 5.6 list the *True Positive* (correctly detecting flocks), *False Positive* (detecting a flock when there is no flock, i.e. separated sheep), *False Negative* (missing a flock) and *True Negative* (correctly not detecting a flock when there is no actual flock) values related to flock identification.

Table 5.6: Frequencies of flock identification in a  $40 \times 40$  workspace (using Player/Stage)

<b>Flock</b>	<b>Exist</b>	<b>Does not exist</b>
Detected	12	1
Not detected	5	2
Total	17	
Identification (%)	70.58	

It is shown that this proposed method has a chance of performing good flock identification in bigger workspaces but the drawbacks of performing in such a workspace is that it consumes more time and more memory. The proposed method seems to be feasible to perform acceptable flock member identification in big workspaces.

In this section, the connected components method has shown its ability to perform flock identification and flock member identification in a satisfyingly high accuracy. The first part of the experiment proved that connected components labelling can be done and is feasible. The second part of the experiment showed that connected components labelling can be used in a multi-robot shepherding scenario by integrating with the Player/Stage robotics simulation platform and its related base codes. The connected components method would perform better compared to the previously used perceived flock centre method in terms of indentifying flocks and subsequently finding the flock centres and optimal steering points. This can be visually seen during the simulation runs of this section.



## 5.5 The INT-X Model: Cooperation with Immune Learning

The other proposed approach is on learning ability of the immunoids. This approach is based on the adaptation of the biological immune systems via the Clonal Selection. The proposed idea is similar to a generalisation of reinforcement learning strategy.

Figure 5.24 shows the *Primary*, *Secondary* and *Cross-Reactive* responses of the immune systems. When an antigen  $Ag_1$  invades the organism, a few specific antibodies are selected to proliferate (i.e. low antibody concentration), but some time is required until a sufficient immune response is mounted against antigen  $Ag_1$ . This required period to reproduce the related antibodies is called *lag phase*, and is longer for the primary response as shown in the Figure 5.24.

Figure is redacted,  
kindly refer [20]  
by de Castro, L. N. and Timmis, J. (2002)

Figure 5.24: Immune Learning after a few immune response have been mounted [20]

In a future or secondary exposure to the same antigen  $Ag_1$ , a faster (i.e. shorter lag phase) and stronger (i.e. higher antibody concentration) response can be mounted thus quickly and effectively killing antigen  $Ag_1$ . Otherwise, if a new antigen  $Ag_2$  is presented, then the response pattern would be similar to that of primary response of  $Ag_1$  in terms of time and antibody concentration.

The immune response is specific in the sense that antibodies successful in recognising a given antigen  $Ag_1$  are specific in recognising that antigen and a similar one,  $Ag_1'$ . Thus, the response of the antibodies initially targeted for antigen

$Ag_1$  to a similar antigen  $Ag_{1'}$  would be similar to a secondary response to  $Ag_1$  which is known as cross-reactive response in the AIS literature [19].

The cross-reactive response is similar to the generalisation capability of neural networks. Cross-reactivity is important for the creation of models of the antigenic universe similar to the importance of generalisation in neural networks for the creation of models of the world. This approach is based on the cross-reactivity of the immune cells in fighting antigen that is similar to the one it was exposed to.

This proposed approach is named as *Immune Network T-cell-regulated—Cross-Reactive* algorithm, or in short the INT-X model (the  $X$  stands for *Cross-Reactive* response). It introduces a stronger response in terms of local group reactions. This reflects in higher concentration,  $s_i(t)$  which influence (i.e. reinforce) more immunoids in the local area to act in the same way (i.e.  $Ab_i$ ).

This is achieved by getting other nearby immunoids' concentration of that specific antibody,  $Ab_i$  via communicating their learnt appropriate action to one another (i.e. reinforce). The approach is added after the memory part as discussed in subsection 3.4.1, as it reinforces the actions that have been stored previously and reinforces the behaviour among the local immunoids.

Algorithm 5.1 shows the algorithm for this approach, which extends algorithm 3.2 in providing a stronger group response by reinforcing the selected appropriate action. The lines 16–18 in algorithm 5.1 is the added part whereby the stimulus and concentration values ( $S_i(t)$  and  $s_i(t)$  respectively) for the specific antibody  $x$ ,  $Ab_x$  (i.e. the previous selected antibody,  $Ab_{max}$ ) is recalculated. However, previous T-cell concentration value,  $c_i(t)$  is maintained since the T-cell model would regulate the antibody concentration to its initial value after successfully executing the previous action.

In order to reinforce the learnt action, a higher concentration of that specific antibody is needed ( $s_i(t)$ ) and this uses the Equation 2.4. That in turn, requires higher stimulus value of that antibody ( $S_i(t)$ ) which uses the Equation 3.1. This requires a higher concentration average of that antibody in nearby immunoids,  $s_j(t)$  in equation Equation 3.1. This higher concentration average shows that the appropriate action learnt,  $Ab_i$  is 'agreed upon' by nearby local immunoids (i.e. achieving local 'consensus'). By reinforcing the learnt action, the immunoid will influence others and this leads to a stronger local group behaviour of that particular action.

This approach can provide some generalisation to the previous INT-M ap-

---

**Algorithm 5.1** Immune Network T-cell-regulated—Cross-Reactive (INT-X)

---

**Require:**  $t = 0$ ,  $S_i(0) = s_i(0) = 0.5$  for  $i = 0 \dots N - 1$ ,  $N$  is number of actions  
**Ensure:** retain previous  $Ab$  if immunoid is not inferior within similar environment, execute  $Ab_{max}$

- 1:  $Ab_{max} \leftarrow Ab_1$  //at start  $Ab_1$  is selected
- 2: immunoid  $\leftarrow$  inferior //at start immunoid is inferior
- 3: environment  $\leftarrow$  similar //at start environment is similar (i.e. static)
- 4: **loop**
- 5:   Execute  $Ab_{max}$
- 6:
- 7:   //immunoid is activated (normal) or excellent
- 8:   **if** immunoid  $\neq$  inferior **then**
- 9:     //environment sensed is similar to previous
- 10:    **if**  $g_i(t) \approx g_i(t - 1)$  **then** //refer Figure 3.3
- 11:      $S_i(t) \leftarrow S_i(t - 1)$  //use previous Stimulus values
- 12:      $s_i(t) \leftarrow s_i(t - 1)$  //use previous Ab concentration values
- 13:      $c_i(t) \leftarrow c_i(t - 1)$  //use previous T-cell concentration values
- 14:
- 15:     //use previous values for all  $i$ , recalculate only for  $x$
- 16:      $x \leftarrow Ab_{max}$  //get the index of the previously selected Ab
- 17:     Calculate  $S_x(t)$  //refer Eq.(3.1)
- 18:     Calculate  $s_x(t)$  //refer Eq.(2.4)
- 19:     //use previous T-cell concentration value
- 20:    **else**
- 21:     environment  $\leftarrow$  changed //need to re-evaluate action
- 22:    **end if**
- 23: **end if**
- 24:
- 25:    //immunoid is inferior or environment has changed
- 26:    **if** (immunoid = inferior) || (environment = changed) **then**
- 27:     //use line 5–21 in Algorithm 3.1
- 28:    **end if**
- 29:
- 30:    **if**  $Ab_i$  has  $max(s_i(t))$  **then** //select Ab with maximum concentration
- 31:      $Ab_{max} \leftarrow Ab_i$
- 32:    **end if**
- 33:
- 34:     $t \leftarrow t + 1$  //each iteration is standard (e.g. 40 unit time)
- 35: **end loop**

---

proach. The INT-M approach provides shorter lag phase for a specific response by retaining previous action while the INT-X approach can add some degree of generalisation to the action-selection phase by communicating (i.e. influencing and reinforcing) with other nearby immunoids. This can be seen in Figure 3.8 (which is only for a single immunoid) if INT-M approach is applied onto the RoboShepherd task which was discussed in subsection 3.5.1. Meanwhile, INT-X had

been discussed in this section that mimics the immune learning feature as shown in Figure 5.24 with the label *Cross-Reactive Response*.

## 5.6 Conclusion

In applying the INT-M model for cooperative shepherding in the earlier chapter, several low-level factors had been identified that affects the group shepherding behaviour. Refinements were made to the cooperative shepherding as discussed in this chapter, namely the Shepherds' Approach, Shepherds' Formation and Steering Points' Distance. The first refinement is just setting a threshold value so that the shepherds do not get too near to the sheep. The other two refinements had been tested and evaluated to get the optimum method and value as discussed in subsections 5.2.2 and 5.2.3, respectively. Furthermore, all these refinements were applied at the same time with the INT-M model and simulations were performed to see whether the model is good for a multi-robot system problem.

Other than that, another factor had been identified later and studied that is the Flock Identification. The refinement that have been proposed is using a method in another domain and applying it in the shepherding problem. However, the INT-X model that had been proposed and discussed in the earlier chapter were not implemented in these latter sections. This is because, from the research done there were various other improvements and modifications that can be further studied in the multi-robot cooperative shepherding problem. Nonetheless, flock identification is important as it affects the cooperative shepherding behaviour (especially the shepherds' locomotion) of the whole multi-robot system. It also affects the task density detected by each shepherd thereby influencing the INT-M model in terms of the stimulation and suppression of actions.

# Chapter 6

## Conclusion

In this research a refined memory-based immune inspired approach for shepherding in multi-robot systems had been studied. I have described the basic concepts of biological immune systems and argued that the immune network is a suitable analogy for multi-robot shepherding problem. The underlying immune inspired cooperative mechanism was described and tested. I have also proposed refinements on the multi-robot cooperation algorithm; the INT-M model, and applied it to the dog-sheep test scenario. Simulation experiments were carried out to evaluate the cooperative mechanism and the whole approach.

### 6.1 Summary

This thesis has laid out the research direction and focus for the study. Immune Systems are described and their applicability to Multi-Robot Systems domain have also been discussed. The description of terminologies and its corresponding robotics use have been stated. The study investigates the Idiotypic Network Hypothesis so that the adapted Immune Network can be used in multi-robot cooperation problems. The immune network is argued to be suitable in achieving desired cooperative behaviour in robots.

The main task scenario that was deeply investigated is the dog-sheep problem. This is because it is generic enough that other domains such as robot patrolling can later be studied. The dog-sheep problem also provides configurable situations such as the number of dogs, sheep, and safety zone. The dog-sheep problem poses a highly dynamic environment for a multi-robot system. In retrospect, the research

objectives set forth in [section 1.3](#) have all been achieved and are restated here in terms of works done.

1. I have proposed two immune-inspired approaches to cooperation.

In relation to the first objective, two models inspired by the immune systems have been proposed in order to solve relevant dynamic cooperative tasks. These proposed models namely the *Immune Network T-cell-regulated—with Memory* (INT-M) model which is discussed in [subsection 3.4.1](#) and *Immune Network T-cell-regulated—Cross-Reactive* (INT-X) model which is described in [section 5.5](#) use the advantages of immune memory and immune learning respectively in order to achieve appropriate local group behaviour. The details of immune inspired models for cooperation have been discussed in [chapter 3](#).

2. I have established an adaptive cooperation algorithm in multi-robot systems.

The ‘Collecting’ task in shepherding behaviour had been studied as the multi-robot system scenario used in this research. Simulation experiments were carried out to see the feasibility of the INT-M model to be used in such a scenario and the results were presented in [section 4.4](#). Furthermore, the underlying ‘cooperation mechanism’ of the INT-M model had been verified in [section 4.5](#) and is shown to be adaptive to dynamic environmental changes. The discussions presented in [chapter 4](#) serves to fulfil the second research objective.

3. I have determined the refinements that can be applied related to cooperation.

The third research objective is regarding the refinements that can be done related to the cooperative behaviour. Three refinements to cooperative shepherding have been investigated in [section 5.2](#), namely Shepherds’ Approach, Shepherds’ Formation and Steering Points’ Distance. These refinements were applied to the INT-M model and simulation experiments were carried out as described in [section 5.3](#). In addition to that, a connected components labelling method for flock identification had been proposed and studied in [section 5.4](#). The details of these refinements were presented in [chapter 5](#).

## 6.2 Main Contributions

This study provides an in-depth understanding of the immune systems and its application in the robotics domain. Below are the main contributions of this research.

1. Two immune-inspired models had been proposed, and one of the model, the INT-M model was implemented and evaluated.
2. The implementation of the cooperative shepherding used in this research is using local ground view; except for the proposed flock identification method which rely on a ‘bird’s eye view’. This sets the study apart from other research, whereby such implementation is indeed difficult but it is more similar to real world situations.
3. The implementation of the immune inspired group behaviour takes into account all the nearby shepherds (i.e. within the communication radius) which is more realistic compared to other works that only uses a one-to-one communication that happens when the shepherds are in contact with one another.
4. The ‘cooperation mechanism’ underlying the immune inspired model (INT-M) had been verified to be adaptive in a dynamic multi-robot scenario.
5. Refinements related to multi-robot cooperative shepherding were identified and tested.
6. This study had recognised the importance of flock identification in relation to cooperative shepherding task and a method to overcome the problem was discussed.
7. The implementation of this study is done on the Player/Stage robotics simulation platform. This means that it can be applied onto real robots with minor changes required.

These contributions have shown that immune inspired multi-robot cooperative shepherding; especially the INT-M model; is feasible and suitable to be used. Several conference papers and articles had been published from this research study, as listed in [Appendix A](#). In the period of the study, several related research activities were performed and achieved as listed in [Appendix B](#).

### 6.3 Suggestions and Future Work

There are other useful features of the biological immune systems that can be further investigated such as the Danger Theory paradigm [1, 2], the B-cell mutation to achieve adaptability during Clonal Selection phase and other interesting processes. Further study can be done to investigate on immune systems based algorithms to be used in multi-robot cooperation. The work so far has enabled a general overview of the area and the feasibility of research in this domain.

The implementation and further study of the INT-X model is highly suggested. It was not implemented in this study since it builds on top of the INT-M algorithm and its refinements. Furthermore, the research is also focused on investigating any refinements to the cooperative shepherding behaviours.

Other than that, other cooperative tasks should also be studied using immune inspired approach, especially the Perimeter Detection and Tracking scenario mentioned in this thesis. Other approaches to multi-robot cooperative shepherding could also be carried out and compared with the INT-M model in this study.

In terms of the shepherding behaviour, this research only studied one task type, which is ‘Collecting’. The study can be extended to the other three task types, namely Herding, Patrolling, and Covering tasks. Other than that, more advanced study can be undertaken by using heterogeneous robots, especially extending this research by integrating it with the ‘Capability Chain’ concept [41, 42].



# References

- [1] Aickelin, U. and Cayzer, S. (2002). The Danger Theory and Its Application to Artificial Immune Systems. In *Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS 2002)*, pages 141–148.
- [2] Aickelin, U. and Greensmith, J. (2007). Sensing Danger: Innate Immunology for Intrusion Detection. *Information Security Technical Report*, 12(2):218–227.
- [3] Arai, T., Pagello, E., and Parker, L. E. (2002). Guest editorial advances in multi-robot systems. *Robotics and Automation, IEEE Transactions on*, 18(5):655–661.
- [4] Bahceci, E., Soysal, O., and Sahin, E. (2003). A Review: Pattern Formation and Adaptation in Multi-Robot Systems. Technical Report CMU-RI-TR-03-43, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA.
- [5] Balch, T. (1997). Social Entropy: A New Metric for Learning Multi-robot Teams. In *In Proceedings 10th International FLAIRS Conference (FLAIRS-97)*, pages 272–277. Georgia Institute of Technology.
- [6] Behringer, R., Sundareswaran, S., Gregory, B., Elsley, R., Addison, B., Guthmiller, W., Daily, R., Bevely, D., and and, R. S. (2004). The DARPA grand challenge-development of an autonomous vehicle. In *Intelligent Vehicles Symposium, 2004 IEEE*, pages 226–231.
- [7] Bennett, B. and Trafankowski, M. (2012). A Comparative Investigation of Herding Algorithms. In *Proceedings of the 2012 AISB/IACAP World Congress*.
- [8] Bhattacharjee, P., Rakshit, P., Goswami, I., Konar, A., and Nagar, A. K. (2011). Multi-Robot Path Planning using Artificial Bee Colony Optimization Algorithm. In *Proceedings of the 3rd World Congress on Nature and Biologically Inspired Computing (NaBIC 2011)*, pages 219–224.
- [9] Bonabeau, E., Dorigo, M., and Theraulaz, G. (1999). *Swarm Intelligence:*

- from Natural to Artificial Systems*, volume 4. Oxford University Press, New York.
- [10] Bradley, D. and Tyrell, A. (2000). Immunotronics: Hardware fault tolerance inspired by the immune system. *Lecture Notes in Computer Science*, 1801:11–20.
- [11] Bradley, D. and Tyrrell, A. (2002). Immunotronics - novel finite-state-machine architectures with built-in self-test using self-nonsel self differentiation. *Evolutionary Computation, IEEE Transactions on*, 6(3):227–238.
- [12] Brownlee, J. (2007). OAT: The Optimization Algorithm Toolkit. Technical Report 20071220A, Swinburne University of Technology, Victoria, Australia.
- [13] Cao, U. Y., Fukunaga, A. S., and Kahng, A. B. (1997). Cooperative Mobile Robotics: Antecedents and Directions. *Autonomous Robots*, 4(1):7–23.
- [14] Chaimowicz, L. and Kumar, V. (2004). Aerial Shepherds: Coordination among UAVs and Swarms of Robots. In *7th International Symposium on Distributed Autonomous Robotic Systems*, pages 243–252. Springer.
- [15] Clark, J. and Fierro, R. (2005). Cooperative hybrid control of robotic sensors for perimeter detection and tracking. In *American Control Conference, 2005. Proceedings of the 2005*, pages 3500–3505.
- [16] Clark, J. and Fierro, R. (2007). Mobile robotic sensors for perimeter detection and tracking. *ISA Transactions*, 46(1):3–13.
- [17] DARPA. DARPA Robotics Challenge. accessed Sept 29, 2013. <http://www.theroboticschallenge.org>.
- [18] DARPA. DARPA Urban Challenge. accessed Sept 29, 2013. <http://archive.darpa.mil/grandchallenge/>.
- [19] de Castro, L. N. (2006). *Fundamentals of Natural Computing: Basic Concepts, Algorithms, and Applications*. Chapman & Hall/CRC.
- [20] de Castro, L. N. and Timmis, J. (2002). *Artificial Immune Systems: A New Computational Intelligence Approach*. Springer.
- [21] Di Stefano, L. and Bulgarelli, A. (1999). A simple and efficient connected components labeling algorithm. In *Image Analysis and Processing, 1999. Proceedings. International Conference on*, pages 322–327. IEEE.

- [22] Dillencourt, M. B., Samet, H., and Tamminen, M. (1992). A general approach to connected-component labeling for arbitrary image representations. *Journal of the ACM (JACM)*, 39(2):253–280.
- [23] Dudek, G., Jenkin, M., Milios, E., and Wilkes, D. (1996). A taxonomy for multi-agent robotics. *Autonomous Robots*, 3(4):375–397.
- [24] Farinelli, A., Iocchi, L., and Nardi, D. (2004). Multirobot systems: a classification focused on coordination. *Systems, Man, and Cybernetics, Part B, IEEE Transactions on*, 34(5):2015–2028.
- [25] Farmer, J., Kauffman, S., Packard, N., and Perelson, A. (1987). Adaptive Dynamic Networks as Models for the Immune System and Autocatalytic Sets. *Annals of the New York Academy of Sciences*, 504(1 Perspectives in Biological Dynamics and Theoretical Medicine):118–131.
- [26] Farmer, J., Packard, N., and Perelson, A. (1986). The immune system, adaptation, and machine learning. *Physica*, 22(2):187–204.
- [27] Ferber, J. (1999). *Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence*. Addison-Wesley Professional.
- [28] Gao, Y. and Wei, W. (2006). Multi-Robot Autonomous Cooperation Integrated with Immune Based Dynamic Task Allocation. In *Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA'06)*.
- [29] Gerkey, B. P. and Mataric, M. J. (2004). A Formal Analysis and Taxonomy of Task Allocation in Multi-Robot Systems. *The International Journal of Robotics Research*, 23(9):939–954.
- [30] Gerkey, B. P., Vaughan, R. T., and Howard, A. (2003). The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems. In *In Proceedings of the 11th International Conference on Advanced Robotics*, pages 317–323.
- [31] Harrison, J. F., Vo, C., and Lien, J.-M. (2010). Scalable and Robust Shepherding via Deformable Shapes. Technical report, Department of Computer Science, George Mason University, USA.
- [32] Hart, E., Ross, P., Webb, A., and Lawson, A. (2003). A Role for Immunology in “Next Generation Robot Controllers. In *Artificial Immune Systems, Proceedings of the Second International Conference on (ICARIS 2003)*, Lecture Notes in Computer Science, pages 46–56.

- [33] Ishiguro, A., Kondo, T., Watanabe, Y., and Uchikawa, Y. (1995). Dynamic Behavior Arbitration of Autonomous Mobile Robots Using Immune Networks. In *Proceedings of the 1995 IEEE International Conference on Evolutionary Computation*.
- [34] Ishiguro, A., Kondo, T., Watanabe, Y., and Uchikawa, Y. (1996a). Immunoid: An immunological approach to decentralized behavior arbitration of autonomous mobile robots. *Lecture Notes in Computer Science*, 1141:666–675.
- [35] Ishiguro, A., Shirai, Y., Kondo, T., and Uchikawa, Y. (1996b). Immunoid: An Architecture for Behavior Arbitration Based on the Immune Networks. In *Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 1730–1738.
- [36] Ishiguro, A., Watanabe, Y., Kondo, T., Shirai, Y., and Uchikawa, Y. (1996c). Immunoid: A Robot with a Decentralized Behavior Arbitration Mechanisms Based on the Immune System. In *ICMAS Workshop on Immunity-Based Systems*.
- [37] Ishiguro, A., Watanabe, Y., Kondo, T., and Uchikawa, Y. (1996d). Decentralized Consensus-making Mechanisms Based on Immune System: Application to a Behavior Arbitration of an Autonomous Mobile Robot. In *Evolutionary Computation, 1996., Proceedings of IEEE International Conference on*, pages 82–87.
- [38] Ismail, A. R. (2011). *Immune-inspired Self-healing swarm robotic systems*. PhD thesis, University of York.
- [39] Jerne, N. K. (1984). Idiotypic Networks and Other Preconceived Ideas. *Immunological Reviews*, 79(1):5–24.
- [40] Kalra, N., Zlot, R., Dias, M., and Stentz, A. (2005). Market-based multirobot coordination: A comprehensive survey and analysis. Technical Report CMU-RI-TR-05-16, Carnegie Mellon University.
- [41] Khan, M. T. (2010). *Robust and Autonomous Multi-Robot Cooperation using an Artificial Immune System*. PhD thesis, University of British Columbia.
- [42] Khan, M. T. and de Silva, C. W. (2008). Autonomous Fault Tolerant Multi-Robot Cooperation Using Artificial Immune System. In *Proceedings of the IEEE International Conference on Automation and Logistics (ICAL 2008)*, pages 623–628.

- [43] Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., and Osawai, E. (1997). RoboCup: The Robot World Cup Initiative. In *Proceedings of the First International Conference on Autonomous Agents*, AGENTS '97, pages 340–347, New York, NY, USA. ACM.
- [44] Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawai, E., and Matsubara, H. (1998). RoboCup: A challenge problem for AI and robotics. In *RoboCup-97: Robot Soccer World Cup I*, pages 1–19. Springer Berlin Heidelberg.
- [45] Kitano, H., Tadokoro, S., Noda, I., Matsubara, H., Takahashi, T., Shinjou, A., and Shimada, S. (1999). RoboCup Rescue: search and rescue in large-scale disasters as a domain for autonomous agents research. In *1999 IEEE International Conference on Systems, Man, and Cybernetics, 1999. IEEE SMC'99 Conference Proceedings*, volume 6, pages 739–743.
- [46] Koenig, N. (2004). Stage and Gazebo: The Instant Expert's Guide. Nate Koenig, USC Robotics Research Lab. accessed Sept 29, 2013. [http://robotics.usc.edu/~nkoenig/stage\\_gazebo\\_tutorial.ppt](http://robotics.usc.edu/~nkoenig/stage_gazebo_tutorial.ppt).
- [47] Lacroix, S., Alami, R., Lemaire, T., Hattenberger, G., and Gancet, J. (2007). Decision Making in Multi-UAVs Systems: Architecture and Algorithms. In *Multiple Heterogeneous Unmanned Aerial Vehicles*, Springer Tracts in Advanced Robotics, pages 15–48. Springer Berlin Heidelberg.
- [48] Lee, D.-W. and Sim, K.-B. (1997). Artificial immune network-based cooperative control in collective autonomous mobile robots. In *Robot and Human Communication, RO-MAN'97. Proceedings of the 6th IEEE International Workshop on*, pages 58–63.
- [49] Lemaire, T., Alami, R., and Lacroix, S. (2004). A Distributed Task Allocation Scheme in Multi-UAV Context. In *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA 2004)*, volume 4, pages 3622–3627.
- [50] Li, J., Xu, H., Wang, S., and Bai, L. (2007). An Immunology-based Cooperation Approach for Autonomous Robots. *ISKE-2007 Proceedings*, 4.
- [51] Lien, J., Bayazit, O., Sowell, R., Rodriguez, S., and Amato, N. (2004a). Shepherd behaviors. *Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on*, 4:4159–4164.
- [52] Lien, J., Bayazit, O., Sowell, R., Rodriguez, S., and Amato, N. (2004b). Shepherd Behaviors. Technical report, Parasol Lab, Texas A& M.

- [53] Lien, J., Rodriguez, S., Malric, J., and Amato, N. (2005a). Shepherding behaviors with multiple shepherds. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005. ICRA 2005*, pages 3402–3407.
- [54] Lien, J., Rodriguez, S., Malric, J., and Amato, N. (2005b). Shepherding Behaviors with Multiple Shepherds. Technical report, Parasol Lab, Texas A&M.
- [55] Lima and Custódio (2005). Multi-Robot Systems. *Innovations in Robot Mobility and Control*, pages 1–64.
- [56] Liu, S., Mao, L., and Yu, J. (2006). Path Planning based on Ant Colony Algorithm and Distributed Local Navigation for Multi-Robot Systems. In *Proceedings of the 2006 IEEE International Conference on Mechatronics and Automation (ICMA 2006)*, pages 1733–1738.
- [57] Lu, G. (2012). Multi-Robot Cooperation: Shepherd Formation. Master’s thesis, Department of Computer Science, Loughborough University.
- [58] Luh, G.-C. and Liu, W.-W. (2007). Motion planning for mobile robots in dynamic environments using a potential field immune network. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 221(7):1033–1046.
- [59] Luh, G.-C. and Liu, W.-W. (2008a). *An Immunological Approach to Mobile Robot Navigation*, chapter 15. InTech.
- [60] Luh, G.-C. and Liu, W.-W. (2008b). An immunological approach to mobile robot reactive navigation. *Applied Soft Computing*, 8(1):30–45.
- [61] Luh, G.-C., Wu, C.-Y., and Liu, W.-W. (2006). Artificial Immune System based Cooperative Strategies for Robot Soccer Competition. In *Proceedings of The 1st International Forum on Strategic Technology*, pages 76–79.
- [62] Mataric, M. J. (1994). *Interaction and intelligent behavior*. PhD thesis, Department of Electrical Engineering and Computer Science.
- [63] Mataric, M. J. (1995). Issues and approaches in the design of collective autonomous agents. *Robotics and Autonomous Systems*, 16:321–331.
- [64] Miki, T., Nagao, M., Kobayashi, H., and Nakamura, T. (2010). A Simple Rule Based Multi-Agent Control Algorithm and Its Implementation using Autonomous Mobile Robots. *World Automation Congress*.

- [65] Miki, T. and Nakamura, T. (2006). An Effective Simple Shepherding Algorithm Suitable for Implementation to a Multi-mobile Robot System. In *Proceedings of the First International Conference on Innovative Computing, Information and Control (ICICIC'06)*.
- [66] Mondada, F., Gambardella, L. M., Floreano, D., Nolfi, S., Deneuborg, J.-L., and Dorigo, M. (2005). The Cooperation of Swarm-bots: Physical Interactions in Collective Robotics. *Robotics & Automation Magazine, IEEE*, 12(2):21–28.
- [67] Nagao, M. and Miki, T. (2010). Cooperative behavior generation method using local communication for distributed multi-agent systems. In *Proceedings of the 2010 Systems, Man and Cybernetics*.
- [68] Neshat, M., Sepidnam, G., Sargolzaei, M., and Toosi, A. N. (2012). Artificial Fish Swarm Algorithm: A Survey of the state-of-the-art, hybridization, combinatorial and indicative applications. *Artificial Intelligence Review*.
- [69] Noreils, F. R. (1993). Toward a Robot Architecture Integrating Cooperation between Mobile Robots: Application to Indoor Environment. *The International Journal of Robotics Research*, 12(1):79–98.
- [70] Obst, O. and Rollmann, M. (2005). Spark - A generic simulator for physical multi-agent simulations. *Computer Systems Science and Engineering*, 20(5):347.
- [71] Ollero, A., Lacroix, S., Merino, L., Gancet, J., Wiklund, J., Remuss, V., Perez, I., Gutierrez, L., Viegas, D., Benitez, M., Mallet, A., Alami, R., Chatila, R., Hommel, G., Lechuga, F., Arrue, B., Ferruz, J., Martinez-De Dios, J., and Caballero, F. (2005). Multiple eyes in the skies: architecture and perception issues in the COMETS unmanned air vehicles project. *Robotics & Automation Magazine, IEEE*, 12(2):46–57.
- [72] Parker, L. (1998). ALLIANCE: an architecture for fault tolerant multirobot cooperation. *Robotics and Automation, IEEE Transactions on*, 14(2):220–240.
- [73] Parker, L. (2008). Distributed Intelligence: Overview of the Field and its Application in Multi-robot Systems. *Journal of Physical Agents*, 2(1).
- [74] Potter, M. A., Meeden, L., and Schultz, A. C. (2001). Heterogeneity in the coevolved behaviors of mobile robots: The emergence of specialists. In *Proceedings of the 2001 International Joint Conference of Artificial Intelligence (IJCAI 2001)*, pages 1337–1343.

- [75] Prencipe, G. (2000). A New Distributed Model to Control and Coordinate a Set of Autonomous Mobile Robots: The CORDA Model. Technical report, Universita di Pisa.
- [76] Raza, A. and Fernandez, B. (2010). Immuno-inspired Heterogeneous Mobile Robotic Systems. In *49th IEEE Conference on Decision and Control (CDC 2010)*, pages 7178–7183.
- [77] Raza, A. and Fernandez, B. (2012). Immuno-inspired Robotic Applications: A Review. *CoRR*, abs/1202.4261.
- [78] Razali, S., Meng, Q., and Yang, S.-H. (2009a). Memory-based Immune Network for Multi-Robot Cooperation. In *Proceedings of the 7th International Simulation Conference (ISC 2009)*, pages 91–96.
- [79] Razali, S., Meng, Q., and Yang, S.-H. (2009b). Multi-robot cooperation inspired by immune systems. *AISB Quarterly*, (128):3–5.
- [80] Razali, S., Meng, Q., and Yang, S.-H. (2009c). Multi-robot cooperation using immune network with memory. In *7th IEEE International Conference on Control and Automation (ICCA 2009)*, pages 145–150.
- [81] Razali, S., Meng, Q., and Yang, S.-H. (2010a). A Refined Immune Systems Inspired Model for Multi-Robot Shepherding. In *Nature and Biologically Inspired Computing (NaBIC), 2010 Second World Congress on*, pages 473–478, Kitakyushu, Japan.
- [82] Razali, S., Meng, Q., and Yang, S.-H. (2010b). Shepherding: An Immune-Inspired Robotics Approach. In *The First UK-Malaysia-Ireland Engineering & Science Conference (UMIES 2010)*, page 31. Published (Abstract).
- [83] Razali, S., Meng, Q., and Yang, S.-H. (2012). Immune-Inspired Cooperative Mechanism with Refined Low-level Behaviors for Multi-Robot Shepherding. *International Journal of Computational Intelligence and Applications (IJCIA)*, 11(1):1250007–1250022.
- [84] Razali, S., Shamsudin, N. F., Osman, M., Meng, Q., and Yang, S.-H. (2013). Flock Identification using Connected Component Labeling for Multi-Robot Shepherding. In *Proceedings of the 5th International Conference of Soft Computing & Pattern Recognition (SoCPaR 2013)*, pages 299–304.
- [85] Reynolds, C. W. (1987). Flocks, Herds, and Schools: A Distributed Behavioral Model. In *Proceedings of the ACM SIGGRAPH Computer Graphics Conference*, pages 25–34.



- [86] Reynolds, C. W. (1999). Steering Behaviors For Autonomous Characters. In *Proceedings of the Game Developers Conference*, pages 763–782.
- [87] Rosenfeld, A. and Pfaltz, J. L. (1966). Sequential operations in digital picture processing. *Journal of the ACM (JACM)*, 13(4):471–494.
- [88] Schultz, A., Grefenstette, J., and Adams, W. (1996). Roboshepherd: Learning a complex behavior. *Robotics and Manufacturing: Recent Trends in Research and Applications*, 6:763–768.
- [89] Shannon, C. E. (2001). A Mathematical Theory of Communication. *SIGMOBILE Mobile Computing and Communications Review*, 5(1):3–55.
- [90] Shapiro, L. (2000). Recursive Connected Components Algorithm for Labeling Binary Images. CSE 326: Data Structures, Online Notes. accessed Mar 1, 2013. <http://courses.cs.washington.edu/courses/cse326/00sp/assignments/prog1.html>.
- [91] Shapiro, L. G. and Stockman, G. C. (2001). *Computer Vision*. Prentice-Hall, New Jersey.
- [92] Shehory, O., Sycara, K., and Jha, S. (1998). Multi-agent coordination through coalition formation. In *Proceedings of the 4th International Workshop on Intelligent Agents IV, Agent Theories, Architectures, and Languages*, pages 143–154. Springer-Verlag, London, UK.
- [93] Sinha, U. (2010). Connected Component Labelling. AI Shack website. accessed Feb 25, 2013. <http://www.aishack.in/2010/03/connected-component-labelling/>.
- [94] Srividhya, S. and Ferat, S. (2002). AISIMAM-An Artificial Immune System based Intelligent MultiAgent Model and its application to a mine detection problem. In *Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS-2002)*, pages 22–31.
- [95] Sun, S.-J., Lee, D.-W., and Sim, K.-B. (2001). Artificial immune-based swarm behaviors of distributed autonomous robotic systems. *Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on*, 4:3993–3998.
- [96] Tadokoro, S., Kitano, H., Takahashi, T., Noda, I., Matsubara, H., Shinjoh, A., Koto, T., Takeuchi, I., Takahashi, H., Matsuno, F., Hatayama, M., Nobe, J., and Shimada, S. (2000). The RoboCup-Rescue project: a robotic approach to the disaster mitigation problem. In *IEEE International Conference on Robotics and Automation, 2000. Proceedings. ICRA '00*, volume 4, pages 4089–4094.

- [97] Tambe, M., Adibi, J., Al-Onaizan, Y., Erdem, A., Kaminka, G. A., Marsella, S. C., and Muslea, I. (1999). Building agent teams using an explicit teamwork model and learning. *Artificial Intelligence*, 110(2):215–239.
- [98] Tarakanov, A. and Nicosia, G. (2007). Foundations of Immunocomputing. In *Foundations of Computational Intelligence, 2007. FOCI 2007. IEEE Symposium on*, pages 503–508.
- [99] Tarakanov, A., Skormin, V., and Sokolova, S. (2003). *Immunocomputing: Principles and Applications*. Springer.
- [100] Technology, B. N. (2013). Google’s Schaft robot wins DARPA rescue challenge. accessed Sept 29, 2013. <http://www.bbc.co.uk/news/technology-25493584>.
- [101] Tews, A. and Wyeth, G. (2000). MAPS: a system for multi-agent coordination. *Advanced Robotics*, pages 37–50.
- [102] Thrun, S. (2006). Winning the DARPA Grand Challenge. In *Knowledge Discovery in Databases (PKDD 2006)*, pages 4–4. Springer.
- [103] Thrun, S., Montemerlo, M., Dahlkamp, H., Stavens, D., Aron, A., et al. (2006). Stanley: The robot that won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9):661–692.
- [104] Vail, D. and Veloso, M. (2003). Dynamic multi-robot coordination. In *In Multi-Robot Systems: From Swarms to Intelligent Automata, Volume II*, pages 87–100.
- [105] Vaughan, R. (2008). Massively multi-robot simulation in Stage. *Swarm Intelligence*, 2(2-4):189–208.
- [106] Wang, Z., Tianfield, H., and Jiang, P. (2003). A framework for coordination in multi-robot systems. In *Industrial Informatics, 2003. INDIN 2003. Proceedings. IEEE International Conference on*, pages 483–489.
- [107] Whitbrook, A. M., Aickelin, U., and Garibaldi, J. M. (2007). Idiotypic Immune Networks in Mobile-robot Control. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 37(6):1581–1598.
- [108] Whitbrook, A. M., Aickelin, U., and Garibaldi, J. M. (2010). Two-timescale learning using idiotypic behaviour mediation for a navigating mobile robot. *Applied Soft Computing*, 10(3):876–887.

- [109] Wooldridge, M. (2002). *Introduction to MultiAgent Systems*. John Wiley & Sons.
- [110] Yingying, D., Yan, H., and Jingping, J. (2003). Multi-Robot Cooperation Method based on the Ant Algorithm. In *Proceedings of the 2003 IEEE Swarm Intelligence Symposium (SIS 2003)*, pages 14–18.
- [111] Zheng, T. and Li, J. (2010). Multi-Robot Task Allocation and Scheduling based on Fish Swarm Algorithm. In *Proceedings of the 8th World Congress on Intelligent Control and Automation (WCICA 2010)*, pages 6681–6685.

# Appendix A

## List of Publications

During the course of this study, the following original contributions were made.

Table A.1: List of publications in refereed academic journals

Details	Publisher	Status
<b>IJCIA 2012</b>	World Scientific	Published [83]
International Journal of Computational Intelligence & Applications, 11(1) <i>“Immune-inspired Cooperative Mechanism with Refined Low-level Behaviors for Multi-Robot Shepherding”</i>		

Table A.2: List of publications in scientific community periodicals

Details	Publisher	Status
<b>AISB Magazine 2009</b>	AISB, UK	Published [79]
AISB Quarterly Magazine, February 2009, No. 128 <i>“Multi-Robot Cooperation Inspired by Immune Systems”</i>		

Table A.3: List of publications in refereed conference proceedings

Details	Venue	Status
<b>ISC 2009</b>	Loughborough, UK	Published [78]
International Simulation Conference 2009 <i>“Memory-based Immune Network for Multi-Robot Cooperation”</i>		
<b>IEEE-ICCA 2009</b>	Christchurch, New Zealand	Published [80]
IEEE International Conference on Control & Automation 2009 <i>“Multi-Robot Cooperation using Immune Network with Memory”</i>		
<b>UMIES 2010</b>	Belfast, UK	Published (Abstract) [82]
UK-Malaysia-Ireland Engineering & Science Conference 2010 <i>“Shepherding: An Immune-Inspired Robotics Approach”</i>		
<b>NaBIC 2010</b>	Kitakyushu, Japan	Published [81]
World Congress on Nature & Biologically Inspired Computing 2010 <i>“A Refined Immune Systems Inspired Model for Multi-Robot Shepherding”</i>		
<b>SoCPaR 2013</b>	Hanoi, Vietnam	Published [84]
International Conference of Soft Computing & Pattern Recognition 2013 <i>“Flock Identification using Connected Components Labeling for Multi-Robot Shepherding”</i>		

# Appendix B

## List of Activities

Table B.1 shows a list of all the related research activities achieved.

Table B.1: List of related activities achieved

No.	Activity	Status
1.	BCS-SGAI Forum, Cambridge University	Presented
2.	BCS-SGAI NCAF Forum, Aston University	Accepted
3.	Research Group Seminar, FK Meeting Room	Presented
4.	PGR Poster Competition, Loughborough University	Participated
5.	Poster Competition, Research School of Informatics	Participated
6.	Poster Session, Bundy Symposium	Presented
7.	TAROS 2008 Paper Submission, Edinburgh	Accepted
8.	Article submission, AISB Quarterly Magazine	Published
9.	Funding Award for Bundy Symposium, AISB	Received
10.	Progress Meeting 1, Director of Research	Completed
11.	Partnership Proposals submitted to three companies	Completed
12.	Faculty Grant Application (teaching & research)	Successful
13.	21 Professional Development courses & workshops	Attended
14.	ISC 2009 Paper Submission, Loughborough	Published
15.	IEEE-ICCA 2009 Paper Submission, New Zealand	Published
16.	Simulation Code-base	Completed
17.	Transfer Viva, Department of Computer Science	Completed
18.	UKCI 2009, University of Nottingham	Attended
19.	Research Student Seminar, CS Department	Presented
20.	Poster Session, UMIES 2010, Queen's University Belfast	Presented
21.	Oral Session, UMIES 2010, Queen's University Belfast	Presented
22.	Virtual Poster Competition, RSI	Participated
23.	NaBIC 2010 Paper Submission, Japan	Published
24.	Article submission, IJCIA 11(1), 2012	Published
25.	SoCPaR 2013 Paper Submission, Vietnam	Published

**Notes:**

- No. 2: Accepted, was unable to present
- No. 7: Accepted for Poster session, decided not to proceed
- No. 9: The award covers travel, accommodation, poster printing & one-year student membership
- No. 11: Wany Robotics (France)<sup>1</sup>, Merlin Systems (United Kingdom) & Videre Design (United States)
- No. 12: The grant applied is for e-puck robots from GCtronics (Swiss)
- No. 14: Accepted as an Extended Paper
- No. 21: Abstract published

---

<sup>1</sup>Listed as 5 finalists: <http://www.wanyrobotics.com/academic-partnership-program.html>