2006-12

Investigating ground swarm robotics using agent based simulation

Ho, Sze-Tek Terence

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/2449

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

INVESTIGATING GROUND SWARM ROBOTICS USING AGENT BASED SIMULATION
by
Sze-Tek Terence Ho
December 2006
Thesis Advisor: Susan M. Sanchez
Second Reader: Chwee Seng Choo

Approved for public release; distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** December 2006	**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE: Investigating Ground Swarm Robotics Using Agent Based Simulation			**5. FUNDING NUMBERS**
6. AUTHOR(S) Sze-Tek Terence Ho			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited			**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The concept of employing ground swarm robotics to accomplish tasks has been proposed for future use in humanitarian de-mining, plume monitoring, searching for survivors in a disaster site, and other hazardous activities. More importantly in the military context, with the development of advanced explosive detectors, swarm robotics with autonomous search and detection capability could potentially address the improvised explosive device (IED) problem faced by foot patrols, and aid in the search for hidden ammunition caches and weapons of mass destruction (WMDs). The intent of this research is to leverage on agent based simulation to model a ground robotic swarm on a search and detection mission in a semi-urban environment rigged with stationary IEDs. Efficient design of experiment (DOE) techniques and data farming are engaged to help identify controllable factors and capabilities that have the most impact on overall effectiveness. The focus of this thesis is to explore agent based simulation applied to swarm robotics; the technological and algorithmic aspects are not delved on. Results from the simulations provide several insights on the impact of both decision and noise factors on the performance of the swarm. Incorporation of virtual pheromones as a shared memory map is modeled as an additional capability that is found to enhance the robustness and reliability of the swarm.

14. SUBJECT TERMS Agent Based Modeling (ABM), Agent Based Simulation, Modeling and Simulation, Design of Experiment (DOE), Swarm Robotics, Map Aware Non-uniform Automata (MANA), Adaptive Robots, Nearly Orthogonal Latin Hypercube (NOLH)			**15. NUMBER OF PAGES** 141
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

INVESTIGATING GROUND SWARM ROBOTICS USING AGENT BASED SIMULATION

Sze-Tek Terence Ho
Captain, Singapore Army
B.Sc. (Mechanical Engineering), Cornell University, 2002
M.Eng (Mechanical Engineering), Cornell University, 2003

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
December 2006

Author: Sze-Tek Terence Ho

Approved by: Susan M. Sanchez
Thesis Advisor

Chwee Seng Choo
Second Reader

James N. Eagle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The concept of employing ground swarm robotics to accomplish tasks has been proposed for future use in humanitarian de-mining, plume monitoring, searching for survivors in a disaster site, and other hazardous activities. More importantly in the military context, with the development of advanced explosive detectors, swarm robotics with autonomous search and detection capability could potentially address the improvised explosive device (IED) problem faced by foot patrols, and aid in the search for hidden ammunition caches and weapons of mass destruction (WMDs). The intent of this research is to leverage on agent based simulation to model a ground robotic swarm on a search and detection mission in a semi-urban environment rigged with stationary IEDs. Efficient design of experiment (DOE) techniques and data farming are engaged to help identify controllable factors and capabilities that have the most impact on overall effectiveness. The focus of this thesis is to explore agent based simulation applied to swarm robotics; the technological and algorithmic aspects are not delved on. Results from the simulations provide several insights on the impact of both decision and noise factors on the performance of the swarm. Incorporation of virtual pheromones as a shared memory map is modeled as an additional capability that is found to enhance the robustness and reliability of the swarm.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ACO	Ant Colony Optimization
DARPA	Defense Advanced Research Projects Agency
Det Capab	Detector Capability
DoD	Department of Defense
DOE	Design of Experiment
DRT	Detector Reset Time
DSO	DSO National Laboratories
DTA	Defence Technology Agency
FCS	Future Combat Systems
FFW	Future Force Warrior
GPS	Global Positioning System
IED	Improvised Explosive Device
INEEL	Idaho National Engineering and Environmental Laboratory
IR	Infra-Red
JRP	Joint Robotics Program
LRF	Laser Range Finder
MANA	Map Aware Non-uniform Automata
MHPCC	Maui High Performance Computing Center
MOE	Measures of Effectiveness
MULE	Multifunctional Utility/Logistics and Equipment
NOLH	Nearly Orthogonal Latin Hypercube
NPS	Naval Postgraduate School

R&D	Research and Development
SA	Situational Awareness
SEED	Simulation Experiments and Efficient Designs
SUGV	Small Unmanned Ground Vehicle
TOT	Time on Target (for IED Detector)
TTP	Tactics, Techniques and Procedures
UGV	Unmanned Ground Vehicle
WMD	Weapons of Mass Destruction

ACKNOWLEDGMENTS

The completion of this thesis would not have been possible without my thesis advisor, Professor Susan Sanchez. I am grateful for her support in the pursuit of my interests and my choice of topic, as well as the flexibility she has given me in the conduct of this research. It was an honor to have worked with such a prominent professor in the field of simulation and design of experiments.

I am also thankful to the faculty and fellow students of the SEED Center. Professor Paul Sanchez provided tremendous help in scripting that literally saved me at least hundreds of man-hours, while Professor Tom Lucas has always been encouraging throughout the process of my research. Special acknowledgements should go to Volker Pfeiffer, Chris Michel, Dave Vaughan and Earl Richardson. All of you have been both great colleagues and friends who have elevated the meaning of knowledge sharing to a whole new level.

Not forgetting, Chwee Seng and Ee Chong from DSO National Laboratories have provided invaluable feedback and guidance throughout the entire process. I have also consulted many others and I appreciate very much the time and efforts that they have put in answering my queries (you should know who you are). I would also like to thank IPO and SEED Center for making my conference trip to The Netherlands possible.

Last but not least, I would like to thank Caz, who many a times have been the source of my motivation in the completion of my thesis and education here in NPS.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

The concept of employing ground swarm robotics to accomplish tasks has been proposed for future use in humanitarian de-mining, plume monitoring, searching for survivors in a disaster site, and other hazardous activities. More importantly in the military context and with the development of advanced explosive detectors, swarm robotics with autonomous search and detection capability could potentially address the improvised explosive device problem faced by foot patrols, and aid in the search for hidden ammunition caches and weapons of mass destruction.

Swarm robotics can be defined as the study of how a swarm of relatively simple physically embodied agents can be constructed to collectively accomplish tasks that are beyond the capabilities of a single one (Sahin, 2005). The origins of swarm robotics can be traced back to nature, where ant and termite colonies have demonstrated the ability to accomplish complex tasks by means of their collective emergent behavior while following simple sets of rules. Swarm robots have the characteristics of being simplistic and low cost, so that they could be manufactured and deployed in mass without being overly concerned about their survivability.

The intent of this research is to leverage on agent based simulation (specifically, Map-Aware Non-uniform Automata, or MANA) to model a ground robotic swarm on a search and detection mission in a semi-urban environment rigged with stationary IEDs. The technological aspects are not delved on in this thesis. Primarily, this research explores the following.

- How can agent based simulation be used to model a ground robotic swarm that searches and detects IEDs?
- What are the capabilities and characteristics of a ground robotic swarm that are critical to a search and detect mission?

The research in this thesis has the following objective in mind – to provide insights to swarm robotic engineers and developers on where they should invest their efforts on in the development of swarm robotics to be used in a military scenario.

Efficient DOE techniques and data farming are engaged to help identify controllable factors and capabilities that have the most impact on overall effectiveness. Results from the simulations provide several insights on the impact of both decision and noise factors on the performance of the swarm. The findings are summarized as follows:

- The number of robots, speed, and sensor (detector) range are the three main factors in determining the performance of the swarm.
- Possible quadratic effects are observed in number of robots, speed, and detector capability (time on target requirement).
- The results strongly suggest that a minimum threshold is required for the number of robots and speed. These thresholds are found to be realistic levels from the perspective of currently available technologies.
- Drastic failures are attributed to low speed settings.

The model is then extended to incorporate the capability of using virtual pheromones as a shared memory map that serves to enhance the spread and coverage of the robots (Wagner, 1999). Results from the simulations suggest significant improvement in performance. The proportion of mission completions increased from approximately 0.74 to 0.83, a 12% improvement over all scenarios in the experiment. More importantly it is found that virtual pheromones enhance the robustness and reliability of the swarm, making it more predictable with fewer dominant terms.

As for the modeling, there are certainly limitations to the capturing of all aspects of swarm robots and its technicalities in the simulation. The "weakest link" of the simulation seems to be modeling the movement of robots and emergent behavior of continuous coverage and spread, due to MANA's hard-coded movement algorithm. This can only be approximated in MANA with the presence of some artificiality. There are fewer difficulties in the modeling of robot capabilities, which are then varied to investigate their impact on the effectiveness of the swarm. There are currently no known prior efforts in using MANA to investigate swarm robotics, so this research also serves as an attempt to validate such an approach.

It is worthwhile to highlight that the insights from this research are applicable largely to a robot swarm with this type of algorithmic setup and detection routine. It is acknowledged by the author that there are many possible rules and routines that a swarm may adopt, but it is hoped that the one captured in this research is a general representation of a ground robotic swarm that is used for a search and detect mission.

In a nutshell, agent based simulation is found to have huge potential as a means to investigate swarm robotics and obtain insights on the impact of various factors on the overall effectiveness. Swarm robots produce much uncertainty in terms of its emergent behavior from multiple dynamic interactions, which is what agent based simulations were designed to examine. With the incorporation of an efficient DOE and data farming methodologies, roboticists and engineers should consider leveraging on this tool to assist in the development and progress of swarm robots to be employed in the real world.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. PROBLEM STATEMENT

U.S. and coalition forces involved in the ongoing military operations in Iraq and Afghanistan have had their hands full in dealing with roadside bombings and, in general, Improvised Explosive Devices (IEDs). Since the commencement of the offensive on Iraq in Mar 2003, there have been over 2,600 U.S. troop fatalities[1], of which approximately one-third are classified as IED fatalities (http://icasualties.org). In response, tremendous efforts have been invested by military, academic and commercial research organizations alike, to harness the latest technology available to come up with a viable solution to counter the IED problem.

For troops conducting foot patrols and door-to-door "flushing" operations, the problem lies in the inability to detect the presence or the location of IEDs in the vicinity of their operations, until a trooper stumbles onto one, or gets close enough that the IED is remotely detonated by the adversary. To solve the IED problem, it is vital that we equip our ground forces with the capability to search and detect IEDs effectively and efficiently without being exposed to the risks.

In general, the search and detect problem is not confined only to IEDs on the battlefield. Critical deficiencies exist in the ability to find hidden materials such as weapon caches, ammunition and explosives stashes, as well as mines buried underground. The basis of this thesis research is to explore a concept that can be developed to carry out search and detection in both wartime and peacetime, addressing the problem of uncovering the location of such targets as shown in Figure 1. For simplicity, the search and detection problem in this thesis will be discussed in the context of IEDs, but the reader should bear in mind that this is an overarching concept that can be extended to searching for other targets that are similar in type or class.

[1] As of Sep 2006

Figure 1. Munitions rigged for an IED discovered by Iraqi police in Baghdad, Nov 2005 (http://en.wikipedia.org/wiki/Improvised_explosive_device)

The concept explored in this thesis for search and detection of IEDs delves into the field of swarm robotics, i.e., a multiple robot system made up of small and simple robotic platforms mounted with "sniff-type" detectors. Imagine the deployment of a swarm of low-cost autonomous ground robots out in the field that are able to detect IEDs and transmit the suspected location to the commander prior to the deployment of his troops. This could have a significant impact in reducing IED fatalities.

B. BACKGROUND

Swarm robotics can be defined as the study of how a swarm of relatively simple physically embodied agents can be constructed to collectively accomplish tasks that are beyond the capabilities of a single one (Sahin, 2005). In reality, the use of swarm robotics to perform military tasks is only a concept in the research and development (R&D) stage. However, there have been many efforts and proofs of concept done on various aspects that will play a key role in the realization of swarm robotics being used on the battlefield, especially in the domains of mobile robots, autonomous cooperative robots, sensors and explosive detectors. These will be laid out further in Chapter II.

Swarm robotics has attracted much attention because of the numerous key advantages it brings, such as simplicity, autonomy, redundancy and ability to produce a desired emergent behavior without the need for a "human in the loop." Researchers and scientists, such as James McLurkin of MIT and iRobot Corporation, have successfully experimented with hundreds of small swarm robotic vehicles that could produce low-level emergent behaviors such as cluster, disperse etc., by pre-programming these entities to merely follow a few simple rules (http://people.csail.mit.edu/jamesm/swarm.php). On the detection front, leading R&D corporation ICx Nomadics (www.icxt.com) has recently developed a sub-3lb device that is able to detect explosive vapor and particles "as low as a few femtograms.[2] They have compared this capability to dogs.

Assuming that the various fields of technology could advance to a stage where swarm robots mounted with miniaturized navigational sensors and explosive detectors could traverse across real-life terrain autonomously, a swarm be employed to overcome the IED problem, which is expected to persist for many years to come with no imminent solution in sight. Nevertheless, if the advancement of these respective fields eventually does take longer than the persistence of the IED problem, search and detection using swarm robotics are still be applicable to many other scenarios such as humanitarian de-mining (Cassinis, 1998) and searching for survivors in a disaster site (Stormont, 2003).

C. MOTIVATION AND OBJECTIVE OF RESEARCH

Undoubtedly, the realization of such an idea will not occur without the development and integration of the various fields of technology. However, whether the different aspects attribute equally in terms of their importance to such a concept is an interesting question that, if answered, could save efforts and expedite the realization process. For example, a faster robot will provide good coverage in a shorter time, but if the detector requires a long time-on-target requirement, then the effectiveness will be hampered by the fact that the robot moves too fast and thus misses targets. In another instance, the lack of speed or mobility of a single robot may be compensated by increasing the number of robots, or vice versa. In fact, one factor of the robot or detector may enhance the overall effectiveness of the search much more than another factor, while some factors may not contribute significantly beyond a certain level; cases where "more is not more," or "more is not better." Such insights can be extremely useful to the developers and shorten the process needed to eventually produce such a capability.

[2] A femtogram is equivalent to a quadrillionth of a gram or 10^{-15}g

Thus, it is imperative that exploration with modeling and simulation proceeds alongside the technological development, to provide insights on what the critical decision factors (particularly technological capabilities), noise factors and other environmental factors are, in the employment of ground swarm robotics to search and detect IEDs. This is the primary goal of the thesis research.

On a different note, agent based simulation has become an increasingly popular tool to investigate various battlefield scenarios and military skirmishes. However, agent based simulations, particularly with MANA (Map-Aware Non-uniform Automata v3.2.1), have not been utilized as much to model autonomous robots or swarm robotics. The basis of swarm robotics lies in its complex adaptivity to produce a collective emergent behavior, which is precisely what Cellular Automation models like MANA were built to model and analyze, i.e., uncertain outcomes via interaction of agents (MANA Users Manual v3.0, Jul 2004). A brief survey of the literature reveals that simulations that have been used to model swarm robotics include Player/Stage (developed at the USC Robotics Research Lab and used by Batalin and Sukhatme (2002), Morlok and Gini (2004) and Rekleitis et al. (2004), MARSS (developed as part of Alistair Dickie's NPS thesis) and Extend (primarily a process simulation package, used by Dudenhoeffer and Jones (2000) from INEEL to model multi robot systems). Screen shots from Player/Stage and MARSS are provided in Figure 2. While little assessment has been documented on the appropriateness and flexibility of these simulation packages for the purpose of swarm robotics, it is certainly interesting to see how alternatives like MANA (and even Pythagoras[3]) measure up to the calling.

If indeed MANA has rarely been dedicated to model swarm robotics, then a valuable by-product of this research is the supplemental insights on the suitability of MANA to model this class of problems. Without going into the intricacies and dissecting the movement algorithms, some discussion and pointers will be put up as part of this thesis to provide any follow-on students with the strengths and limitations of modeling swarm robotics with MANA.

[3] Pythagoras is an agent based simulation package developed by Northrop Grumman

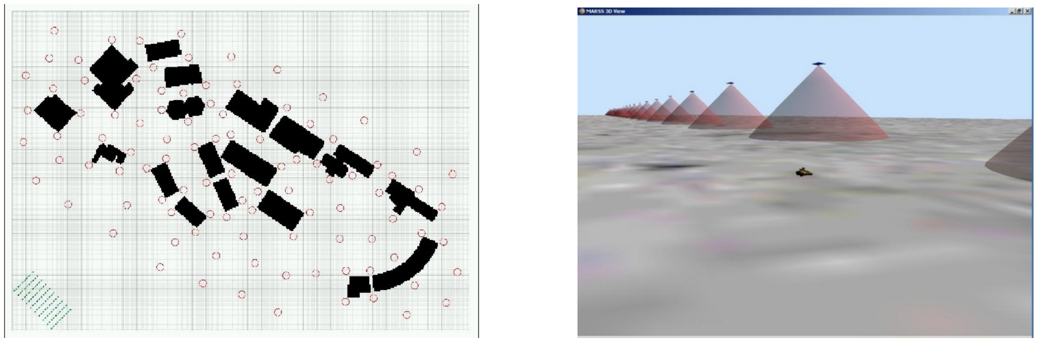


Figure 2. Simulation packages used to models robot swarms. Left: Multiple target localization problem environment setup by DSO National Laboratories in Player/Stage (Picture credited to a DSO National Laboratories report) Right: A 3D view from the MARSS software (Dickie, 2002)

D. APPROACH

In short, agent based simulation is chosen for its fundamental origins in modeling entities that produce an emergent behavior, much like the robotic swarm. Leveraging on DOE and data farming methodologies, a thorough analysis can then be performed using statistical analyses that will meet the objectives of this thesis research.

The decision variables of interest are the main robot and detector capabilities, specifically the number of robots, sensor range (detector range)[4], speed of robots, detector capability (TOT requirement) and detector reset time. The noise factors to be modeled are repulsion from fellow robots (for spread and coverage), repulsion from obstacles (obstacle avoidance) and precision of movement (an approximate representation of how much a robot deviates from the intended and supposed moves). These factors reflect the basic nature of autonomous robotic movement algorithm experimented by Batalin and Sukhatme (2002) as well as those examined by Morlok and Gini (2004). Different levels of terrain difficulty will be incorporated to investigate the impact of terrain on swarm effectiveness. The model is then extended to incorporate the usage of virtual pheromones as a shared memory map that strives to enhance the coverage and spread of the swarm. This idea has been experimented and advocated by at least three researchers who worked on the autonomous robot coverage problem (Wagner et al., 1999; Payton et al., 2004; Sauter et al., 2005).

[4] Sensor range and detector range are used interchangeably in this thesis

The scenarios are created in MANA which is an agent based simulation package written by New Zealand's Defense Technology Agency, DTA. DOE methodology (Sanchez, 2005) and a data farming tool (the Tiller provided by Referentia Systems) are utilized to set up a DOE to be run automatically on a high performance computer cluster in Maui (MHPCC). Analysis on the results is done using the statistical package, JMP (JMP: The Statistical Discovery Software™ v5.1). In addition, prior to the statistical analyses, some data extraction will have to be performed using batch files and scripts.

E. SCOPE AND THESIS ORGANIZATION

The goals of this thesis research are mainly to explore the concept of employing swarm robotics for search and detection of IEDs and to provide insights on the critical factors that influence most on the success of such a concept. Indeed, the scope will cover the attempts at modeling such a scenario, the procedures of the experiments and simulation runs, as well as the application of various simulation output analytical tools. With regards to identifying and spelling out of the details of the enabling technologies, this will be beyond the scope of this thesis. In addition, it is not the intention of the research to define the technological challenges we have at hand, and how such a "capability package" can be realized, which the author acknowledges. Rather, the motivation of this thesis lies in the hope that this research can help shorten the process of developing such a capability by providing insights such as "what's important and what's not" and "what's needed and what's not."

Overall, the thesis is divided into three main portions. The first portion will touch on a literature survey of swarm robotics and other related enabling technologies which will form the basis of how the swarm robotic agents are modeled in MANA. A detailed discussion on the scenario to be modeled and the formulation of Measures of Effectiveness (MOEs) will be made. There will be a discussion about how decision and noise factors are captured in the model and how accurately or inaccurately they are being modeled. This is all part of the scenario building and modeling.

The second portion of the thesis will describe how the DOE is set up and the data farming procedure. This will include how the MOEs are extracted from the output files generated and imported into the statistical package. This will be followed by an in-depth data analysis using tools like data plots, curve fitting, regression trees, analysis of variance (ANOVA) and other analytical methods.

The final portion of the thesis will encompass a summary of the limitations and conclusions, including the suitability of MANA to model swarm robotics. This will be followed by some proposed future work and possible spin-offs from this thesis research.

Specifically, the above-mentioned is laid out in as follows.

Chapter II includes a literature survey on swarm robotics and other related fields of enabling technology that form the basis of the agents in the scenario.

Chapter III describes the scenario in detail and contains a discussion of the MOEs we attempt to draw conclusions from. There will also be a discussion on how various aspects of the swarm robots and the detector are being captured in the model, along with the qualitative factors of interest. Most importantly, the assumptions of the model will be listed and deliberated.

Chapter IV will involve the formulation of the DOE and the data farming methodology, which is followed by the treatment of output to extract the relevant MOEs.

In Chapter V, the data are summarized, the analysis tools are introduced, and analyses are performed using data plots, curve fitting, regression trees and other simulation output analysis methods.

In Chapter VI, the overall conclusions from the analysis are presented, including a short assessment of the suitability of MANA as a simulation package to model swarm robotics. This chapter also briefly discusses future research opportunities and spin-offs based upon the current work done and extensions that could be explored.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

A. ROBOTICS ON THE BATTLEFIELD

In recent years, robotics has taken a huge step toward getting involved on the battlefield. According to Col. Edward M. Ward, logistics chief of the Robotic Systems Joint Project Office (RS JPO) at Redstone Arsenal, U.S. military forces in Afghanistan and Iraq were operating over a total of 2,400 combat robots in 2005 (www.decaturdaily.com, 11 Apr 2006). The impact of employing robotics in the battlefield is irrefutable. Military robotic platforms are now frequently employed by troops to perform highly dangerous tasks such as bomb disposal and site reconnaissance, functioning to take the man out of the loop. Instances have been reported where remote controlled robotic platforms used by troops for reconnaissance were blown off by booby traps (illustrated in Figure 3), proving the extent of risk mitigation these robots have provided (FY2005 JRP Master Plan).

Figure 3. A robot destroyed by an IED in Iraq. (FY2005 JRP Master Plan)

A robot employed by the U.S. military that has enjoyed much success is the MARCbot (shown in Figure 4), which is now in its fourth variant. It is used to help dismounted soldiers who are performing IED sweeps to remotely interrogate suspicious objects. The MARCbot IV is able to provide remote observation of over 100m and traverse across challenging terrain and obstacles. It was reported in Aug 2005 that over 300 MARCbot IVs were procured and will be deployed by end of 2006 (www.estripes.com, 19 Aug 2005). In an interview with Director of Rapid Equipping

Force, Col. Gregory Tubbs (www.defenseindustrydaily.com, 1 Jun 2006), it was revealed that there had been a one-week period where MARCbots interrogated 32 potential IEDs, of which 26 turned out to be actual IEDs.

Figure 4. The MARCbot IV in action (www.defenseindustrydaily.com)

According to the JRP Master Plan, it is acknowledged by the Services that current and future unmanned ground systems play a "critical warfighting role." Particularly, the integration of robotics into the Army is demonstrated in the Future Combat System (FCS) as part of its Future Force Warrior (FFW) program as shown in Figure 5. It is projected that unmanned systems, in the form of UAVs and UGVs from man-packable, sub-30lbs ones like the SUGV to vehicular platform systems over 30,000lbs like the MULEs, will be heavily leveraged upon to address the spectrum of potential threats that any adversary pose in tomorrow's battlefields (www.globalsecurity.org).

Figure 5. Overview of U.S. Army's FCS Program (www.globalsecurity.org/military/systems/ground/fcs.htm)

The Master Plan lays out the four key technological thrusts, namely man-portable robots, intelligent tactical behaviors, innovative platforms and autonomous mobility. In addition, an evolution roadmap was charted out as depicted in Figure 6, with plans for robot autonomy being featured on the battlefield by 2020. UGV applications listed in the Master Plan include minefield detection and neutralization, reconnaissance of unexploded ordnance, and search and rescue operations in peacetime.

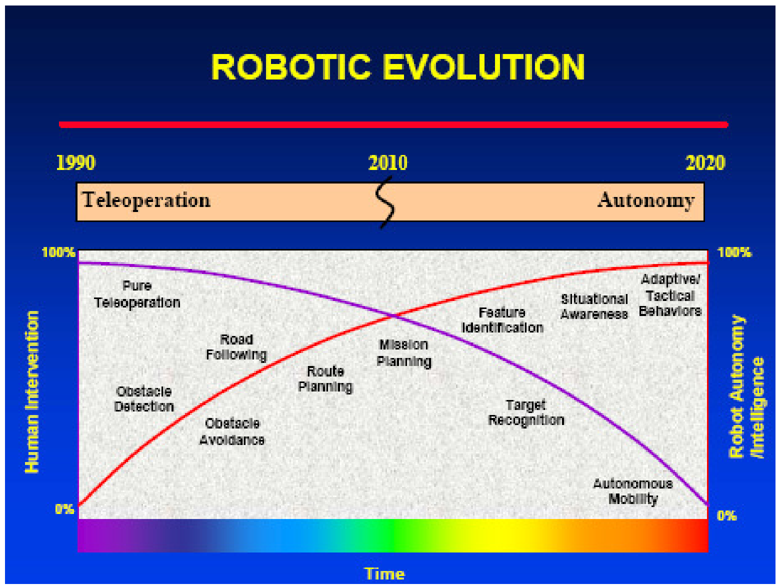


Figure 6. Robotic evolution (FY2005 JRP Master Plan)

Of closer relevance to the type of robots explored in this thesis is a program in the Master Plan to acquire the ThrowBot (the latest variant is known as the COTS-M by ReconRobotics). The ThrowBot (as seen in Figure 7) was conceived at the University of Minnesota and is designed to provide additional situational awareness to dismounted troops. It is a small, cylindrical, robotic platform that is designed to be thrown into potential areas of interest by soldiers, who then remotely operate it to search the area before they enter. The ThrowBot measures less than six inches in length, weighs under 12 ounces, and is equipped with a video camera that transmits streaming video to the controller. The first ThrowBots were evaluated in Jun 2004 with several deficiencies identified, but since then improvements have been made to it continuously (Kratochavil et al., 2003). Essentially, one could think of swarm robots as being an advanced variant of the ThrowBot, except in greater numbers and capable of self-organized autonomous movement and target-detection.

Figure 7. Left: A recent ThrowBot variant called the COTS-Scout. Right. A larger ThrowBot variant, the “MegaScout” with actuated-wheels (Kratochvil, 2003)

On another front, the Defense Advanced Research Projects Agency (DARPA) has actively funded projects in the field of swarm robotics and distributed robotics. The agency sponsored the Centibots Project in 2004 (www.ai.sri.com/centibots), which aimed to deploy up to 100 autonomous robots for missions such as urban surveillance, searching and tracking. In 2003, Icosystems was funded to develop ways to carry out missions such as minesweeping and search and rescue with minimum intervention from human operators using a squad of 120 robots fitted with swarm intelligence software (www.newscientist.com, 25 Apr 2003). Leading swarm roboticist James McLurkin, who created a swarm of 100 robots running on swarm algorithms, was similarly sponsored by DARPA from 2002-2004. His work continues to be developed at iRobot as part of the R&D program; it is an integral stepping stone to the goal set in the National Defense Authorization Act of 2001 to have one-third of the operational ground combat vehicles of the U.S. Armed Forces to be unmanned by 2015 (NDA Act, 2001).

The preceding section is a brief overview of how swarm robotics is slowly coming online in several military related initiatives. The following section will focus on the characteristics of swarm robotics and some accompanying technologies that are being explored and modeled in this research.

B. SWARM ROBOTICS

Swarm robotics can be defined as the study of how a swarm of relatively simple physically embodied agents can be constructed to collectively accomplish tasks that are beyond the capabilities of a single one (Sahin, 2005). The pioneer of the concept of integrating robotics with swarm intelligence is Professor Gerardo Beni who, together with Professor Jing Wang, coined the term "swarm intelligence" in 1989 (Sahin, 2005; Kennedy and Eberhart, 2001). Beni strived to make a distinction between swarm robotics and multiple robot systems, a term which was already in existence. Where multiple robot systems are appropriate whenever several robotic platforms are used to achieve a mission, swarm robotics emphasizes self-organization and emergent behavior, and focuses on issues of scalability and robustness. The concept of swarm robotics is envisaged to involve the use of huge numbers of low-level robots that are cheap and hence dispensable. Decentralized control strategies are made possible by individual robotic platforms having localized sensing abilities and scalable communication means (Beni, 2004).

A set of criteria that distinguishes swarm robotics research was promulgated by Erol Sahin during the Swarm Robotics International Workshop in 2004 (Sahin, 2005). The following are the main criteria, along with are some examples for better illustration.

1. Large Number of Robots

A robotic swarm should consist of large numbers of robots with the population size varying anywhere between the order of 10^2 to $10^{<<23}$. James McLurkin, who back in 2003 invented the world's smallest self-contained autonomous robot (measuring a little over one inch on each side as shown in Figure 8), built a fleet of over 100 of such robots to investigate swarm behavior as part of his doctoral research and study for iRobot. His software and programming techniques are scalable to swarms in the 10 to 10,000 range (www.irobot.com).

Figure 8. McLurkin's Swarm Robots
(http://people.csail.mit.edu/jamesm/swarm.php)

2. Homogeneous Robots

The system should consist of one, or relatively few homogeneous groups of robots, and that the number of robots in each group should be large. Where McLurkin's experimental swarm robots "fit the bill," a good counter-example is that of a robo-soccer team found in Robocup competitions (D'Andrea, 2003) as shown in Figure 9. In general, various roles are assigned to sets of robotic platforms in robo-soccer, such as goalkeeper, attacker, etc. In such an environment where many small groups of homogeneous robotic platforms exist, it will not be considered as a robotic swarm although they are autonomous in operation.

Figure 9. Robo-soccer platforms built by Cornell University in 2003
(www.cornell.edu)

Conversely, in a scenario where a large number of robots are deployed primarily to search and detect targets in a hostile environment, there may be a strong motivation to deploy two sets of homogenous robots, e.g., one that only does search and detect of targets and another that defends against and eliminate hostilities. This concept would probably be deemed more "swarm robotic" in nature than the robo-soccer robots.

3. Simple Platforms, Incapable Individually, Capable Cooperatively

Swarm robotic members should be relatively simple and incapable such that accomplishment of tasks requires the cooperation of robots. Part of a key characteristic of swarm robotics is the emergent behavior that results from the individual platforms' conformance to a simple set of overarching rules, which have no direct causal relationship to the behavior.

The inspiration of this originates from nature, where low-level organisms such as ants deposit chemicals called pheromones along the path of its movement. If a swarm of ants is put in an environment of multiple alternative routes to the target (more often than not, a food source), it turns out that the swarm will eventually figure out the shortest path to the target and converge on this path over all alternatives. This phenomenon happens because the pheromone deposit evaporates over time, and the intensity of pheromones remains stronger on shorter paths than longer paths if each path starts off being equally utilized. The solitary golden rule of "go to the path that has the highest concentration of pheromone" hence causes the ant to converge on using the shortest path over time, a remarkable end-product as a result of individual conformity to a simple rule. The scientific term of this process is known as Ant Colony Optimization or ACO (Bonabeau, 1999) as depicted in Figure 10.

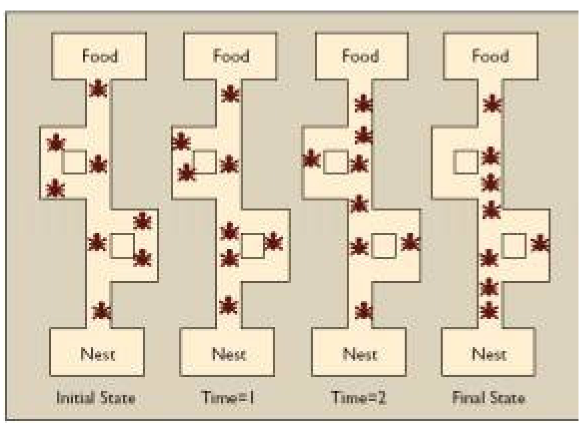


Figure 10. Time-snapshot illustration of Ant Colony Optimization (Tarasewich, 2002)

Recently, a project sponsored by the European Commission called SWARM-BOTS was headed by swarm intelligence guru Marco Dorigo to study a novel approach to the design and implementation of self-organizing and self-assembling artifacts

(Dorigo, 2004). The objective was to construct a number of simple robots with cheap components, that together they are able to self-organize and cooperate to adapt to the environment. One of the graphical simulations from the project precisely illustrates the idea of "incapability of individual robots achieving a task, but cooperatively they are able to". This is shown in Figure 11, which depicts the swarm-bots making use of their gripper arms to grip onto one another to cross over a gap that is wider than any individual platform. This is a simple concept that requires these robotic entities to merely follow a set of rules, i.e., gap detected---form line---grip fellow robot---move towards gap---cross gap. It is not clear in their report if this particular task was achieved by the physical platforms that were constructed, but the project team was successful in getting the robots to cooperatively transport an object that cannot be moved by a single swarm-bot.

Figure 11. Swarm-bot project simulation of concept of crossing gaps (Dorigo, 2004)

4. Localized and Limited Sensing and Communication Abilities

The swarm robots should only have localized and limited sensing and communication abilities. This characteristic can be seen as a by-product of keeping the robot entities simple and cheap. However, with technological advances, there should be no limit on the robots' sensing and communication abilities as it does not increase complexity or cost, or eliminate any key characteristic of swarm robotics such as scalability.

There have been some recent successes by Yamauchi in getting man-portable UGVs to perform autonomous reconnaissance in urban terrain that could build a grid-map of its surrounding terrain and feedback to the user in near to real-time as shown in

Figure 12 (Yamauchi, 2006; www.robotfrontier.com). While such technology may lie beyond the boundaries of "limited sensing" and overly-complex to be fitted on a swarm robot in current context, this may not be true in the future.

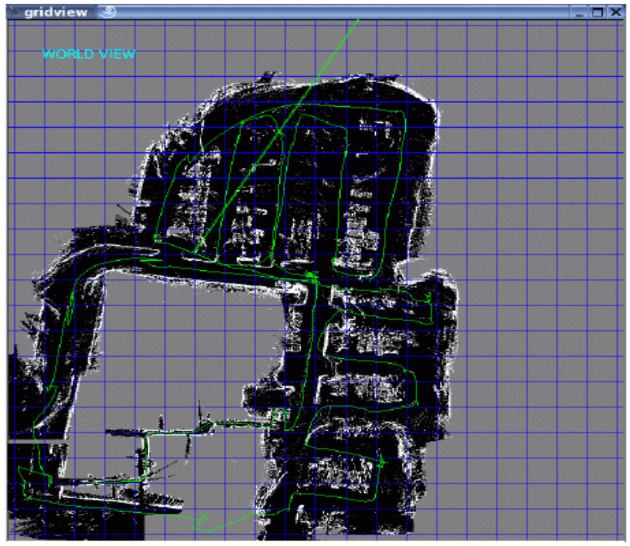


Figure 12. Mapping of exterior and interior of vicinity using pure odometery (Yamauchi, 2006)

An example of the sensing and communication capability of swarm robots is taken from the research of Bruemmer and Dudenhoeffer from INEEL, who worked on using robotic swarms for spill finding and perimeter formation (Bruemmer, 2002). Their research is in line with the concept of swarm robotics, "deploying and tasking of a real-world collective of cost-effective, small mobile robots and to escape the limitations of centralized control". The robotic entities (as shown in Figure 13) in their experiments use multi-modal communication architecture including acoustical chirping, infrared (IR) and radio frequency (RF) transmissions. Specifically, each robotic entity is comprised of two processors, one for communication and the other for navigation. The sensors equipped include a spill detection sensor, two bump sensors, two whisker-like light sensors, four IR sensors for obstacle avoidance, a ring of IR for local communication, a piezoelectric speaker and two directional hearing aid microphones. This is probably a good representation of how well sensing and communication abilities can be currently be packaged into a swarm robot.

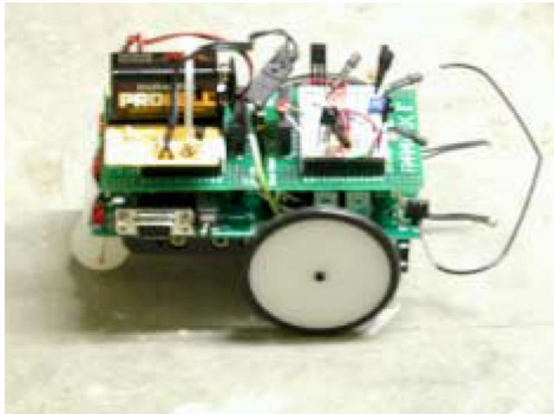


Figure 13. The "GrowBot" built by the INEEL research team (Bruemmer, 2002)

5. Key Advantages of Swarm Robots

There are plenty of advantages that arise from the preceding criteria for swarm robots that were proposed by Sahin. Some of these have been proclaimed by Beni; that swarm robots could be, in principle, mass produced, modularized, interchangeable and disposable. A swarm's high reliability and robustness comes from its inherent redundancy collectively, which allows it to adapt dynamically to the working environment.

In conclusion, swarm intelligence addresses the problems of computational intensive, single point of failure and heavy communication requirements that arise with other types of robotics. Each robot is an independent entity that acts on information that is available via its sensors. Cooperation emerges through individual behaviors and cross-interactions. This distributed form of approach allows for fast response to ever-changing conditions and reduces dependency on communications requirement. Little computation is needed since each robot only needs to execute its own activities. Overall, the system is more robust and is scalable to larger numbers of robots.

C. CHOOSING A SWARM ROBOTIC BASIS FOR MODELING

From the brief survey of swarm robotics, it is revealed that even within this specific field, there exist numerous variants and levels of swarm algorithms that have been researched and experimented.

At the higher end of complexity is the type of swarm robot that closely mimics swarms found in nature. The group of Wagner, Lindenbaum and Bruckstein investigated the ability of a group of robots to communicate by leaving traces, in order to perform the task of cleaning the floor of an un-mapped building, or any task that requires coverage of unknown areas. Their experiments involved robots that leave chemical odor traces that evaporate with time. Through the ability to evaluate the strength of smell at every point, the robots are able to select paths to go on, resulting in an overall desired emergent behavior of the swarm (Wagner et al., 1999). This novel approach to induce exploration and coverage in swarm robots is inspired from stigmergy (exhibited in insects like the ACO illustration in the previous section), and is highly sophisticated as far as modeling the algorithmic aspect of it is concerned.

McLurkin's robots also take inspiration from swarms in nature. His objective was to use local interactions between nearby robots to produce large-scale group behaviors from the entire swarm. He has formulated, using standard C functions, a vast array of "group behavior building blocks" of code that can be combined to form larger, more complex applications. They range from simple tasks like dispersing and clustering, to complex tasks like temporal synchronization and gradient tree navigation.

On a simpler level in terms of function and algorithm is the Growbot (Bruemmer et al., 2002) that implements social potential fields using a combination of IR, obstacle avoidance, light sensing and audible chirping. Through these behaviors, each robot is able to exert attractive and repulsive force fields, which then control the dispersion and coverage of the swarm.

Last but not least, Batalin and Sukhatme (2002) focused specifically on the problem of multi-robot coverage, in an environment where there is no prior map or information. The motivation of their problem stems from the exploration problem of unknown environments such as Mars and in the urban search and rescue domains.

It is timely to reiterate here that the emphasis of this research is not on optimizing swarm robotic movement algorithms. Rather the intent is to model a desired emergent swarm robotic behavior that can be approximately replicated in agent based-simulation, and from there, investigate what decision factors or capabilities, such as number of robots, speed, sensor range, etc., would impact the effectiveness of the swarm in searching and identifying IEDs in an unknown environment. However, if the reader seeks the basis which to which the simulations in this research are built and modeled upon, then Batalin's *Molecular* approach algorithm would probably be the closest. As will be elaborated, the robots modeled as agents in the simulations will only given a propensity to repel from each other when a neighboring robot is sensed. In some sense, this is similar in principle to Batalin's proposed algorithms (to be discussed in the next section), which produce good global coverage. However, it is more appropriate to treat the robotic movement algorithm modeled in MANA as a blackbox, on the premise that the emergent behavior of good swarm coverage does manifest in the simulations.

The next section provides a brief overview for readers who would like to gain a better understanding of Batalin's proposed algorithms.

D. BATALIN'S ALGORITHMS

The robots that are being modeled in the agent based simulation follow the principles of the robots that Batalin and Sukhatme discuss in their published article "Spreading Out: A Local Approach to Multi-Robot Coverage." The choice of adopting the principles of Batalin's robots is two-fold. Firstly, it is attributed to the fact that his system seeks to achieve natural global coverage based on simple, local interactions between robots. Secondly, the simplicity of the principles behind Batalin's algorithm allows it to be captured on MANA up to a satisfactory level of accuracy.

Batalin's research is motivated by the general exploration problem such as exploring the Mars surface as well as urban search and rescue missions, where maps, blueprints, and mapping devices such as Global Positioning System (GPS) are unavailable or inaccessible. This is similar to the problem in context, where a swarm of robots are being deployed to search and detect for IEDs in an urban environment where the operator has no prior knowledge of the layout. From his findings, the premise of good coverage of a multi-robot system was found to be that of local dispersion. This was realized from his simulations of three different proposed approaches, which he termed Informative, Molecular and Basic. Batalin found that the Informative and Molecular techniques attained within five to seven percent of the (manually generated) optimal solution and significantly outperformed the Basic technique. Moreover, the Molecular approach did slightly better than the Informative approach.

Batalin defined the Informative approach to incorporate the exchange of identities of interacting robots when robots are within each other's sensor range. This approach relies on ephemeral identification, local identities and mutual relative location information to cooperatively spread out in a coordinated manner.

The Molecular approach is simpler than the Informative, in the sense that there is no communication between robots and no local identities are created. Each robot simply gets repelled from all its neighbors that it senses at any time and there is no concerted effort to coordinate dispersion. The robots in both approaches are able, however, to tell the difference between a fellow agent, a target, and obstacles by their array of sensors.

The Basic approach is simply a degraded form of Molecular; there is no distinction made between fellow agents and obstacles. In general, the three levels of approaches can be viewed along a continuum of being able to communicate and share information with each other to not being able to do so; and subsequently from being able to differentiate between fellow robots and obstacles to not being able to do so.

Batalin's objective for his multi-robot system is to attain maximum global coverage (i.e., maximum spread), in a scenario where a team of robots is thrown into a catastrophic site and activated. The system will act as a communication network to be used by rescue workers to find humans and casualties. It was not clear if the robots would end up in a stable state where each will wander about a position, or whether they would continue moving throughout the environment while still maintaining the spread that has been achieved. However, Batalin did mention that his simulations terminate either when a pre-specified time threshold is exceeded, or if the locations of the robots have not changed for a certain amount of time (which may not be attainable if we have a small number of robots and limited sensor range).

Batalin stated in his paper that "the proposed techniques are adaptive"—a key characteristic of swarm behavior. The techniques proposed by Batalin are behavior based, meaning the robots follow a predetermined set of prioritized rules. These include obstacle avoidance, walk, observe and dance, in that order of priority as shown in Figure 14. The execution of each level of function is based on sensor information input at all times.

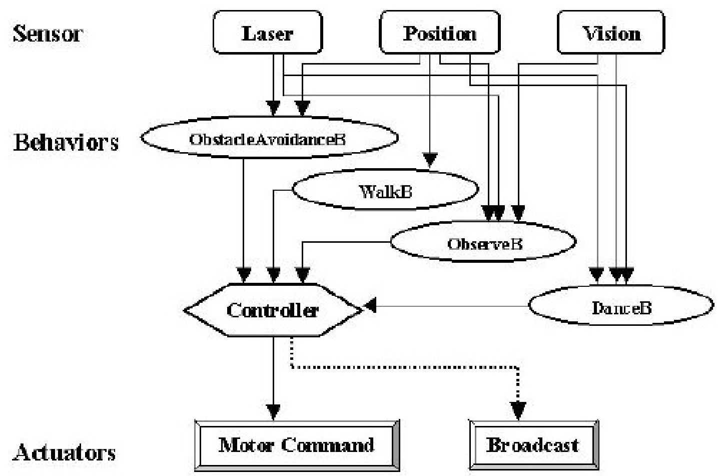


Figure 14. System architecture of Batalin's robots (Batalin, 2002)

The movement algorithm in MANA is based on a "best move choice," i.e., calculating the penalty functions of all its possible next moves and then making a choice at every time step. The intention here is not to implement exactly the algorithm formulated by Batalin or any other swarm robotic algorithm. The objective here is to incorporate the basic principles of Batalin's algorithm in the effort to **replicate the emergent behavior**. It is found in the simulations created that, with the incorporation of these principles, we are able to produce a behavior that closely resembles Batalin's robots. Our robots spread out in the attempt to attain the maximum coverage.

Another interesting point that Batalin makes is that the reason why the Molecular approach outperforms the Informative approach was hypothetically due to the additional overhead of passing additional information and attempting coordinated local coverage analysis whenever the robots could sense one another. Batalin further emphasized that the ability of the robots to tell each other apart from obstacles is critical, as illustrated by the superiority of Informative and Molecular approach over the Basic approach.

Batalin went on to question the definition of steady state and whether a static state needs to be achieved before steady state can be attained. Interestingly, he acknowledges that different problems or scenarios may require a different steady state. In Batalin's simulation, the desire was to achieve a steady state where the robots do not move much relative to their positions after spreading out. However, there may be other scenarios where the desire is to have continuous movement throughout the environment, which he terms as "patrolling steady state." In fact, this is exactly the scenario in our context as our aim is for the swarm robotic agents not only to achieve a spread, but also to be moving constantly throughout the environment in order to detect IEDs in both explored and unexplored areas. Our MOE in the scenario depends on this patrolling behavior, i.e., more patrols equate to multiple detections by different robots. This different aspect of "coverage" is in fact "exploration," and this is what Batalin intends to focus his efforts in his future work.

E. OTHER ENABLING TECHNOLOGIES

Clearly, the deployment of intelligent swarm robots to search and detect IEDs in a real environment is only half the battle won. Nothing can be achieved without other requirements such as mobility and IED detection capability. The following section addresses these technological aspects to give the reader some idea of where we currently stand.

1. Mobility of Mini-Whegs, by CWRU

The concept of swarm robots revolves primarily on its characteristic of simplicity and ability to produce an emergent behavior. However, the realization of this concept will not happen if a swarm does not possess the physical ability and speed to move autonomously across difficult terrain and negotiate around unknown obstacles. This is probably the most challenging technological aspect that faces researchers dealing with any type of autonomous robot that need to be deployed in real-life terrain. This problem is further amplified for the case of swarm robots because of the criterion that they be small and simple. For swarm robots to overcome the problem of autonomous mobility, a possible direction is to look into biologically-inspired robots.

Biologically-inspired robots are basically robotic platforms that are designed and built to resemble systems found in nature such as insects. Some researchers try to reverse-engineer living things in nature to solve problems. One of them is Otto H. Schmitt, who coined the term biomimetics back in 1969, which means “the subject of copying, imitating and learning from biology.” For billions of years, nature has been the laboratory to everything that existed. The challenges it posed to all living things results in survival of the fittest and natural selection. Mimicking nature and its functional morphology could lead us to find efficient ways of how we go about performing certain tasks. Robots have been built based on the neuromechanics of animals such as cockroaches, crickets, snakes and even lobsters (Ayers et al., 2002). Each of these species is highly efficient in mobility in their own domains. For example, the lobster is able to efficiently navigate and crawl along the sandy bottoms through turbulent and murky waters near shore. A robot that successfully mimics this morphology could be the solution for automated searching and disarming mines along shorelines.

In our context, one particular platform that could address the mobility problem in ground swarm robots is the Mini-Whegs, built by the Biologically Inspired Robotics Lab at Case Western Reserve University (CWRU) directed by Dr. Roger Quinn (Schroer et al., 2004; http://biorobots.cwru.edu). The Mini-Whegs robot (as seen in Figure 15) is a small and simple, highly mobile robot that takes advantage of a movement mechanism called a “wheel-leg.” This design combines the advantages of wheels and legs, allowing the platform to climb obstacles higher than that of a wheeled platform. The robot measures approximately 8-9cm (~0.1m) and can run over 10 body lengths per second (~1m/s).

Figure 15. Mini-Whegs™ 1 (http://biorobots.cwru.edu)

A modified version of the robot (seen in Figure 16) has been developed that can overcome a step of up to two or three body lengths high. This variant is similar in terms of its normal mobility, but boasts a spring loaded mechanism that extends a retracted device to enable the jump to be made.

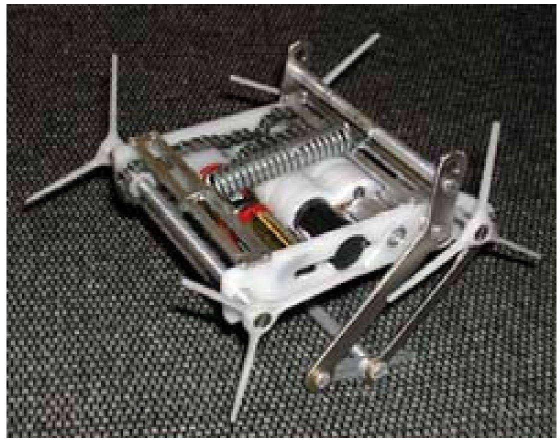


Figure 16. Jumping Mini-Whegs™ 1 (http://biorobots.cwru.edu)

The Mini-Whegs possess the basic mobility that swarm robots require. In fact, for challenging terrains, the platform may well have to be more advanced than this in terms of its mobility. The ultimate goal is to create a swarm robotic platform that combines mobility with the intelligence and autonomy of robots such as those experimented by Batalin, McLurkin and Wagner. The resulting robotic platform from the combination is what the agents modeled in the simulation experiments are based on. Whether the technology could support the realization of this "combo-platform," is beyond the scope of the thesis. However, it is hoped that the results and analysis in this thesis could shed some light on where engineers and researchers should invest their efforts on the concept of swarm robotics in a search and detect mission.

2. Explosive Detector – FIDO XT by ICx Nomadics Inc

The U.S. military currently has no robust solution in looking for explosives materials such as IEDs, roadside bombs and munitions caches. Employment of remotely-operated robots is fast becoming a popular option among troops to interrogate suspected sites because it completely removes the risks faced by troops. Unfortunately, without detection abilities, swarm robots will be ineffective as we will have no idea where to employ them. The only way of employment would be to send the robots into suspect sites and inspect by visual links, which is the current conventional method of bomb disposal and IED interrogation.

The FIDO XT that is used by U.S. military (shown in Figure 17) is claimed to be able to detect explosive vapors through bomb casings and landmines buried six inches underground (www.icxt.com). The device can also be used at checkpoints to detect traces of explosive residues on the skin of bomb makers. A second generation of this detector has been claimed to be at least 30 times more sensitive and could be available within one to two years (www.forbes.com, 3 Mar 05). The detecting process is done real-time; after an explosive detection, the sensor refreshes itself (resets to baseline) in a few seconds to perform the next detection. This current model weighs less than three pounds and is suitable for robot-mounted operations.

Figure 17. FIDO XT by ICx Nomadics Inc (FIDO XT Brochure)

The modeling of the detection capability of the swarm robots in the agent based model will take after this type/category of detector, with the assumption that there will be a miniaturized version that can be mounted onto a swarm robotic platform without impeding its mobility and intelligence. Again, the technological aspect and possibility of incorporating this "future" version of detector is beyond the scope of the thesis. However, it is hoped that some inference on how detection capability (TOT) and characteristic (such as detector reset time) impact mission success could be drawn from the analyses.

III. SCENARIO AND MODEL DEVELOPMENT

A. SCENARIO INTRODUCTION

The scenario is based on a swarm of robots mounted with an IED detector and an array of sensors, deployed in an unexplored area of operations to search for IEDs planted by the adversary in unknown locations, prior to troops being sent in to conduct flushing operations or gather intelligence materials. Once any detection is made, the information is sent back via the communications link to the operator, who is situated just outside the area of operations in a safe location. There will be no requirement for operator command and control during the search process. After a 30min (equivalent to 18,000 time steps simulation time) search window, the commander concludes the locations of the IEDs and will determine the next step to be carried out. In the agent based simulation, the focus will only be on modeling the search and detection process. Each time step of the simulation is assumed to represent 0.1sec of mission time.

B. CHOOSING AN AGENT BASED SIMULATION SOFTWARE

The software used to model the scenario is MANA v3.2.1 (Map-Aware Non-uniform Automata), which is an agent based simulation package written by New Zealand's Defence Technology Agency, DTA, to look into implications of chaos and complexity theory for combat and other military operational modeling (MANA Users Manual v3.0, Jul 2004). A large part of the inspiration of investigating swarm robotics for this thesis is attributed from initial observations of existing MANA scenario models. MANA is typically used to model large numbers of agents with personalities manifested in the form of movement propensities. Furthermore, the capabilities of agents in the sensor, weaponry and communication departments can be captured with significant fidelity in the model. What is powerful about MANA is that it incorporates stochasticity in its simulation by using a random seed generator, allowing for probabilistic events to be factored in such as detections of target and movement disparities from intention. During the simulation, the continuous interactions of agents and the dynamics of personality and capability settings result in an emergent behavior and outcome. This is, in fact, the core principle of swarm robotics; a mass of simple robots given a set of rules to follow autonomously that will produce a desired emergent behavior. It is felt that MANA is a tool that was highly suitable in modeling the concept of swarm robotics and could deliver excellent insights.

MANA is also relatively more well-established in the agent based simulation arena than other modeling platforms. It is a popular package utilized in the NPS SEED Center as well as the military research arm in Singapore, particularly DSO National Laboratories. With the incorporation of an efficient DOE and the access to high performance computers, we are able to data farm over a variety of variables simultaneously and quickly, each with several replications, something that deterministic models are unable to do.

C. TERRAIN

The types of environment and terrain where swarm robots will operate are as vast as where troops operate in both the modern and future battlefield, ranging from multiple story buildings with stairwells and vertical climbs to plain flat open terrain with little obstacles and hazards. Just as robots are currently phased into the battlefield, it is expected that swarm robotics will probably be deployed in less challenging terrains in the beginning, in vicinities with easy access throughout and few obstacles. A terrain file is picked that is based on a semi-urban residential area (mostly one to two story houses) in the Jolan District in Fallujah, Iraq. This was created by Capt Mike Babilot, USMC, as part of his NPS thesis (Babilot, 2005) on distributed operations in urban combat. Babilot's terrain file is based approximately on a 200m by 200m portion of the Jolan District as shown in Figure 18.

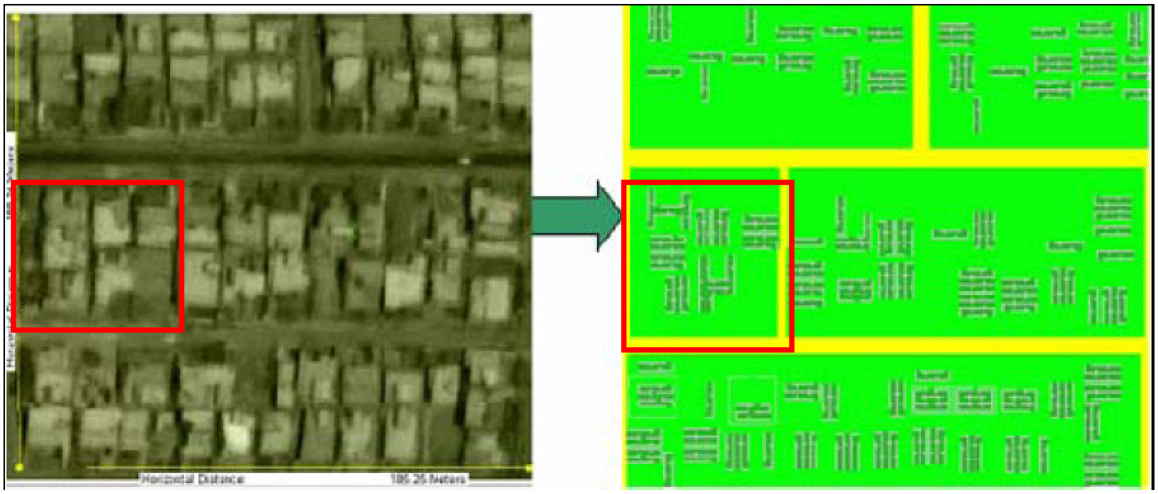


Figure 18. 200m by 200m of Jolan District of Fallujah, Iraq converted to MANA terrain file. Blocked in red is the 50m by 50m extract shown in Figure 19 (Babilot, 2005)

Preliminary test simulation runs reveal a terrain of this size proves too challenging for a manageable (from the agent based software perspective) robot swarm with a common starting point. Decent coverage within the pre-defined 30min mission time is only attained using immensely more or faster robots, or having the swarm split into multiple starting locations. Since the findings from a simulation based on a smaller area are scalable to a larger area given that the setup of robots can be replicated (i.e., using two similar setups of robots will give similar results if area is doubled), the terrain size is scaled down to 50m by 50m as shown in Figure 19.

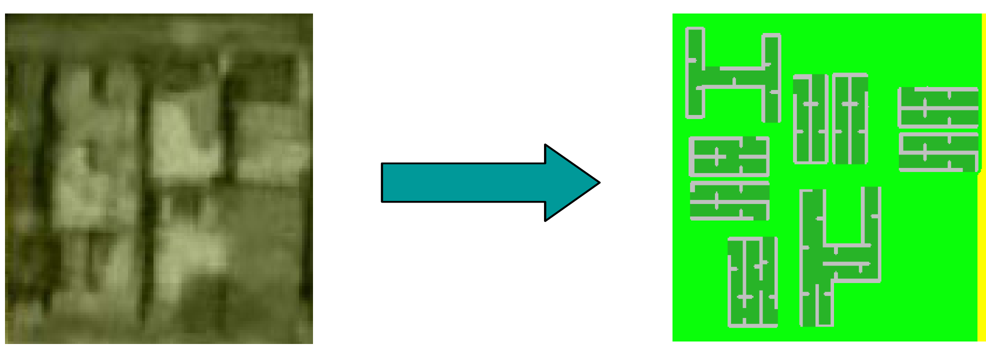


Figure 19. 50m by 50m extract of Jolan District

The terrain settings are then adjusted to accommodate the context of the scenario as tabulated in Table 1. Cover is adjusted to zero (except for walls) but this does not matter in the final model as weapons are not involved in the modeling. Concealment is set to zero (except for walls) so that any IED detection is attributed solely to the capability of the detector. In other words, the capability of detecting an IED inside or outside a house is similar, if the detector capability is unchanged. This is also done to ensure no detection can be made through walls. The rest of the terrain settings are left at default values. The implication is that the dark green terrain represents higher movement impedance, taking into account the obstacles faced by the robotic entities within the houses as compared to the exterior environment. **(Note: the different shades of green in the terrain can only be viewed in the soft copy version of this thesis.)** Grey simply represents the walls of the houses. Yellow represents the road and provides slightly better mobility than the light green area exterior of the houses.

Edit Terrain Properties

	Going	Cover	Conceal	Red	Green	Blue
BilliardTable	1.00	0.00	0.00	0	0	0
Wall	0.00	1.00	1.00	192	192	192
Hilltop	0.90	0.00	0.00	64	64	64
Road	1.00	0.00	0.00	255	255	0
LightBush	0.75	0.00	0.00	10	255	10
DenseBush	0.20	0.00	0.00	40	180	40

New Edit Delete Close

Table 1. Terrain settings for Going, Cover and Concealment in MANA

Three different terrains with increasing levels of difficulties are also incorporated to investigate the impact of terrain on the performance of the swarm. The baseline terrain used is shown in Figure 19. The second terrain incorporates, to a certain extent, aggregated objects and obstacles within houses, with additional patches of difficult terrain in the exterior of the houses that can be thought of as rough grass patches and debris. The third terrain is similar to the second except that there are additional sealed doorways, allowing investigation of the impact of limited access in and out of the houses. The second and third levels of terrain are as shown in Figure 20.

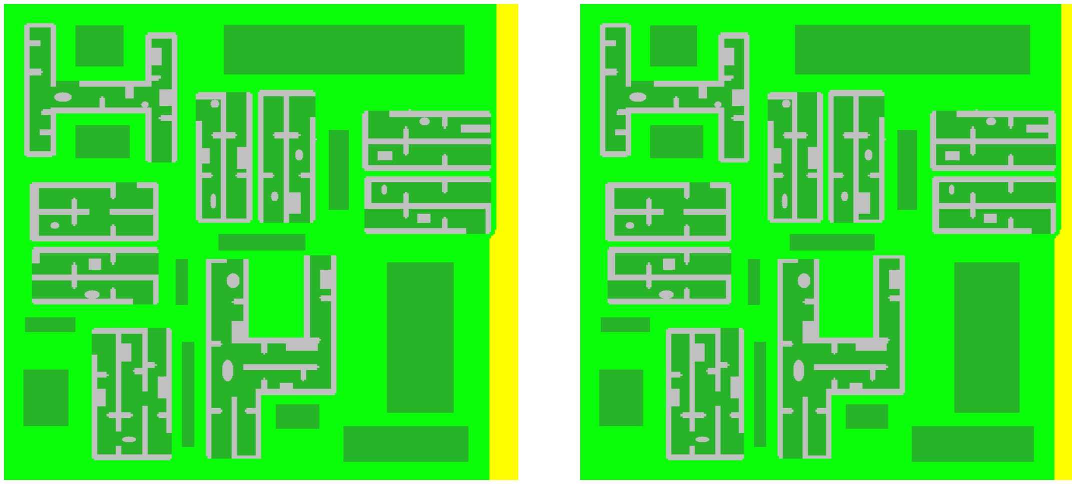


Figure 20. Two additional difficulty levels of terrain modeled in MANA

To ensure the scaling compatibility of terrain size and speed range explored, 200pixels by 200pixels is chosen as the battlefield pixel setting to represent the 50m by 50m area of operations. This works out as 1 pixel representing 0.25m, which is equivalent to the resolution of the scenario.

D. MODELING FACTORS

This section will address the factors that are modeled and investigated in the agent based model. The factors can be divided into decision and noise factors, identified to potentially have an influence or impact on the response. The difference between decision and noise factors lies in whether the user has any control over the particular factor. In this context, factors governing the characteristic of the robot and the detector are considered to be decision factors, while terrain is considered a noise factor. The movement propensity of a robot is considered a noise factor, although we have some control of it in reality. This will be discussed in detail in the movement propensity parameter sub-sections. Figure 21 is a depiction of what an agent in the simulation entails—a swarm robot integrated with the required sensor, detector and actuator to search and detect IEDs autonomously and cooperatively.

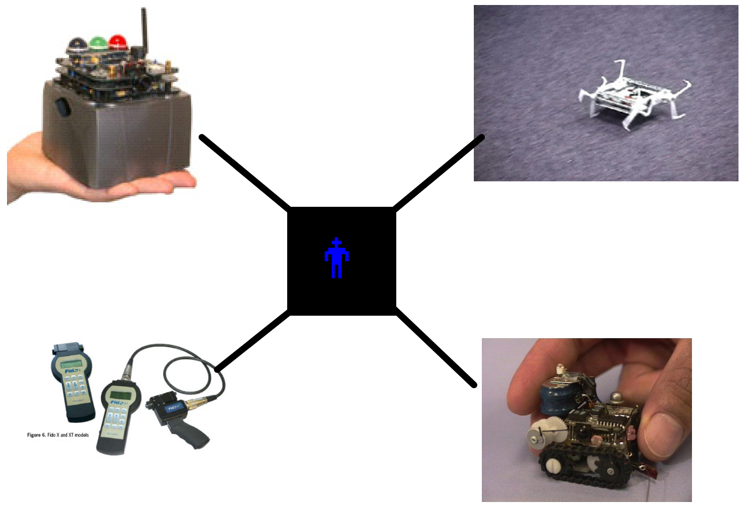


Figure 21. An agent represents a swarm robot that integrates intelligence, mobility, detection capability and miniaturized sensors and actuators. Clockwise from top left: iRobot Swarm Robot, CWRU Mini-Whegs, MIT's 1 cubic inch Fingrant mounted with 17 emitters and sensors, ICx Nomadics FIDO XT explosive detector

1. Number of Robots

This decision factor is an important element to determine prior to deploying a robotic swarm on a search and detect mission. Although these platforms are envisaged to be low cost entities so there is not much concern about their survivability and accountability, there will certainly still be a limitation on the number of robots that could be deployed. Preliminary test simulation runs show that 200 robots seem to provide more than adequate coverage to facilitate multiple detections within the 30min mission time, while a swarm with 20 or fewer robots seems to fare badly. The range of interest for the number of robots is hence decided to be 20 to 200.

2. Sensor Range (Detector Range)

The IEDs in the scenario have been modeled as stationary enemies, so that the sensor meant to detect and classify an enemy is essentially used to model the IED detector. Using the FIDO detector as a baseline, it is not clear what its operational standoff distance is, but it claims to have "demonstrated under field conditions the detection of landmines with performance approaching that of canines" (www.icxt.com). A conservative estimate will be to model the detector to have a standoff distance of at least 0.5m. The upper bound of the sensor range experimented in the simulations is chosen to be a rather optimistic 10m, but it should be noted that according to iRobot CEO Colin Angle, who recently unveiled the integration of the FIDO detector onto the Packbot (ICx Nomadics press release on 25 Sep 2006), the new FIDO could detect explosive residues from 80ft away (approx. 25m).

Unfortunately, a limitation of MANA is its inability to provide separate sensor ranges for different types of agents. In this case, setting the sensor range parameter will also imply that the sensor range for detection of fellow robots (e.g., the sensor range of its IR and acoustic sensors) will be pegged to the same value. This is a significant limitation that is discussed later. The range of 0.5m to 10m corresponds to 2 pixels to 40 pixels in the MANA scenario.

3. Speed of the Robot

Based on the current speed capability of the Mini-Whegs discussed in the previous section, the range of speed for the swarm robot modeled is 0.1m/s to 2m/s. The speed of the Mini-Whegs falls roughly in the middle of this range (http://biorobotics.cwru.edu). A general recommendation for MANA is that the speed setting of agents does not exceed 100 pixels per 100 time steps (1 pixel per time step). This speed range, which corresponds to 4 pixels per 100 time steps to 80 pixels per 100 time steps, adheres to the recommendation. This is especially important for modeling ground agents, as it ensures that situations where agents "skip" or "jump" over walls will not occur during the simulation.

4. Detector Capability

The detector capability is modeled to capture the probability of detection of the IED detector mounted on each robot. Taking reference to the FIDO detector, the sampling of its surrounding environment is done in real time. This translates to a higher probability of detection should the detector be near to the target for a longer duration. It turns out that MANA is able to model this aspect quite accurately by using the parameter of camouflage per turn of the enemy (modeled as the IED). Figure 22 illustrates the interpretation of personal concealment per turn to the probability of detection over time.

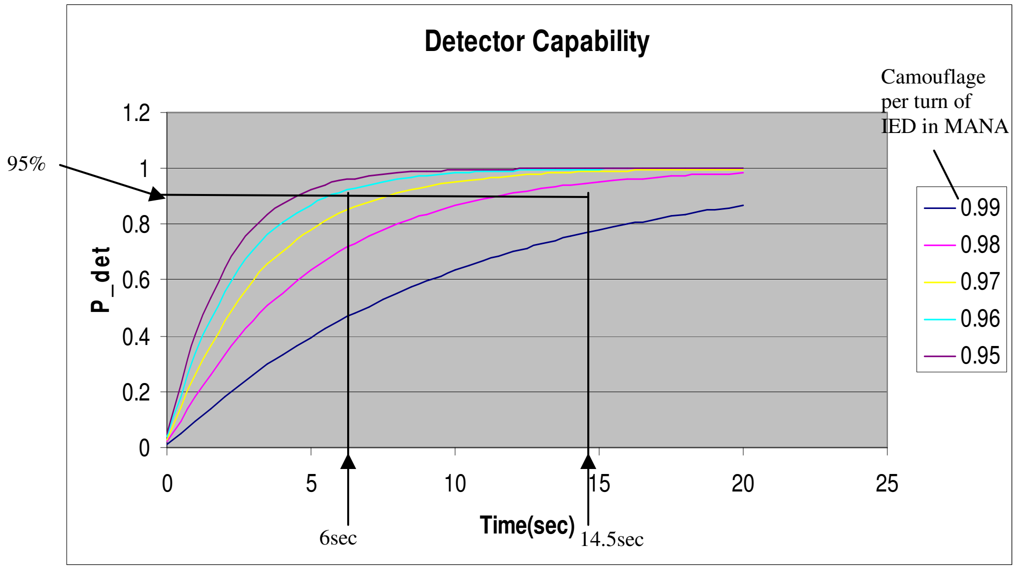


Figure 22. Modeling probability of detection over time

Each curve in Figure 22 represents a different camouflage per turn assigned to the IED. A 0.98 camouflage per turn implies that the IED has a 2% chance of being detected per time step. With the probability being compounded over time, the IED has a 95% chance of being detected if it is within the sensor range of the detector (mounted on the robot) for 145 time steps (approx 15sec). Alternatively, we can say that the detector has a 95% chance of detecting the IED if it is within range for 15sec. On the other hand, if the camouflage per turn is 0.95, this will correspond to a more capable detector. In fact, this gives the detector a 95% chance of detecting the IED if it is within range for 6sec. A range of camouflage per turn from 0.74 to 0.98 basically means that we are modeling a detector that needs a time on target (TOT) from 1 sec to 15 sec to achieve a 95% chance of detecting the IED. A conversion equation from camouflage per turn (CPT) to TOT is as follows:

$$TOT(\text{sec}) = \frac{\lg 0.05}{10 \lg CPT}.$$

5. Detector Reset Time

Another aspect that can be modeled quite accurately in MANA is the detector reset time when an IED is detected. The FIDO detector has a sensing element that has a reversible response once an actual detection of explosive materials occurs, allowing it to be reused many times. Upon detection of a target, it takes seconds for the sensing element to return to baseline for the subsequent detection. In MANA, this reset time can be modeled by incorporating a trigger state of the robotic agents; such that when a detection is made, a robot gets triggered into a state where its sensors are switched off for the duration. This is basically the trigger state time, after which the robot automatically reverts back to its original state and continues to seek out other IEDs. The range of detector reset time is varied from 0.1 sec to 10 sec (1 time step to 100 time steps)

6. Repel Uninjured Friends

As stated in the previous chapter, the attempt is made to replicate the emergent behavior of swarm robotics by applying the principles of a swarm robotic algorithm. Batalin's Molecular approach is modeled by having each agent repel from all its neighbors that it senses at any time, without any concerted effort to coordinate dispersion. There is no communication between robots and no local identities are created. However, the robots are able to differentiate between fellow robots and obstacles. Batalin's algorithm for his robots was a set of clear, prioritized rules. MANA's movement algorithm determines the best move choice at every time step based on the penalty function calculated for all possible moves, including staying put (Gill and Greiger, 2003; MANA Users Manual, 2004). The argument here is that by assigning the agents a negative movement propensity with respect to each other, it captures the rule that the robots repel each other. To quantify this parameter, e.g., selecting a value like -100, -50, -1, etc. and fixing it for a run, may not produce the desired emergent behavior that Batalin obtained from his experiments. From a series of preliminary test runs, a range of -60 to -25 is observed to produce coverage from the swarm. However, this is treated as a noise factor. We have no direct interpretation of the quantity, except that it should be a negative value.

7. Repel Cover

The parameter repel cover is varied to reflect the autonomous obstacle avoidance and navigational ability of the robot. For example, laser range finder (LRF) and ultrasonic range sensors mounted on the robot will cause the robot to steer clear of any obstacles it detects. Since only walls have the element of cover in the model, it is valid to assign the robotic agents a negative value in this aspect to represent their reaction to obstacles. In addition, it is observed from preliminary test runs that repulsion from cover produces fewer instances of agents getting stuck in corners (an inherent phenomenon in MANA under certain parameter combinations). The range that is observed to produce good coverage, in conjunction with repulsion from uninjured friends, is from -50 to -10. Again, this is treated as a noise factor because we have no direct interpretation of the quantity, except that it should be a negative value.

8. Precision Move

This parameter is a noise factor that is incorporated to model uncertainties of movements, e.g., disparities between movement choices of a robot from its intention due to random instances such as bumps in terrain, malfunctions of sensors or actuators, etc. This parameter is not expected to cause an impact on the performance of the swarm as long as it is not too high (1000 being Brownian motion) or too low (which tends to get agents stuck). From initial simulation test runs, a range of 100 to 300 produces good coverage.

In summary, the eight factors are displayed in Table 2 for the 50m by 50m scenario. It should be noted that terrain is also a noise factor but with three discrete levels.

	50m by 50m Terrain		
	Decision Factors	**Low**	**High**
1	No of Robots	20	200
2	Sensor	2 (0.5m)	40 (10m)
3	Speed of Robot	4/100 (0.1m/s)	80/100 (2m/s)
4	Detector Capability, continuous function (95% detection after x seconds)	0.74 (1sec)	0.98 (15sec)
5	Detector Reset Time	1 (0.1sec)	100 (10sec)
	Noise Factors	**Low**	**High**
1	Uninjured Friends	-60	-25
2	Cover	-50	-10
3	Precision Move	100	300

Table 2. Summary of the 8 factors and their ranges. Not listed are the 3 discrete levels of terrain

E. TARGET RANDOMIZATION AND SWARM STARTING LOCATION

In an operational environment, the location of IEDs is uncertain. We seek to capture this uncertainty in the model as well. The approach taken in the modeling is to pick 30 candidate IED locations across the terrain, including both the interiors and exteriors of the houses, and have the simulation utilize its random seed to randomly pick 10 out of the 30 possible locations at the start of every run (Figure 23). This is achieved by using a "super agent" with a universal sensor range (unchecked option for terrain affects LOS) and a "runstart" trigger state, enabling the super agent to randomly pick 20 out of the 30 IEDs and kill them off within the first five time steps of every simulation. The super agent is given 20 rounds and 100% probability of kill to ensure exactly 10 IEDs remain in the scenario after five time steps. This also requires the 30 IEDs to have 0% camouflage per turn for the first five time steps by means of the runstart trigger state. The 10 IEDs remaining will revert to the camouflage per turn as dictated in the setting for that particular run. This is an effort to model the uncertainty of IED placements in a terrain, but one can argue that it is still not truly random as the initial 30 candidate IED location are still required to be fixed and may be biased. However, this method is still a significantly better representation than fixing 10 locations right from the start and ignoring location uncertainty.

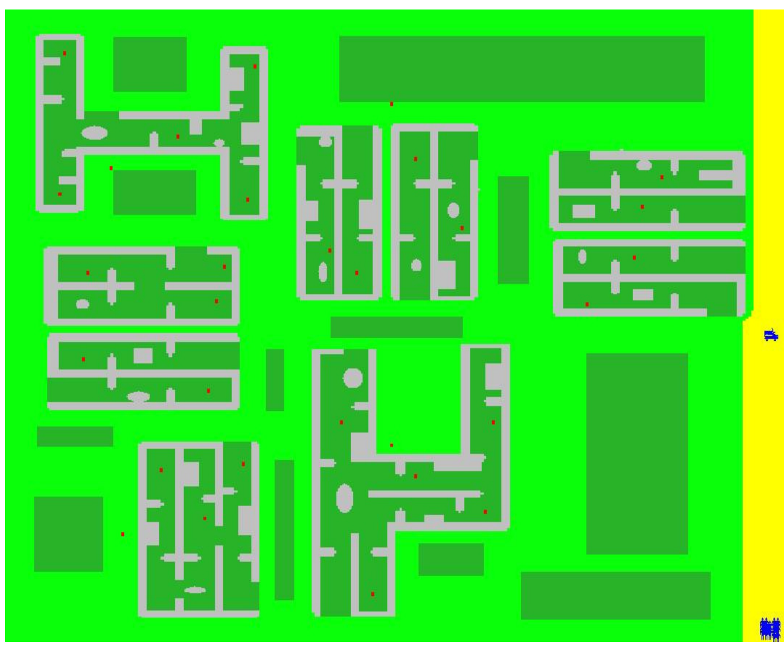


Figure 23. The 30 candidate IED location. Bottom right shows the starting location of the swarm robots. The truck icon is the super agent that kills 20 of the 30 agents at the start of each run

In addition, the swarm is set to always begin its movement from the lower right corner of the area of operations. This starting location will apply for all scenarios, though it is expected that multiple starting locations will definitely enhance the performance of the swarm. In fact, this is proposed as one of the possible factors to investigate for future work.

F. ASSUMPTIONS OF THE MODEL

This section addresses the assumptions made in the modeling of the scenario. They are listed below; some have been mentioned in the preceding sections.

1. In general, the agents follow two basic movement propensity rules; i.e., avoiding obstacles and avoiding other robots within their sensor range. The agents have no waypoints and their behavior arises purely from interactions with each other and the surroundings. There certainly exists the "inner workings" behind MANA's movement algorithm (the Stephen Algorithm was selected) and movement settings such as "Diagonal Motion Correction," "Navigate Obstacles (momentum)," and "Squad Moves Together." However, the overall movement effects on the agents are dominated by the two basic movement propensity settings. The option of "Navigate Obstacles (momentum)" also play a key role in causing the agents to move somewhat in a general direction when external influences are absent, to replicate robots moving in a straight line when its sensors do not capture anything.

2. Robots are modeled not to require GPS or positional knowledge for navigation. However they will still need this capability for transmitting the positional information when a target is detected. Whether this is achieved with GPS, or with a combination of compass/odometry and laser range finder instruments, will not be explicitly defined.

3. The swarm robots do not stray out of the area of interest. This implies that they have been programmed accordingly and have the ability to know that an option that brings them beyond the perimeter will not be considered. One possible solution is to place "virtual wall units" along the sides and corners of the area of interest, causing an invisible infrared barrier that robots will not cross, similar to the concept used by the iRobot Roomba vacuum cleaners as shown in Figure 24.

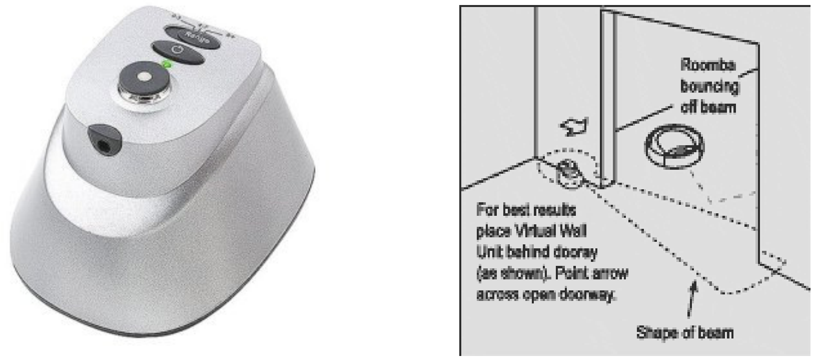


Figure 24. iRobot Virtual Wall Unit (www.irobot.com)

4. The robots have the communication means to transmit positional data back to the commander, regardless of whether they are indoors or outdoors. Conventional direct radio frequency links such as Wi-Fi is one possible way, but the presence of walls and range requirements may pose problems. Again, this will not be explicitly defined, but there has been at least one suggestion in the literature (Howard et al., 2001) that the robotic swarm itself could form a wireless communication network domain that links unreachable robots in buildings to the operator on the outside. It should be noted that the communication of detection information from the robots back to the operator will not be modeled, but is assumed to exist.

5. The IED detector on the robot operates continuously, i.e., the detector is in fulltime operation while the robot is moving, and is continuously sampling the environment for explosive content. Once a detection is made, the robot automatically transmits a message back to the operator on its position. The detector will take a finite amount of time to reset back to baseline, during which it is inactive and incapable of performing any sampling or detection. The implication of the preceding is that the longer a robot happens to be around an IED, the higher the probability of detection. This implies that there may be instances where a robot that moves too fast may miss out the detection of IEDs. Speed may impede the number of detections made by the swarm!

6. Robots are capable of overcoming basic obstacles like a single step, uneven ground, grasslands, curbs, etc. This assumption also implies that the general layout of the terrain is restricted to obstacles that these robots could overcome. Implicitly, the buildings are single-floor houses with rooms and open doors. This

assumption is made to balance the realism of robotics mobility technology attainable in the near future, and operational requirements in a real-life environment. The interiors of the houses have been modeled to incorporate an aggregated movement inhibition that is higher compared to the exteriors. This is an effort to reflect the generalization that the insides of houses have tables, chairs, appliances, hence making it more difficult to move around than exteriors, which tend to be roads, gardens, paths, etc.

7. It is assumed that the mission time is 30min (equivalent to 18,000 simulation time steps), therefore the runs will be terminated after this duration regardless of the status of the detections. This maximum allowable time reflects the fact that time constraints typically exist for clearing an area of interest. The simulation could be termed as a time-terminating one.

G. MEASURE OF EFFECTIVENESS (MOE)

The first-cut MOE that can be used is the measure of the time it takes for all 10 IEDs to be detected. However, this measure alone is not enough to give us a full picture in practice because of the following reasons.

1. A faulty detector may malfunction and transmit a "detection made" signal when there is none.

2. False alarms may occur due to the inherent probability of false alarms of any type of detectors (the probability of false alarm of the FIDO is five percent according to the FIDO XT brochure). This implies that we need more than one detection to reduce the number of false alarms.

In reality, the positional data that is transmitted back is in fact the positional data of the robot itself, as there is no way to pinpoint the location of the IED since the sampling is done on its surrounding environment. For cases where the sensor range capability of the robot is relatively large, the position reported of the robot when the detection is made may not give an adequately accurate position of the IED. One way to mitigate this is to shorten the sensor range of the detector. A better option is to increase the requirement for the number detections made, in the hope that the aggregate of all those detections made of a particular IED will give us a better representation of where its actual location is as illustrated in Figure 25). This implicitly assumes that IEDs are located relatively far apart, so that a cluster of detections in a general neighborhood can reasonably be considered to be the detections of the same IED. We set this number of detections to be three.

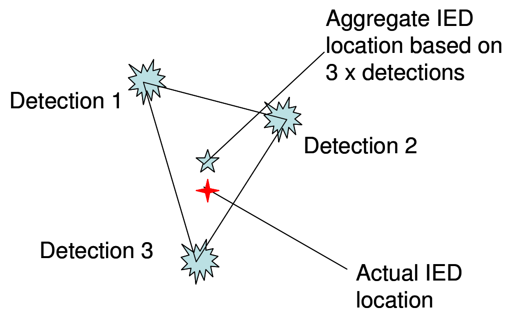


Figure 25. The aggregate of more detections will mitigate the inaccuracy of position reported

We would like to measure the time taken for all 10 IEDs to be detected at least three times. However, there is one more issue. There may be instances where a single robot with a faulty detector remains in a general vicinity, and transmits more than three detections. To prevent this from qualifying as an "official detection," the following adjustment is made.

MOE 1 = Time taken for all 10 IEDs to be detected 3 times*

***(of which the 3 detections are made by unique robots/detectors)**

This MOE is deemed to be stringent enough to provide an accurate measure of how effective the swarm is based on any set of factor settings.

In an operational scenario, the number of targets is unknown. One way of classification is to consider those with more than or equal to three detections to be "confirmed" IEDs, while those with less than three are "suspect" IEDs. With this classification logic, it is then up to the commander to decide the course of action, particularly with "suspect" IEDs, based on time constraint and overall objective (he may choose to ignore and skip "suspect" IEDs).

Alternate MOEs are needed to take into account simulation runs that did not have all 10 IEDs detected by the end of 30min. These cases will reflect a mission complete time of the 30min for MOE 1, when in fact they may take much longer. To take these cases into account, we define the two additional MOEs:

$$\textbf{MOE 2} = \begin{cases} 1 \longrightarrow mission \cdot accomplished \cdot under \cdot 30\min \\ 0 \longrightarrow otherwise \end{cases}$$

MOE 3 = Number of IEDs Detected in 30min

MOE 2 is a binary response variable that will require logistic regression modeling while MOE 3 is a continuous response variable. Both MOEs give better insights for runs that did not complete within the mission time.

H. MODELING OF VIRTUAL PHEROMONES

An additional feature is also incorporated in the model to enhance the performance of the swarm robots. The concept of using "virtual pheromones" was suggested in at least two papers on swarm robots (Wagner et al., 1999 and Van Dyke Parunak et al., 2002).

Wagner derived this concept from ants and some insects found in nature that are known to use chemicals called pheromones for communication and coordination tasks. This is briefly mentioned in Chapter II: recall that ants are able to eventually find the shortest path to the food source by using a shared memory built by depositing pheromones which evaporate over time. Wagner investigated the ability of a group of robots that are able to communicate by leaving chemical odor traces to perform the task of cleaning the floor of an unmapped building or any tasks that requires the coverage of an unknown region (seen in Figure 26). These odor traces evaporate over time and the robots are able to evaluate the intensity of the traces at every point they traverse. Hence, robots are able to compare trace levels and differentiate locations that are visited more recently. Wagner further proposed that this concept could be extended to deployment of swarm robots in hazardous environment clean-up and surveillance patrols in hostile environments. Wagner's simulations were coded in the C programming language; he mathematically spelled out the equations as part of the algorithms.

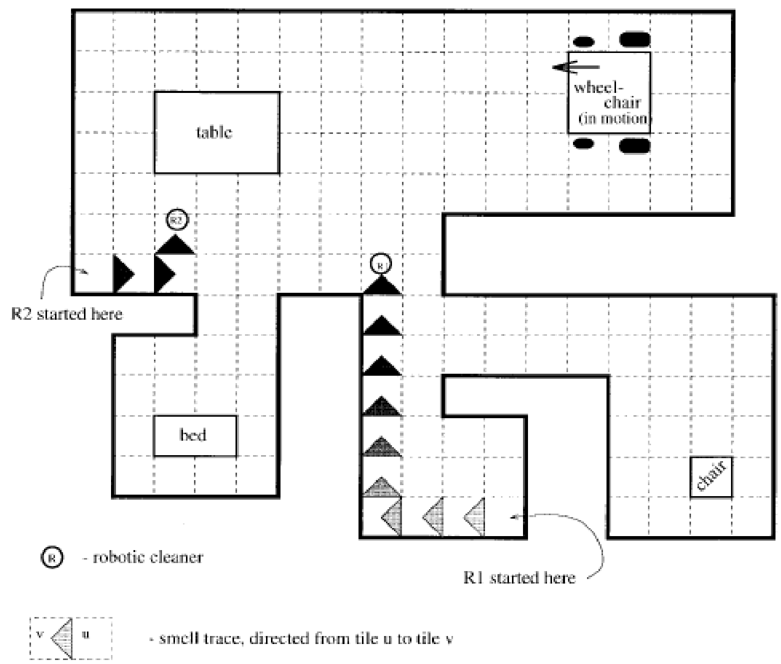


Figure 26. A pictorial illustration of cleaning robots in a system of rooms. Two cleaning robots are shown using smell traces that degrade over time (Wagner, 1999)

In this thesis, we attempt to utilize agent based simulation to explore such a concept, by setting up a shared memory that can be built and dissipated over time during the simulation, with agents reacting to in a certain manner. This memory map can be created by establishing an outbound communication link of the entire swarm (or squad) to itself, which sends information on all the positions of the robots at any time as shown in Figure 27. This memory map is manifested as the swarm's inorganic situational awareness map (Inorg SA Map). If the inorganic contact persistence of the swarm is set to a finite value, say 1000 time steps, then the old positions occupied by the robots will persist on the Inorg SA Map for 1000 time steps. This is the basis of how a pheromone shared memory map is modeled. A snapshot of this map is shown in Figure 28.

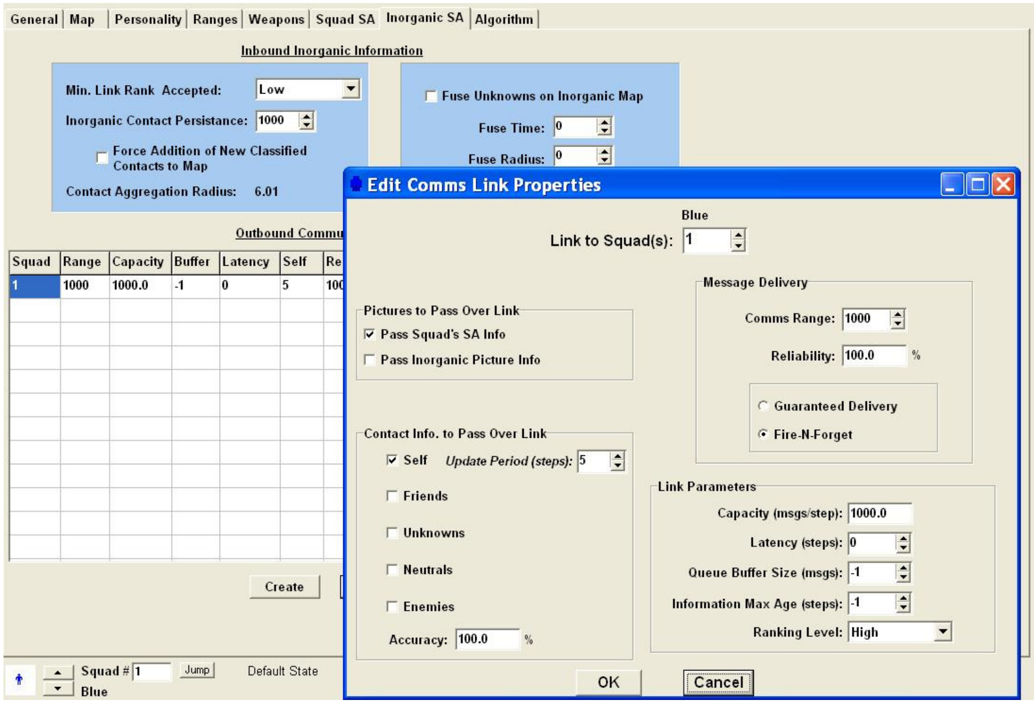


Figure 27. Comms link to itself to create pheromone shared memory map

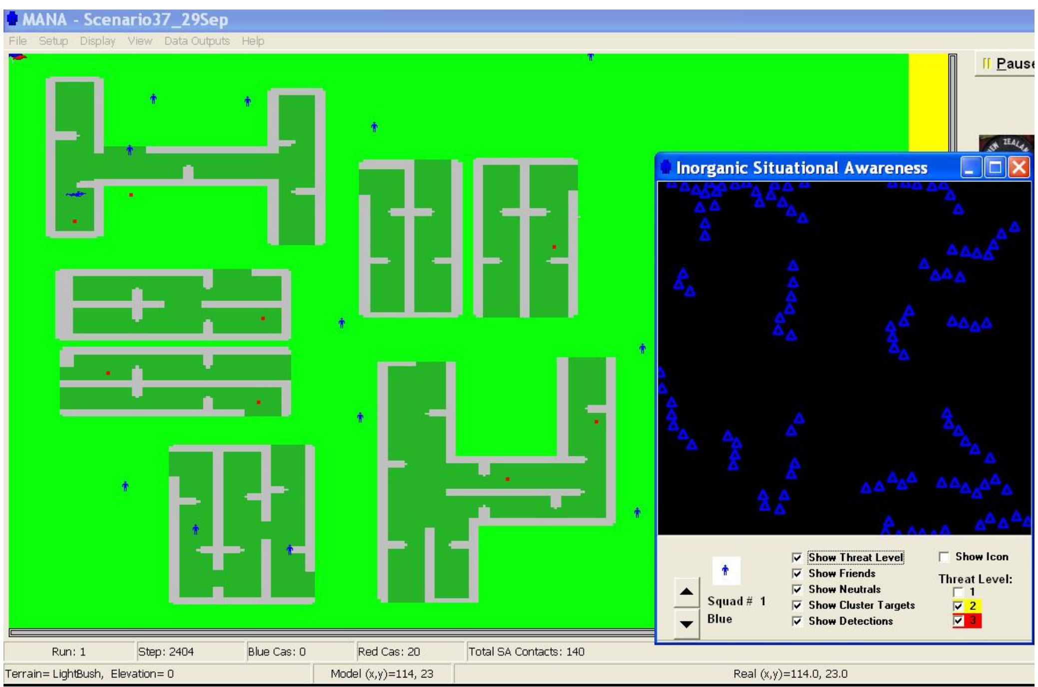


Figure 28. Snapshot of Inorganic SA Map during a simulation run

The next step will be to input a desired movement propensity of the robot to the Inorg SA. Altogether, the introduction of virtual pheromone requires additional three factors to be farmed over.

1. Repel Friends under Inorganic SA

This parameter states the movement propensity with respect to fellow robots that appear on the Inorganic SA map. In our context, this represents the movement propensity with respect to virtual pheromones. Since the objective is to enhance spread and coverage, this parameter should be negative. However, there is no direct interpretation of the magnitude of this value. We decided to peg the range to Repel Uninjured Friends in the Agent SA since we can treat repulsion from a fellow robot to be roughly the same magnitude as repulsion from a pheromone to encourage spread and coverage. The range is -60 to -25.

2. Max Inf for Friends under Inorganic SA

This represents the influence distance of the above parameter, which can be thought of as the sensor range of the pheromone detector/sniffer. We decided to peg this range to the capability of the sensor range of the robot, for lack of a better proposed range. The range is 2 to 40 pixels, or 0.5 to 10m in real world measurements.

3. Inorganic Contact Persistence for Inbound Inorganic Information

The inorganic contact persistence governs how long a contact stays on the shared memory map. In this case, this represents the persistence of the virtual pheromones once it is being deposited. We explore the range of 10sec to 100sec (100 time steps to 1000 time steps). In MANA, 1000 is the maximum inorganic contact persistence possible, which seems adequate to examine the impact of virtual pheromones that evaporate slowly.

Table 3 summarizes the ranges for the additional factors.

	Additional Factors for Pheromone Capable Robots	**Low**	**High**
1	Repel pheromones	-60	-25
2	Pheromone sensor range	2 (0.5m)	40 (10m)
3	Pheromone persistence	100 (10sec)	1000(100sec)

Table 3. Factor settings for pheromone capable robots

There are, however, a few limitations of this method of modeling the virtual pheromones:

1. A virtual pheromone disappears when its inorganic time persistence time is up. There is no gradual decrease of intensity over time and the robot does not compare the intensity of pheromones between locations. The only differentiation a robot can make is whether or not a location has a pheromone.

2. Using the Max Influence parameter to simulate the sensor range for pheromones implies that pheromones can be detected through walls. This is because this parameter in MANA does not take LOS into account. As long as it is on the Inorg SA Map and it is within Max Inf, the robot will see it. One could argue that this can lead to optimistic conclusions. If this is a concern, conclusions can still be drawn from the results of lower ranges of Max Inf, which results in fewer (if any) occurrences of pheromone detection through walls.

IV. DESIGN OF EXPERIMENT AND TREATMENT OF OUTPUT

A. DESIGN OF EXPERIMENT

The previous chapter discusses the factors being modeled in the simulation. The number of factors that are of interest totals up to eight, excluding three levels of terrain. A full factorial design that is able to incorporate all possible combinations of the high and low settings of each factor will take 256 (2^8) design points or runs. This approach is straightforward and simple, but inadequate. First of all, it fails to provide insights of the response in regions within the range of each factor. In addition, to investigate the impact of stochasticity, more than one replication is required for each design point.

To increase the efficiency of the experiment, the Nearly Orthogonal Latin Hypercube (NOLH) is utilized (Cioppa, 2002; Cioppa and Lucas, 2006). Using the NOLH will provide a design of experiment that can give us the fidelity to get insights on responses from the entire range of each factor. In fact, the NOLH requires fewer design points than a full factorial design of high and low settings, because of its high efficiency and space filling property. With eight continuous factors, only 33 design points are required to construct the NOLH, using the Microsoft Excel spreadsheet created by Professor Susan M. Sanchez (Sanchez, 2005). To investigate stochastic responses from uncertain and probabilistic events such as movements and detections, 100 replications are performed for each design point, giving a total of 3,300 runs. Figure 29 shows the NOLH for the 8-factor design, while Figure 30 shows the space filling property of the NOLH design. It should be emphasized here that the savings that the NOLH design bring are tremendous. To achieve the same extent of fidelity using a full factorial design, it will easily take much more than 25,600 runs (assuming 100 replications), versus the 3,300 runs that is performed here.

20	**2**	**4**	**74**	**1**	**-60**	**-50**	**100**
200	**40**	**80**	**98**	**100**	**-25**	**-10**	**300**
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**
No of agents	**Sensor range**	**Speed of robot**	**Det capab**	**Trigger duration**	**Repel friends**	**Repel cover**	**Precision movt**
200	6	37	79	88	-38	-23	194
183	40	14	83	47	-53	-20	163
178	19	73	78	4	-39	-21	106
121	35	80	84	94	-55	-18	113
189	3	40	79	69	-35	-34	213
194	38	28	81	44	-52	-44	275
144	20	78	80	1	-37	-35	281
116	28	75	82	91	-51	-41	300
138	12	21	87	72	-49	-50	138
155	27	25	91	23	-41	-46	175
149	10	61	97	35	-58	-45	131
161	29	54	97	75	-26	-31	181
127	8	18	88	60	-56	-11	256
172	25	33	95	16	-40	-13	244
133	9	68	96	38	-60	-24	250
166	26	49	98	81	-28	-28	231
110	21	42	86	51	-43	-30	200
20	36	47	94	13	-47	-38	206
37	2	71	89	54	-32	-40	238
43	23	11	94	97	-46	-39	294
99	7	4	88	7	-30	-43	288
31	39	44	93	32	-50	-26	188
26	4	56	91	57	-33	-16	125
76	22	6	92	100	-48	-25	119
104	14	9	90	10	-34	-19	100
82	31	63	85	29	-36	-10	263
65	15	59	82	78	-44	-14	225
71	32	23	75	66	-27	-15	269
59	13	30	76	26	-59	-29	219
93	34	66	85	41	-29	-49	144
48	17	52	77	85	-45	-48	156
88	33	16	76	63	-25	-36	150
54	16	35	74	20	-57	-33	169

Figure 29. The 8-factor NOLH DOE is used that only requires only 33 design points

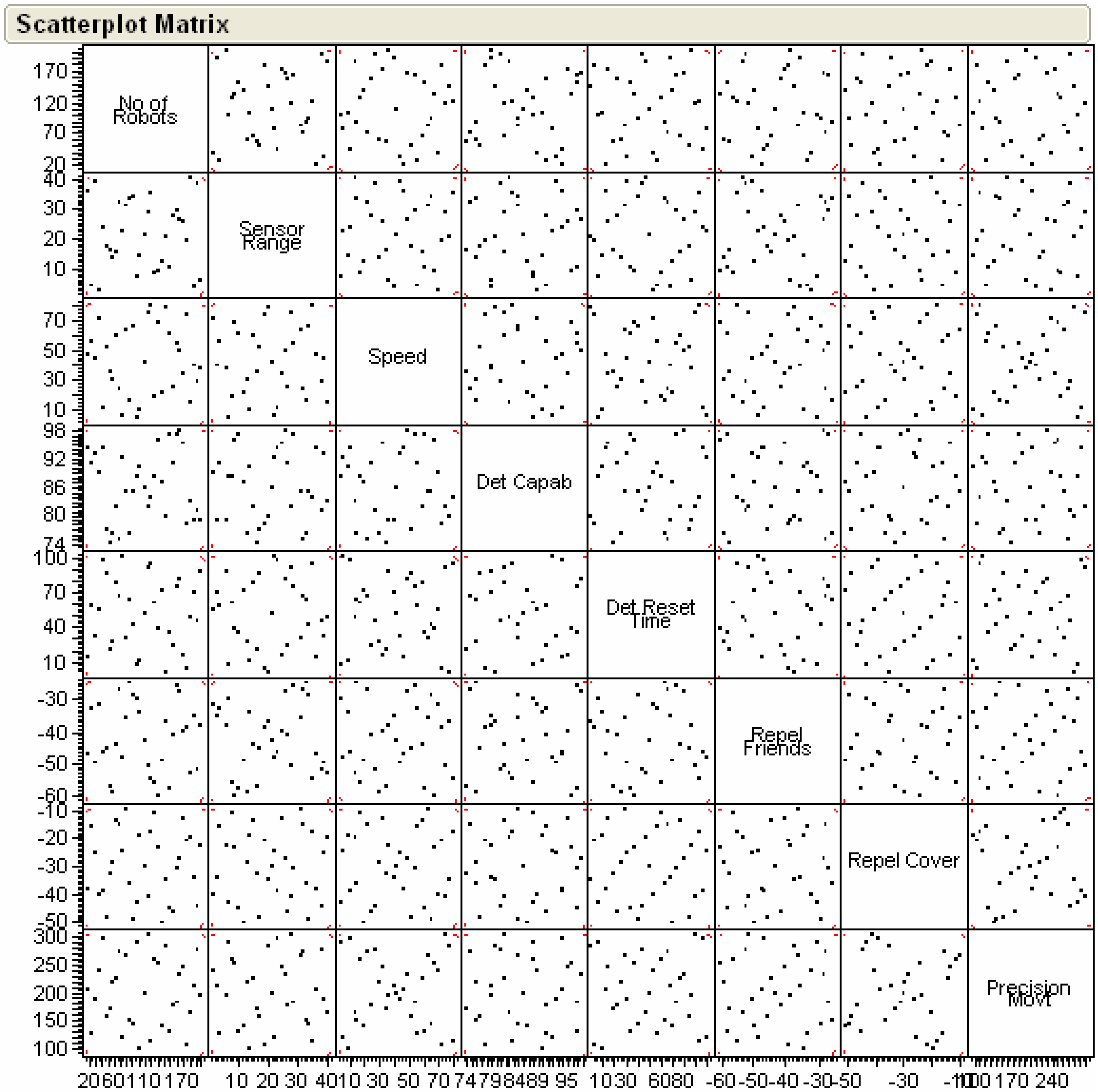


Figure 30. Space-filling property of the 8-factor NOLH

There is still the need to integrate the three levels of terrain in the design of experiment (Figure 31). A simple method to do this is to use a crossed design, where the three different terrains are crossed with the 8-factor NOLH design. An alternative way is to input the three discrete levels (0, 1 and 2) numerically into the NOLH, which then becomes a 9-factor NOLH design. The former method is chosen for ease of interpretation because terrain is a qualitative factor with only three levels. The DOE thus requires the 3,300 runs to be performed for all three terrains, resulting in 9,900 runs that need to be executed.

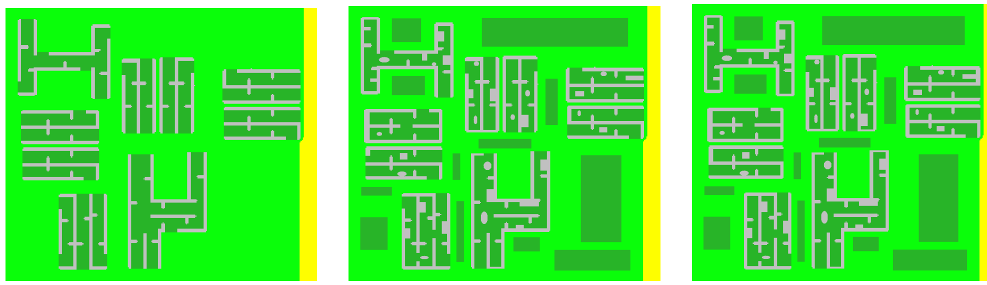


Figure 31. Terrain settings 0, 1 and 2 from left to right

A separate DOE is required to investigate the pheromone capable robots, which requires three additional quantitative factors to be varied. It turns out that an 8-factor NOLH and an 11-factor NOLH use the same number of design points. The intuitive way is to add these three factors into the original NOLH, which becomes an 11-column matrix with 33 design points. It should be noted that the settings of the initial 8-factors in the 8-factor NOLH and in the 11-factor NOLH are identical since the columns are not swapped. One could decide to swap the columns of the initial 8-factors in the 11-factor NOLH to get different settings but this approach is not taken as there is no added significant benefit to do so[5]. This DOE requires another 9,900 runs to be performed, taking into account the 100 replications and the three levels of terrain. The 11-factor NOLH is shown in Figure 32 and the space filling property of the design is shown in Figure 33.

[5] Swapping columns to produce different design points will give more fidelity to the insights on quadratic behavior and interactions

20	**2**	**4**	**74**	**1**	**-60**	**-50**	**100**	**-60**	**2**	**100**
200	**40**	**80**	**98**	**100**	**-25**	**-10**	**300**	**-25**	**40**	**1000**
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
No of agents	**Sensor range**	**Speed of robot**	**Prob Det**	**Trigger duration**	**Repel friends**	**Repel cover**	**Precision movt**	**Repel pheromones**	**Pheromone sensor range**	**Pheromone persistence**
200	6	37	79	88	-38	-23	194	-25	28	634
183	40	14	83	47	-53	-20	163	-28	19	859
178	19	73	78	4	-39	-21	106	-49	39	438
121	35	80	84	94	-55	-18	113	-45	6	297
189	3	40	79	69	-35	-34	213	-58	7	719
194	38	28	81	44	-52	-44	275	-59	22	775
144	20	78	80	1	-37	-35	281	-29	2	353
116	28	75	82	91	-51	-41	300	-41	38	269
138	12	21	87	72	-49	-50	138	-39	25	156
155	27	25	91	23	-41	-46	175	-30	9	128
149	10	61	97	35	-58	-45	131	-47	26	691
161	29	54	97	75	-26	-31	181	-52	8	606
127	8	18	88	60	-56	-11	256	-48	13	100
172	25	33	95	16	-40	-13	244	-51	32	213
133	9	68	96	38	-60	-24	250	-35	15	916
166	26	49	98	81	-28	-28	231	-32	31	578
110	21	42	86	51	-43	-30	200	-43	21	550
20	36	47	94	13	-47	-38	206	-60	14	466
37	2	71	89	54	-32	-40	238	-57	23	241
43	23	11	94	97	-46	-39	294	-36	3	663
99	7	4	88	7	-30	-43	288	-40	36	803
31	39	44	93	32	-50	-26	188	-27	35	381
26	4	56	91	57	-33	-16	125	-26	20	325
76	22	6	92	100	-48	-25	119	-56	40	747
104	14	9	90	10	-34	-19	100	-44	4	831
82	31	63	85	29	-36	-10	263	-46	17	944
65	15	59	82	78	-44	-14	225	-55	33	972
71	32	23	75	66	-27	-15	269	-38	16	409
59	13	30	76	26	-59	-29	219	-33	34	494
93	34	66	85	41	-29	-49	144	-37	29	1000
48	17	52	77	85	-45	-48	156	-34	10	888
88	33	16	76	63	-25	-36	150	-50	27	184
54	16	35	74	20	-57	-33	169	-53	12	522

Figure 32. 11-factor NOLH design to investigate pheromone capable robots

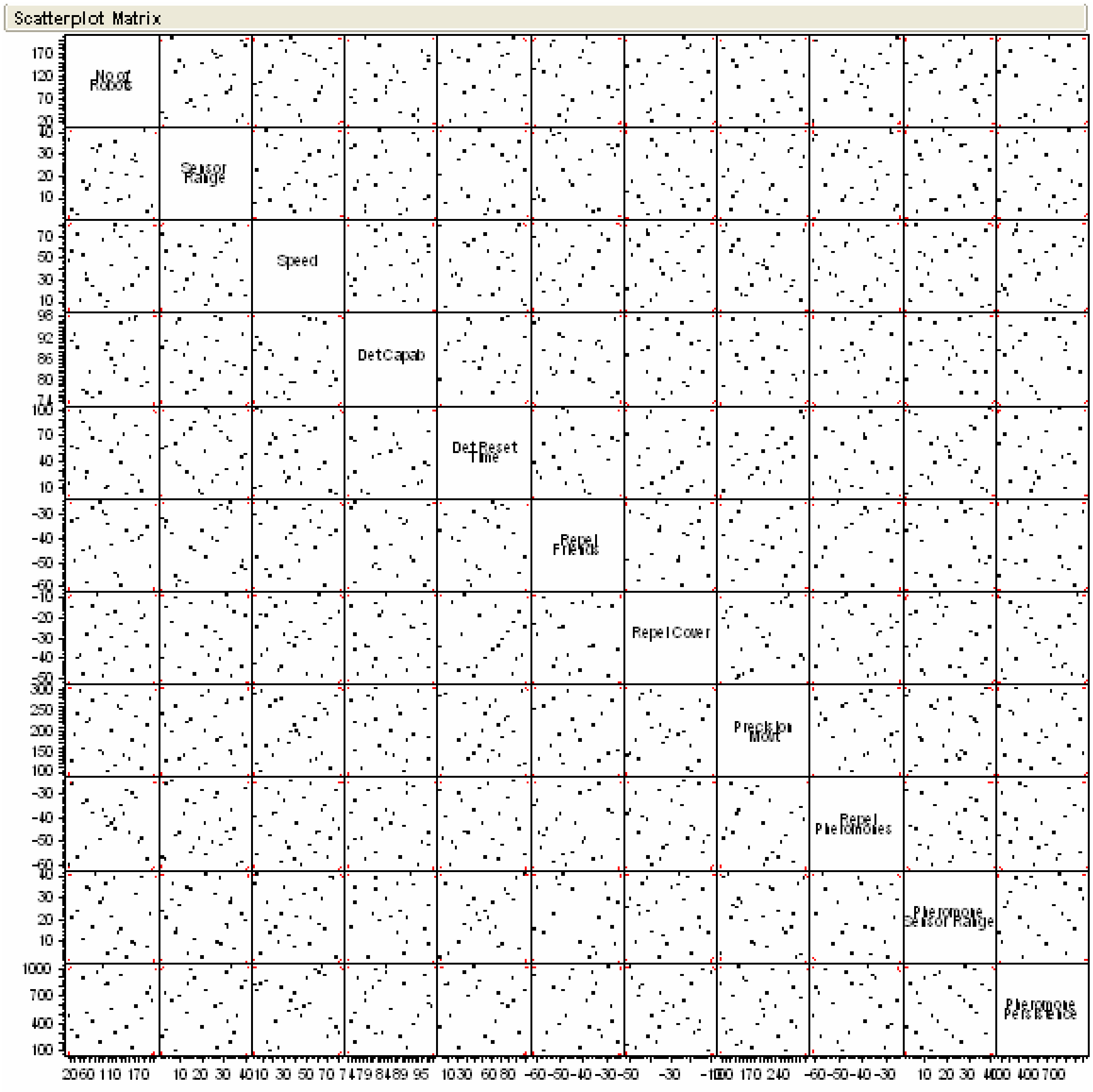


Figure 33. Space-filling property of the 11-factor NOLH

B. DATA FARMING

Overall, 19,800 runs are executed and the outputs of the detections from every run are recorded and generated. On average, each run of the scenario with non-pheromone robots takes approximately 15min to complete on a personal computer. As expected, the runs with design points that have a large number of robots take longer than average. For the scenario with pheromone robots, each run takes a significantly longer time than the previous scenario. This is due to the modeling of virtual pheromones, which makes use of a communication link that updates the positions of all robots at every time step. This requires a considerably larger computational effort at every time step and hence slows down the simulation. Such a run can easily take up to more than an hour on a personal computer.

The computational power that is required to data farm over the DOE within a reasonable time is provided by the MHPCC. There is also the requirement to incorporate the NOLH design into XML-format as part of the XML code of the relevant MANA scenario. This is done with the Tiller, which is a tool provided by Referentia Systems that sets up a folder with the basecase and study file with the corresponding terrain files. The Tiller converts the NOLH design into XML format as part of the study file and constructs the replications setup as defined by the user. This folder is then zipped and forwarded as a job submission to MHPCC, where the simulations are run in batch automatically. At the end of each run, the output files are generated and transferred to a downloadable site accessed by the user. The 9,900 runs for the scenario with non-pheromone robots take about a full day to run. The other 9,900 runs for the scenario with pheromone robots take about four to five days.

C. TREATMENT OF OUTPUT

The information that is required from the output of each run includes the detections made in the entire history of each simulation. It turns out that one of the output file option provided by MANA, i.e., "Record Multiple-Contact Detections," records all detections that are made in chronological order. This is shown in Figure 34.

# MANA Multi-Contact Detection Results File
# Version: 3.2.1
# Machine Name: C21N19
# Run Started at 9 1:20:20 PM
RandSeed= 123

Step	Squad of Detector	Squad of Classified Agt	Detector Agent	Classified Agent	x	y	Range	Detector Deadtime
1	2	1	230	212	60	142	141	0
1	2	1	230	204	20	81	176	0
1	2	1	230	224	122	159	93	0
1	2	1	230	228	28	164	178	0
1	2	1	230	214	90	83	107	0
1	2	1	230	219	160	78	42	0
1	2	1	230	223	105	147	101	0
1	2	1	230	225	122	128	77	0
1	2	1	230	218	162	62	52	0
1	2	1	230	213	81	76	117	0
1	2	1	230	226	98	135	102	0
1	2	1	230	210	41	146	160	0
1	2	1	230	229	13	13	203	0
1	2	1	230	202	61	18	158	0
2	2	1	230	216	115	68	87	0
2	2	1	230	205	96	31	122	0
2	2	1	230	209	50	119	146	0
2	2	1	230	208	20	109	175	0
2	2	1	230	201	41	37	167	0
2	2	1	230	207	55	82	141	0
460	0	1	175	220	150	91	6	0
468	0	1	192	220	150	91	6	0
497	0	1	48	220	150	91	5	0
527	0	1	94	216	115	68	5	0
550	0	1	198	220	150	91	6	0
595	0	1	141	220	150	91	5	0
607	0	1	7	220	150	91	5	0
638	0	1	198	220	150	91	6	0
646	0	1	135	217	167	51	5	0
670	0	1	129	220	150	91	5	0
701	0	1	31	217	167	51	6	0
742	0	1	135	217	167	51	4	0

Figure 34. A truncated sample detection output file generated at the end of a run

Unfortunately, output files in MANA are not easy to manipulate to output exactly what the user requires. In the above truncated sample output file, it can be seen that some work still needs to be done in order to extract the MOEs that are of interest. For example, time step one and two register detections that are an artifact of modeling target randomization. The detections are made on the first 30 IEDs by the super agent, which goes on to eliminate 20 of them. The first detection by a swarm robot does not happen until time step 460, therefore the detections made before this time step have to be discounted.

The next issue involves filtering out the unique detections. It can be seen from the sample output file that IED 220 (Classified Agent 220) gets uniquely detected by three different swarm robots by time step 497; subsequent detections of this agent are of no interest. As for IED 217, it is detected three times by time step 742, however two of the detections are not unique (it is detected by Robot 135 twice). Thus, the detection at time step 742 needs to be discounted, and this IED should not be considered as detected until the third unique detection occurs later on. It is possible for one to do this filtering process manually to track the time taken for all 10 IEDs to be uniquely detected more than three times, but it would be an arduous task (prone to human error) to manually process all 19,800 output files. It is clear that a script needs to be written to automate the process of reading in all the output files and siphon out the MOE that is needed.

The required script file is written in Ruby with much assistance from Professor Paul Sanchez. The Ruby script is able to filter out all detections made by the super agent that occur in the first five time steps (refer to section on Target Randomization), and then track all detections made on each of the 10 remaining IEDs. Repeated detections by the same robot on a particular IED are filtered, and the time is recorded when three unique detections are reached for that IED. The core of the script lies in the usage of the "hash table," which is able to store all robot ID numbers that detect an IED, for each IED. When the hash table fills up to three, the time step is noted and is classified as a confirmed detection of that IED. The script also reads in the corresponding design point, producing an output file (in text format) that matches the design point with the corresponding confirmed detection times for each IED. This text file can be read directly into the statistical package, JMP. For each run, the maximum of the 10 confirmed detection times is the time taken to accomplish mission (MOE 1). For runs that do not complete within the 30min mission time, their MOE 1 values are assigned 30min automatically. Such runs will also record less than 10 confirmed detection times, and are assigned a zero for mission accomplishment or one otherwise (MOE 2). The number of confirmed detection times recorded indicates the number of IEDs detected (confirmed detection), which is MOE 3. The Ruby script is attached in Appendix A. All the MOE compilations mentioned are done manually by inserting the formulae in JMP.

It turns out that a batch file is also essential in order for the Ruby script to read in the output files in order to match the design points correctly. This problem arises because of the nature of how scripts recognize sequences of numerical characters. Initially, output files are read in the order of 0, 1, 10, 11, ..., 19, 2, 20, 21, …, 29, 3, 30, 31, 32, when the intended order is 0, 1, 2, …, 9, 10, 11, …, 19, 20, …, 29, 30, 31, 32.

This problem can be resolved either by manipulating the Ruby script, or by renaming all the output files into the form of 000, 001, 002, …, 032, so that the intended sequence is read in correctly by the script. The latter option is chosen in view of the simplicity in constructing a batch file that automates the process of renaming multiple files. A truncated portion of this batch file is extracted and depicted in Figure 35.

```
copy    multi_detect.0.0.csv    multidetect.000.000.csv
copy    multi_detect.0.1.csv    multidetect.000.001.csv
copy    multi_detect.0.2.csv    multidetect.000.002.csv
copy    multi_detect.0.3.csv    multidetect.000.003.csv
copy    multi_detect.0.4.csv    multidetect.000.004.csv
copy    multi_detect.0.5.csv    multidetect.000.005.csv
copy    multi_detect.0.6.csv    multidetect.000.006.csv
.
.
.
```

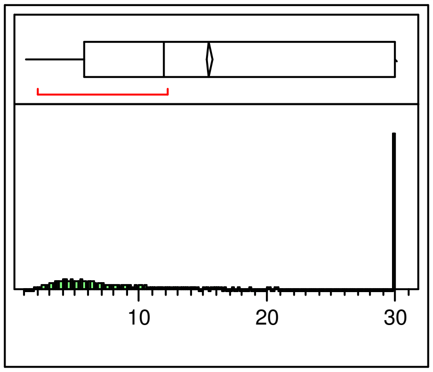
Figure 35. Batch file to rename all output files

V. RESULTS AND ANALYSIS

A. NON-PHEROMONE ROBOTS

1. Overview of MOEs

Time to Accomp Mission (min)



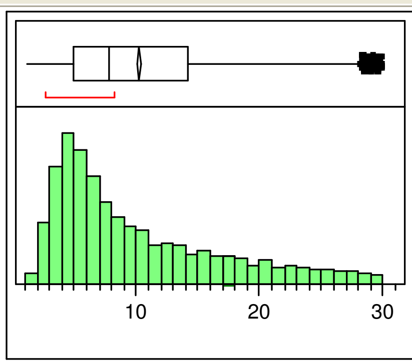
Quantiles

100.0%	maximum	30.000
99.5%		30.000
97.5%		30.000
90.0%		30.000
75.0%	quartile	30.000
50.0%	median	11.876
25.0%	quartile	5.692
10.0%		3.894
2.5%		2.621
0.5%		1.906
0.0%	minimum	1.128

Figure 36. Distribution of MOE 1

Overall, not all runs registered a mission accomplishment. Some runs are terminated at the end of the 30min pre-defined mission time with less than 10 IEDs detected[6]. The distribution in Figure 36 shows a spike at 30min, which represents all the runs with mission failures.

Time to Accomp Mission (min)



Quantiles

100.0%	maximum	29.990
99.5%		29.349
97.5%		27.256
90.0%		21.512
75.0%	quartile	14.216
50.0%	median	7.746
25.0%	quartile	4.872
10.0%		3.511
2.5%		2.446
0.5%		1.829
0.0%	minimum	1.128

Moments

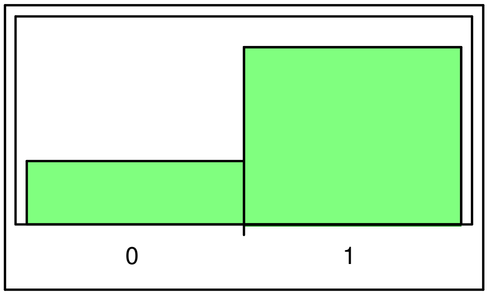
Mean	10.261008
Std Dev	6.9507209
Std Err Mean	0.0813298
upper 95% Mean	10.420438
lower 95% Mean	10.101579
N	7304

Figure 37. Distribution of MOE 1 (conditioned on mission accomplishment)

Out of 9,900 runs, 7,304 runs have accomplished missions, with the distribution of the time taken as shown in Figure 37. Essentially, this is a distribution of MOE 1, conditioned on mission accomplishment. The mean turns out to be about 10.26min with standard deviation of 6.95min, with the median at a more optimistic 7.75min.

[6] For the analysis, "detected" implies the "at least 3 unique detections" status.

Mission Accomplished



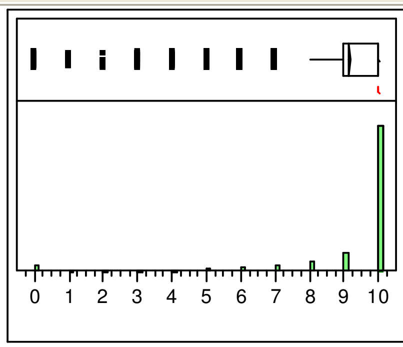
Frequencies

Level	Count	Prob
0	2596	0.26222
1	7304	0.73778
Total	9900	1.00000
N Missing	0	
2 Levels		

Figure 38. Distribution of MOE 2

Figure 38 summarizes the proportion of mission accomplishment; an overview of MOE 2. The proportion of mission accomplishment over for the non-pheromone robots is 0.738.

No of IEDs Detected



Quantiles

100.0%	maximum	10.000
99.5%		10.000
97.5%		10.000
90.0%		10.000
75.0%	quartile	10.000
50.0%	median	10.000
25.0%	quartile	9.000
10.0%		7.000
2.5%		0.000
0.5%		0.000
0.0%	minimum	0.000

Moments

Mean	9.1525253
Std Dev	2.0399507
Std Err Mean	0.0205023
upper 95% Mean	9.1927139
lower 95% Mean	9.1123366
N	9900

Figure 39. Distribution of MOE 3

Figure 39 summarizes the response of MOE 3. The mean of MOE 3 turns out to be approximately 9.15 IEDs with standard deviation of 2.04. It is observed that there are some runs which did not register a single IED detection. A subset of the data is created to investigate the cause.

2. Analysis of Ineffective Robotic Swarms

A "first-cut" subset of the data is extracted for the more "drastic" mission failures (with less than half the IEDs detected), which narrows the data down to 570 observations. Histograms of the factor levels corresponding to these drastic mission failures appear in Figure 40. There are no obvious trends except that the swarm is very ineffective in detecting IEDs whenever the speed is low (below 0.3m/s).

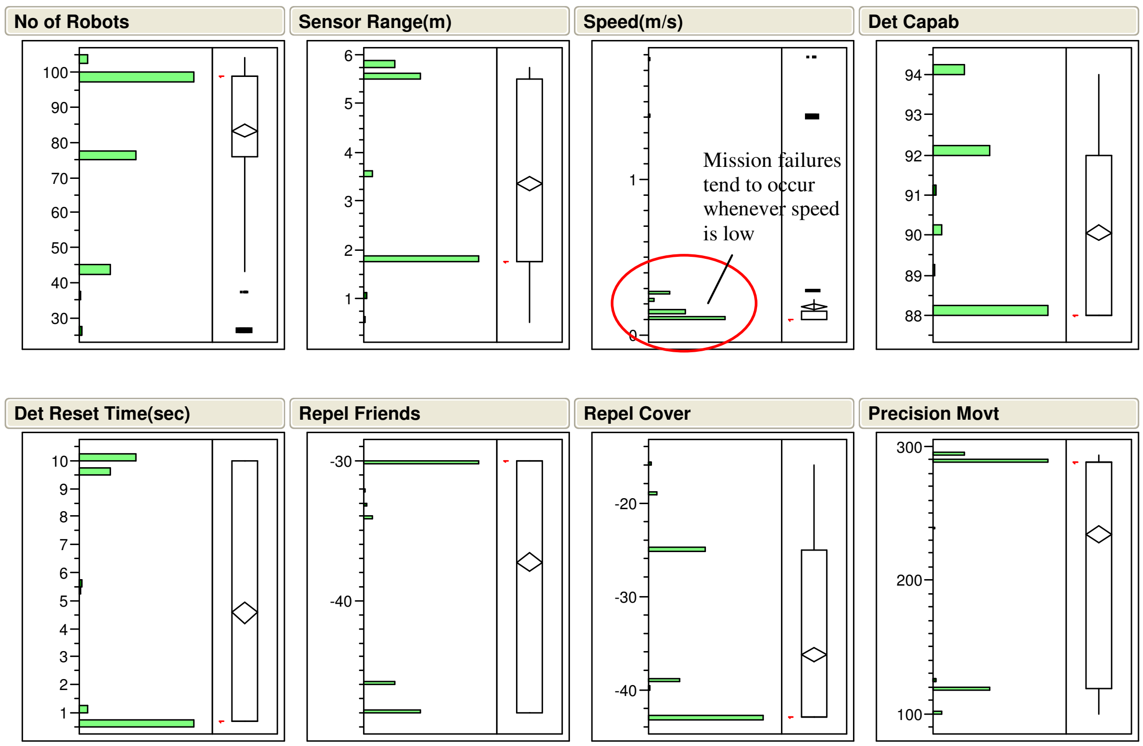


Figure 40. Subset of data with <=5 IEDs detected

The occurrences of the number of IEDs detected is monotonically decreasing as the number of IEDs detected decreases, with the exception of zero where there is a large number of occurrences as shown in Figure 41. This implies a behavior, sometimes called "falling off the cliff," where a particular setting of a parameter causes a drastic degradation in performance.

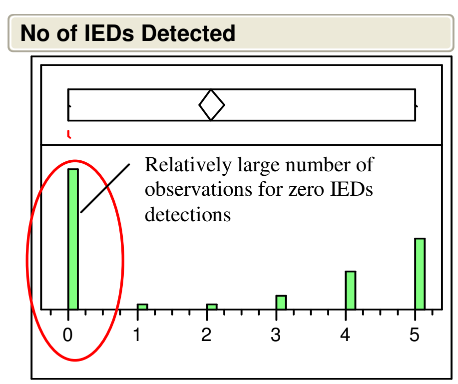


Figure 41. Distribution of MOE 3 with 5 IEDs detected or less

A further subset is extracted for runs without a single IED detection, which leaves 288 observations. It is observed that they belong to the same design point where speed is at a minimum setting of 0.1m/s, over the three types of terrain. This is an indication that a swarm with speed setting of 0.1m/s is most likely unable to even detect a single IED, regardless of other factor settings.

Figure 42. Snapshot of end of simulation for design point which failed to acquire a single IED detection

Verification of this design point is done in MANA by running and watching the simulation, input with the corresponding parameters. It is observed that the swarm barely makes it to the center of the area of operations by the end of the 30min and has even more trouble moving within the houses because of its speed. One replication of this is shown in Figure 42. It is also noted that the swarm has trouble spreading out, due to a relatively short sensor range (1.75m) for this particular design point. More insights can be obtained using regression fits and trees, to be discussed in analyses later in this chapter.

3. Logistic Regression of Mission Accomplishment (MOE 2)

A stepwise logistic regression fit of the response mission accomplishment (MOE 2) is performed on the data using all factors, shown in Appendix B. This gives us an R-square of 0.662 over the 9,900 observations. In the previous chapter, repel cover, repel friends, precision movement and terrain were described as noise factors that are difficult to determine quantitatively. What is known is that repel cover and repel friends are negative, based on the principles of the swarm algorithm being incorporated. Precision movement has no direct interpretation in the physical sense, except that it is incorporated to introduce noise to the movement selection process of the agents and to prevent agents from getting stuck in the simulation. As for terrain, it is observed that it has an impact on the MOE but there is no control over this factor in practice.

Whole Model Test

Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	3550.7055	19	7101.411	0.0000
Full	2145.4425			
Reduced	5696.1480			

RSquare (U) 0.6234

Observations (or Sum Wgts) 9900

Converged by Gradient

R-square is 0.623, degraded from 0.662 from the full model with all factors

Effect Wald Tests

Source	Nparm	DF	Wald ChiSquare	Prob>ChiSq
No of Robots	1	1	531.756014	0.0000
Sensor Range(m)	1	1	253.643246	0.0000
Speed(m/s)	1	1	377.435057	0.0000
Det Capab	1	1	32.5241107	0.0000
Det Reset Time(sec)	1	1	1.3537723	0.2446
No of Robots*Sensor Range(m)	1	1	11.7558749	0.0006
No of Robots*Speed(m/s)	1	1	71.477179	0.0000
No of Robots*Det Capab	1	1	44.5368248	0.0000
No of Robots*Det Reset Time(sec)	1	1	26.2626556	0.0000
Sensor Range(m)*Speed(m/s)	1	1	35.783768	0.0000
Sensor Range(m)*Det Capab	1	1	62.3701127	0.0000
Sensor Range(m)*Det Reset Time(sec)	1	1	76.2448241	0.0000
Speed(m/s)*Det Capab	1	1	17.9159998	0.0000
Speed(m/s)*Det Reset Time(sec)	1	1	51.6682806	0.0000
Det Capab*Det Reset Time(sec)	1	1	6.3114926	0.0120
No of Robots*No of Robots	1	1	6.03662282	0.0140
Sensor Range(m)*Sensor Range(m)	1	1	24.0528627	0.0000
Det Capab*Det Capab	1	1	28.6551384	0.0000
Det Reset Time(sec)*Det Reset Time(sec)	1	1	6.03785929	0.0140

Prediction Profiler

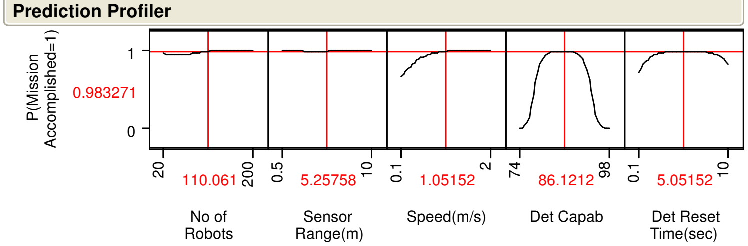


Figure 43. Logistic regression of MOE 2 without noise factors

It is desired to leave these factors out of the regression fit as they have no direct correspondence to a quantity in the real world and hence are unable to provide much practical guidance to someone seeking to plan an IED clearing mission. A new stepwise logistic regression of MOE 2 is performed with just the decision factors as shown in Figure 43. Fortunately, the regression fit does not degrade R-square much (down to 0.623.

From the logistic model, the main effects for number of robots, sensor range and speed are the most influential. Det capab and Det Reset Time (DRT) appear as main effects mainly due to their interactions with other factors.

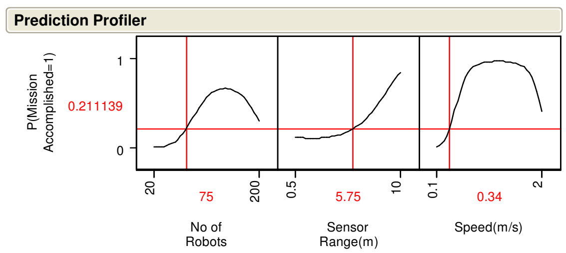
Significant interaction terms include number of robots*speed, number of robots*det capab, sensor range*det capab, sensor range*DRT and speed*DRT. Quadratic terms that turn out to explain a significant part of the model variability include sensor range and det capab.

Whole Model Test

Model	-LogLikelihoc	DF	ChiSquar	Prob>ChiS
Difference	3433.5753	9	6867.151	0.000(
Full	2262.5727			
Reduced	5696.1480			

RSquare (U) 0.6028
Observations (or Sum Wgts) 9900
Converged by Gradient

Prediction Profiler



Effect Wald Tests

Source	Nparm	DF	Wald ChiSquare	Prob>ChiSq
No of Robots	1	1	894.658544	0.0000
Sensor Range(m)	1	1	235.172212	0.0000
Speed(m/s)	1	1	437.751177	0.0000
No of Robots*Sensor Range(m)	1	1	25.4245674	0.0000
No of Robots*Speed(m/s)	1	1	47.1897317	0.0000
Sensor Range(m)*Speed(m/s)	1	1	3.34495714	0.0674
No of Robots*No of Robots	1	1	224.746164	0.0000
Sensor Range(m)*Sensor Range(m)	1	1	45.4532445	0.0000
Speed(m/s)*Speed(m/s)	1	1	567.713281	0.0000

Figure 44. Logistic regression of MOE 2 with only Number of Robots, Speed and Sensor Range

An interesting follow up to the regression of MOE 2 is to only include the three main factors that are deduced previously as shown in Figure 44. The R-square degrades slightly to 0.603 and all interactions and quadratic terms become significant. Although this model is **not** preferred over the previous one with all decision factors included, it is interesting to note that the number of robots comes up significant as a quadratic term. This is counter-intuitive at first glance, but further observations of simulations revealed that having an extremely high robot density within an area can be counter-productive. This is because as the area gets more crowded with robots, the repulsions from one another hinder continuous movement from the robot's point of view, such that it makes it more difficult for IEDs to be detected by different robots. For instance, a robot moving towards a building can be repelled from entering if there already is another robot roaming in it. Further research will need to be done on this hypothesis of optimal swarm robot density. More insights on quadratic and interaction terms will be discussed in subsequent sections.

4. Regression Tree of MOE 2

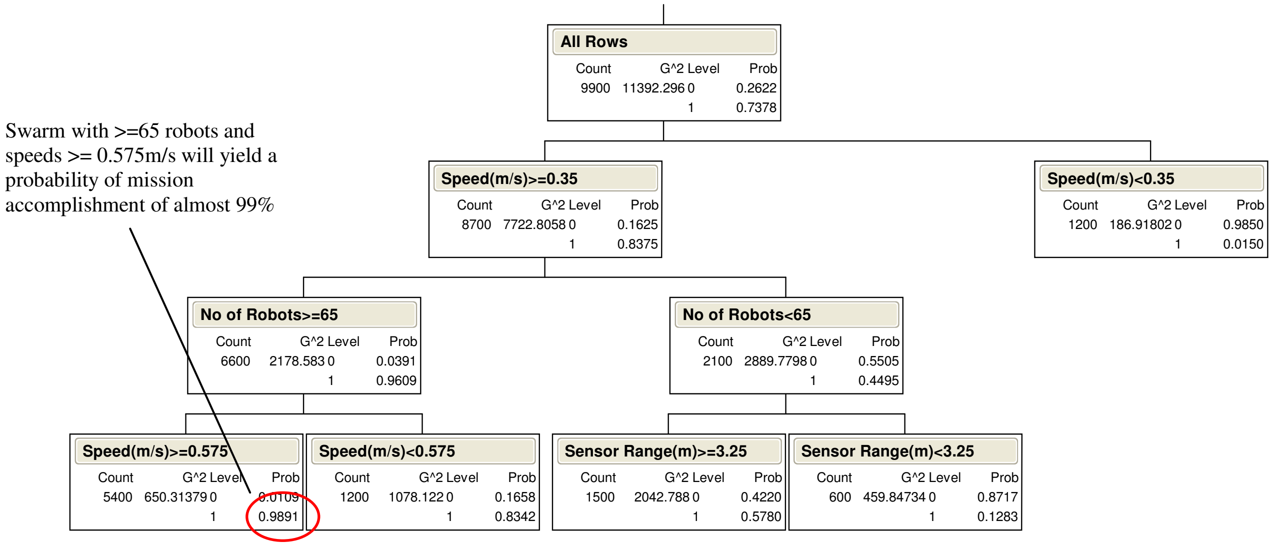


Figure 45. Regression tree of MOE 2 with all factors

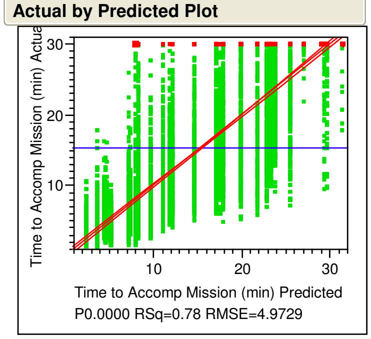
A more intuitive analysis to get insights on the importance of factors on the response is by using the regression tree. A regression tree with MOE 2 as the response and includes all factors (both decision and noise factors) is shown in Figure 45.

It is apparent from the regression tree that speed is the single most important factor in determining mission accomplishment. Out of 9,900 runs, swarms with speed settings less than 0.35m/s have a probability of 0.015 for mission accomplishment, while swarms with speed settings more than 0.35m/s have a probability of 0.838. The next most important factor is the number of robots, indicating a probability of 0.961 of mission accomplishment if this is more than 65. Speed turns out to again be important even for this subset, since it determines the next split. From the graph, deploying more than 65 units of robots with speeds of up to 0.575m/s will yield a probability of mission accomplishment of 0.989. This conclusion is critical to decision makers who are concerned about making sure all IEDs are detected before the allowable time is up.

5. Regression of Time taken to Accomplish Mission (MOE 1)

MOE 1, which is the time taken to accomplish mission, is an important indicator for search and detect missions in view of time constraints. A commander may not have the full 30min at his disposal and may need more than just having the swarm complete its search successfully within this time. MOE 1 is a continuous response variable, thus more analyses can be done on interactions that may provide further substantiation from previous findings (interactions are more challenging to deduce from logistic regressions).

Actual by Predicted Plot



Summary of Fit

RSquare	0.777691
RSquare Adj	0.777264
Root Mean Square Error	4.972932
Mean of Response	15.43701
Observations (or Sum Wgts)	9900

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
No of Robots	1	1	389042.94	15731.59	0.0000
Sensor Range(m)	1	1	134720.81	5447.656	0.0000
Speed(m/s)	1	1	240458.55	9723.334	0.0000
Det Capab	1	1	2643.02	106.8748	<.0001
Det Reset Time(sec)	1	1	219.99	8.8956	0.0029
No of Robots*Sensor Range(m)	1	1	3155.97	127.6167	<.0001
No of Robots*Speed(m/s)	1	1	1275.75	51.5872	<.0001
No of Robots*Det Capab	1	1	3690.83	149.2448	<.0001
No of Robots*Det Reset Time(sec)	1	1	1318.72	53.3245	<.0001
Sensor Range(m)*Speed(m/s)	1	1	3063.94	123.8953	<.0001
Sensor Range(m)*Det Capab	1	1	4134.59	167.1890	<.0001
Sensor Range(m)*Det Reset Time(sec)	1	1	1761.49	71.2289	<.0001
Speed(m/s)*Det Capab	1	1	3555.89	143.7881	<.0001
Speed(m/s)*Det Reset Time(sec)	1	1	5967.48	241.3046	<.0001
Det Capab*Det Reset Time(sec)	1	1	8064.17	326.0879	<.0001
Sensor Range(m)*Sensor Range(m)	1	1	536.14	21.6795	<.0001
Speed(m/s)*Speed(m/s)	1	1	9965.08	402.9542	<.0001
Det Capab*Det Capab	1	1	11842.03	478.8517	<.0001
Det Reset Time(sec)*Det Reset Time(sec)	1	1	8156.73	329.8305	<.0001

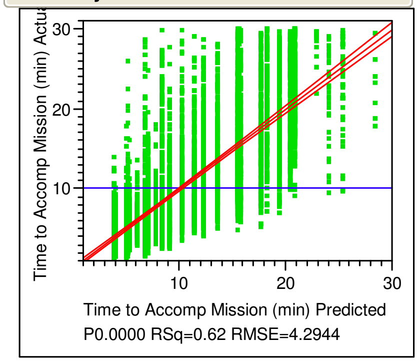
Figure 46. Regression of MOE 1 without noise factors

A regression fit over the full 9,900 runs give an R-square of 0.806 with several interaction and quadratic terms shown in Appendix B. As discussed earlier, it is desired to leave the noise factors out. The model is narrowed down to just the decision factors, with a slight degradation in R-square to 0.777 shown in the Figure 46 (details in Appendix B).

The first inference made from this model (specifically, from the F-ratios) is that the most influential main effects for determining mission completion time are number of robots, sensor range and speed. Det capab and DRT are significant only in interactions and quadratic terms. Several interaction terms surfaced, including number of robots*sensor range, number of robots*det capab, sensor range*speed, sensor range*det capab, speed*det capab, speed*DRT, det capab*DRT. Speed, det capab and DRT appear to have a quadratic effect on the response.

In the experiments that consist of 9,900 runs for non-pheromone robots, only 7,304 runs achieved mission accomplishment within 30min. The regression in Figure 46 takes into account of all the mission failures as well, which is recorded as having mission completion time of 30min. This basis suggests that the model may be skewed by the 30min ceiling imposed on the response of MOE 1 and hence unsuitable for missions that completed considerably less than 30min. A new regression fit is performed on MOE 1 conditioned on mission completion (7,304 points, see Figure 47).

Actual by Predicted Plot



Summary of Fit

RSquare	0.618952
RSquare Adj	0.618272
Root Mean Square Error	4.294442
Mean of Response	10.26101
Observations (or Sum Wgts)	7304

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	26.396328	1.203779	21.93	<.0001
No of Robots	-0.065682	0.002707	-24.26	<.0001
Sensor Range(m)	-0.963412	0.032116	-30.00	<.0001
Speed(m/s)	-4.647512	0.221254	-21.01	<.0001
Det Capab	-0.036194	0.013054	-2.77	0.0056
Det Reset Time(sec)	0.1122617	0.024847	4.52	<.0001
(No of Robots-126.004)*(Det Reset Time(sec)-5.0376)	0.002669	0.000723	3.69	0.0002
(Sensor Range(m)-5.71985)*(Speed(m/s)-1.16514)	0.5007533	0.080371	6.23	<.0001
(Speed(m/s)-1.16514)*(Det Capab-85.3449)	-0.409879	0.057711	-7.10	<.0001
(No of Robots-126.004)*(No of Robots-126.004)	0.0008048	0.000089	9.08	<.0001
(Sensor Range(m)-5.71985)*(Sensor Range(m)-5.71985)	0.114081	0.028055	4.07	<.0001
(Speed(m/s)-1.16514)*(Speed(m/s)-1.16514)	11.16551	0.770526	14.49	<.0001
(Det Capab-85.3449)*(Det Capab-85.3449)	0.0320816	0.002529	12.68	<.0001
(Det Reset Time(sec)-5.0376)*(Det Reset Time(sec)-5.0376)	-0.222202	0.029826	-7.45	<.0001

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
No of Robots	1	1	10857.346	588.7218	<.0001
Sensor Range(m)	1	1	16595.806	899.8805	<.0001
Speed(m/s)	1	1	8137.155	441.2239	<.0001
Det Capab	1	1	141.772	7.6874	0.0056
Det Reset Time(sec)	1	1	376.473	20.4136	<.0001
No of Robots*Det Reset Time(sec)	1	1	251.337	13.6284	0.0002
Sensor Range(m)*Speed(m/s)	1	1	715.918	38.8195	<.0001
Speed(m/s)*Det Capab	1	1	930.284	50.4431	<.0001
No of Robots*No of Robots	1	1	1519.470	82.3908	<.0001
Sensor Range(m)*Sensor Range(m)	1	1	304.939	16.5348	<.0001
Speed(m/s)*Speed(m/s)	1	1	3872.538	209.9820	<.0001
Det Capab*Det Capab	1	1	2966.695	160.8642	<.0001
Det Reset Time(sec)*Det Reset Time(sec)	1	1	1023.599	55.5030	<.0001

Prediction Profiler

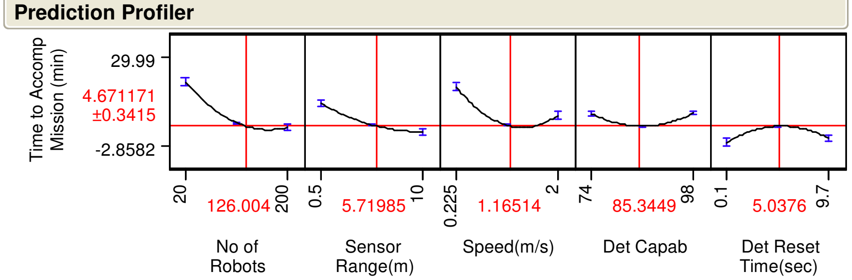


Figure 47. Conditional regression of MOE 1 without noise factors

The conditioned regression model of MOE 1 gives a considerably worse R-square (0.619) as compared to the model regressed over the entire 9,900 runs (0.777). However, it can be seen that the Root Mean Square Error (RMSE) has dropped from 4.972 to 4.294. While the previous regression may explain a higher proportion of the variability in mission completion time with the mission failure observations in the model, the unexplained variability has decreased in the new regression. A likely implication is that the previous model is only valid for missions with high completion times. The model that gives a more accurate insight on MOE 1 is the latter, i.e., the one that leaves out the mission failure observations; but one has to bear in mind this only "tells the story" of the missions that are completed on time.

A check on the signs of the main effects of the conditional regression model of MOE 1 shows consistency with the unconditional regression model. The signs for the main effects for this regression and the logistic regression for mission accomplishment are also similar, implying that main effects have similar effect on MOE 1 (mission accomplishment time) and mission accomplishment (MOE 2). The main effects, number of robots, sensor range and speed again play the main roles in terms of explaining the simulation's behavior. Fewer interaction terms show up as compared to the unconditional regression. They are sensor range*speed and speed*det capab. Quadratic terms that are significant include number of robots, speed and det capab. The terms in this model can be thought of as being important in determining a short mission completion time, conditioned on a mission success. This model is useful together with the deductions from the logistic regression that determine mission accomplishment. It should not be used as a stand-alone model as it does not incorporate mission failures, though the main effects do have similar impact on both MOEs.

6. Regression tree of MOE 1

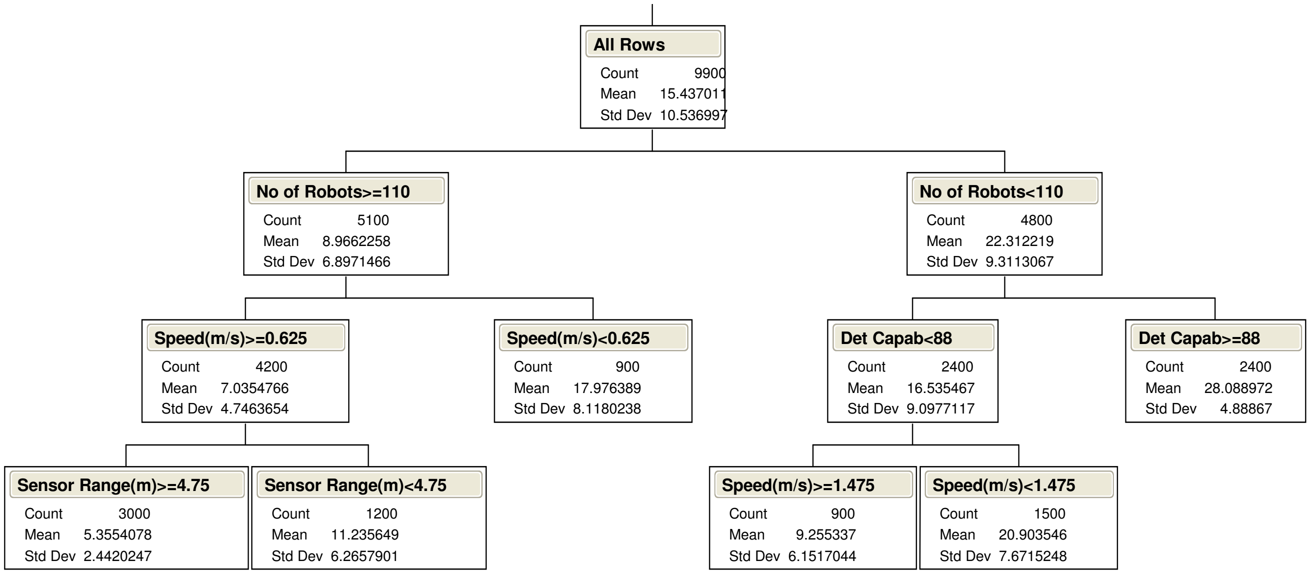


Figure 48. Regression tree of MOE 1 with all factors

A regression tree is performed on the data with MOE 1 as the response, shown in Figure 48. It turns out that the number of robots is the most important factor in achieving a short mission completion time. In fact, with more than 110 robots, the swarm can accomplish its mission with a mean of approximately 9min. A lack in quantity of robots (<110) can be made up for by using a shorter TOT detector (det capab <88) and faster robots (speed at least 1.475m/s), but the mean mission time will be degraded to about 20.9min if these speeds cannot be achieved. In addition, the mean mission time can be further reduced for large robot swarms by using robots with a speed of more than 0.625m/s and detection range of more than 4.75m.

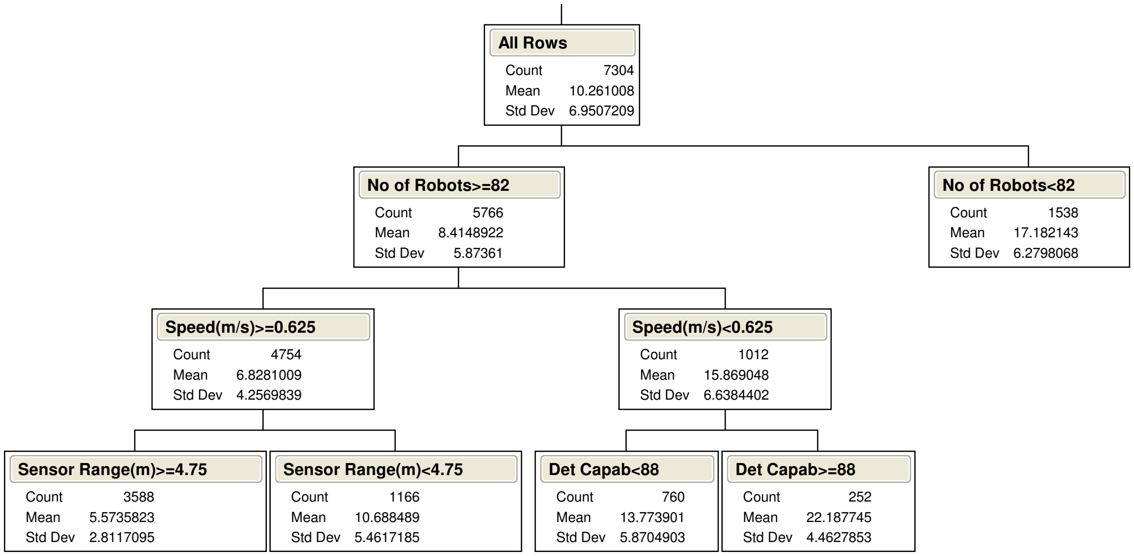


Figure 49. Conditional regression tree of MOE 1 with all factors

It has been acknowledged that the simulation terminates when the mission time hits 30min. Beyond this, the mission accomplishment time is recorded at the 30min ceiling even though there could be zero detections made. As such, it is necessary to delve deeper to ensure this regression tree is not skewed by runs that saw mission failures. An additional regression tree on MOE 1 is done in Figure 49, conditioned on mission accomplishment.

It can be seen from this conditional regression tree that the most important factor for a short mission time is again the number of robots, followed by speed and sensor range. The findings here are consistent with those from the unconditional regression tree for MOE 1, except that the number of robots here is lower (82 vs 110).

7. Quadratic Terms Analyses

a. Detector Capability (Time on Target) as a Quadratic Term

Detector capability turns out as a quadratic term in at least one of the regressions performed. For illustration of this behavior, the prediction profiler in Figure 50 is taken from the logistic regression of MOE 2 with only decision factors. When all factors are set approximately in the middle of the respective range, the probability of mission accomplishment approaches one as the speed of swarm increases, and as the detection capability approaches its middle setting. Detection capability seems to have a quadratic behavior, i.e., the system does not perform so well when the detector requires a low TOT or high TOT. This is not expected as a detector with a low TOT should enhance the probability of mission success[7].

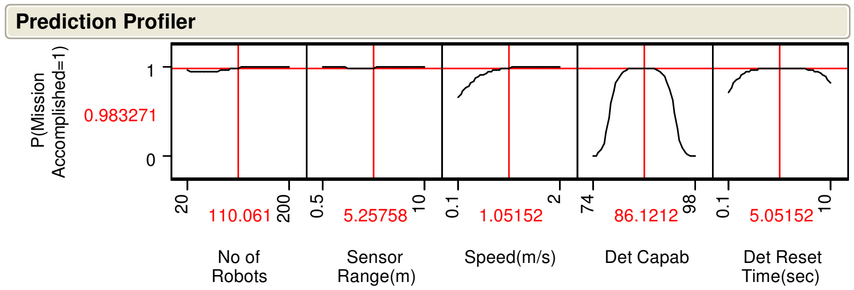


Figure 50. Prediction profiler taken from logistic regression of MOE 2 without noise factors

One possible explanation stems from the fact that a detector triggers into an inactive mode to reset itself whenever a detection is made. If the TOT requirement is low, the detector will be triggered off frequently. Due to the modeling limitation that the sensor responsible for detecting IEDs is also responsible for detecting other robots in the vicinity, the time taken for the detector to reset is also a duration when the robot is unable to sense fellow robots. Hence, it will not repel from other robots for the duration of this triggered state. This in turn lessens the ability to spread out for coverage. Further investigations will be needed to confirm this theory. It must also be noted that this observation should be substantiated further with evidence from other analyses, since the profiler shows only a snapshot of how the response vary with the factors.

b. Speed as a Quadratic Term

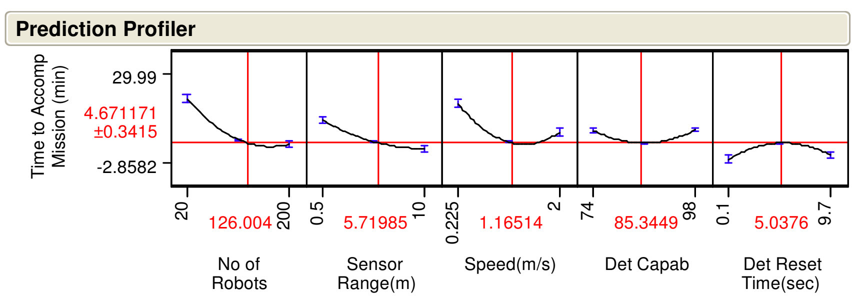


Figure 51. Prediction profiler from conditional regression of MOE 1 without noise factors

A snapshot of the prediction profiler of the regression of MOE 1 with only decision factors is shown in Figure 51, which gives the graphical representation of how the response varies with each of the decision factors. Although the number of robots

[7] Mission accomplishment and mission success are used interchangeably here.

shows up as being quadratic in the regression, this manifests as MOE 1 leveling off rather than fairing worse as it approaches its maximum value (possible quadratic behavior of number of robots is explained in Section 3 of Chapter V). Speed and detector capability show degraded returns, rather than diminishing returns, via their quadratic behavior. In the previous section, the logistic regression for MOE 2 also produced a quadratic detector capability term, with the ensuing discussion on possible detector TOT effect. As for speed, its quadratic behavior can be explained more intuitively. A swarm with slow robots will not help in achieving the coverage needed, but robots that are too fast may end up missing IEDs due to the TOT nature of the detector. Therefore, some degradation in detection performance may be experienced with robots that are too fast.

8. Interaction Terms Analyses

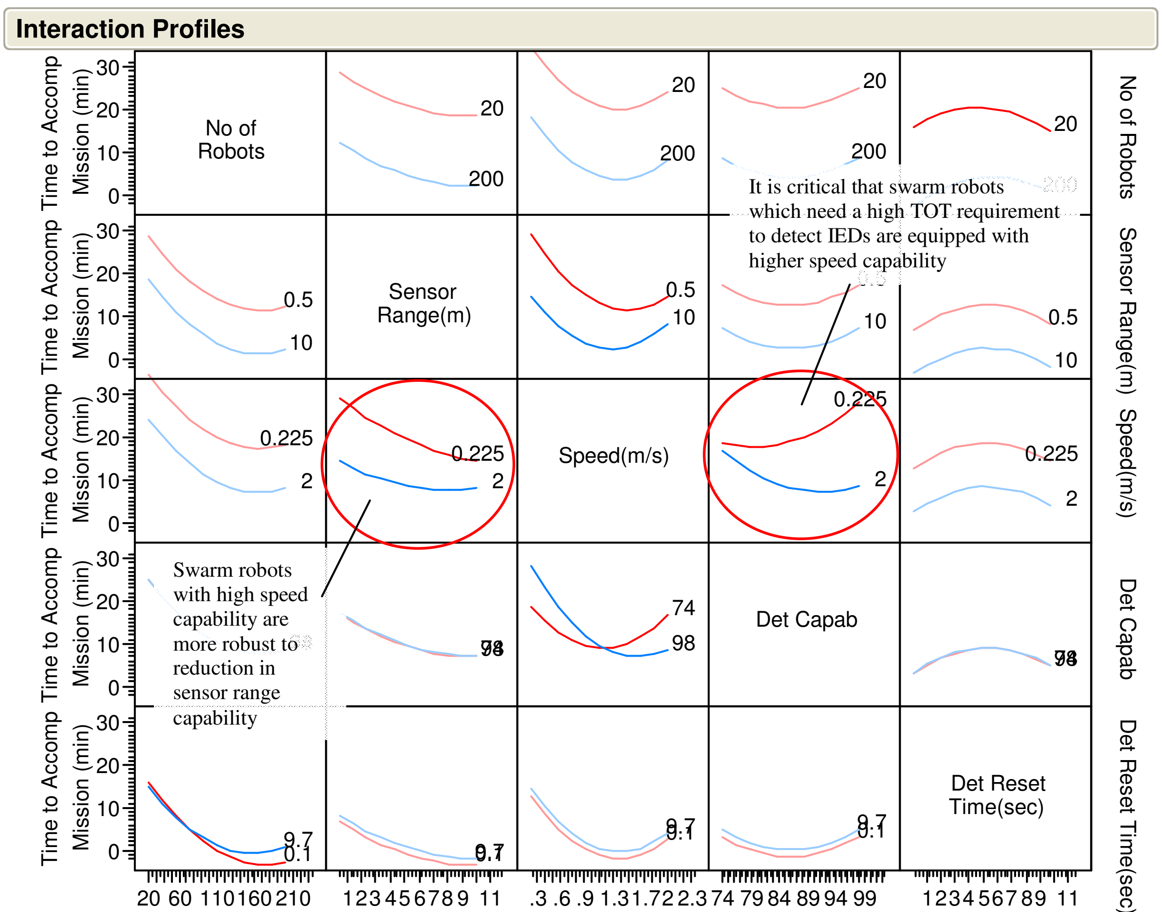


Figure 52. Interaction plot taken from conditional regression of MOE 1 without noise factors

Two-way interactions for the conditional regression of MOE 1 are plotted to explore the interaction effects between the decision factors as shown in Figure 52. In the plot, only the significant interactions are shown as solid lines. It can be observed for the sensor range and speed interaction, a swarm of robots with a high speed setting (2m/s) is more robust to a reduction in sensor range, as compared to having a low speed setting (0.225m/s).

Between speed and detector capability, the interaction is not as straightforward. Speed does not seem to make much difference in enhancing mission completion time when the detector capability is at its lowest TOT setting, but for detectors that have high TOT, it is crucial that the swarm is capable of high speed (2m/s), else there may be a drastic deterioration of performance. In addition, at high speeds, a detector with high TOT performs considerably better than a detector with a low TOT, with quadratic behavior in between. This counter-intuitive behavior has been discussed in Section 7a of Chapter V on detector capability as a quadratic behavior.

Contour plots are a useful way to explore interaction terms further as well as to depict the interaction behaviors better. Contour plots involving interaction terms that have appeared more than once in different regression fits are constructed. The complete series of contour plots done is in Appendix B. The following (Figures 53-55), best viewed in color, are the more interesting ones. Blue regions indicate short mission completion times, while red regions indicate long mission completion times.

a. *Number of Robots vs Detector Capability*

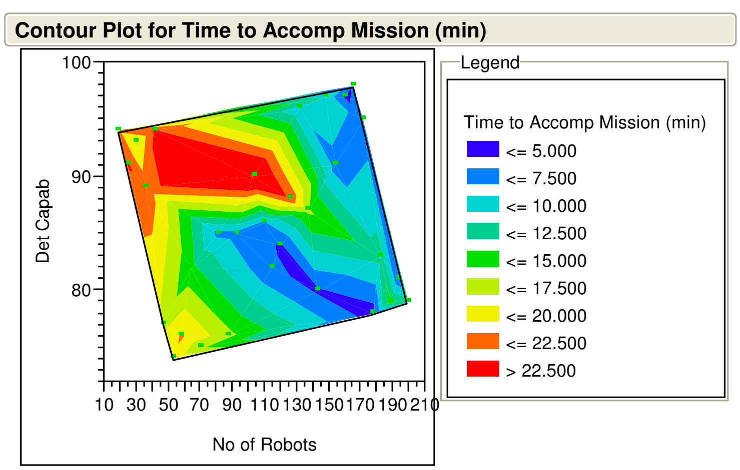


Figure 53. Contour plot for Number of Robots vs Detector Capability

Figure 53 shows that a large number of robots coupled with a capable detector (with a short TOT) work together for short mission completion times. The response variable seems to go into a state of significant deterioration once the number of robots is less than 130 and the detector capability is more than 88 (=2.5sec TOT). The quadratic behavior of detector capability can be observed here where the red region seems to intrude into the middle of two green/blue regions.

b. Detector Capability vs Sensor

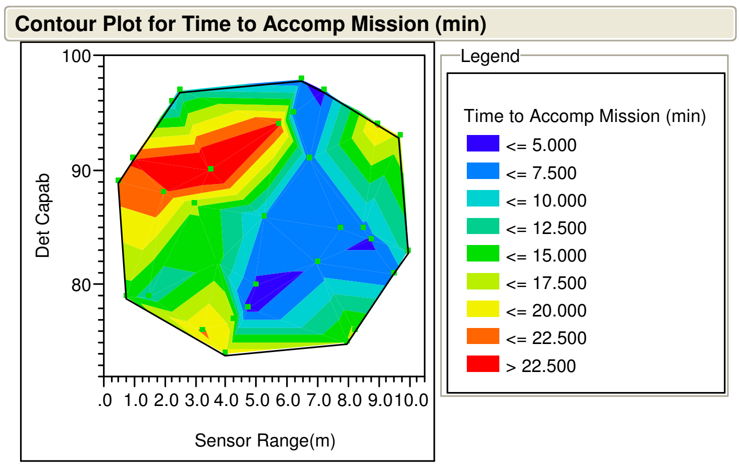


Figure 54. Contour plot for Detector Capability vs Sensor Range

Figure 54 suggests that the robotic swarm generally has a short-medium mission completion time when the sensor range is above 6m. When the sensor range is less than 6m, then a high TOT degrades the effectiveness of the swarm tremendously. Some quadratic behavior can also be observed for TOT.

c. Speed vs Sensor Range

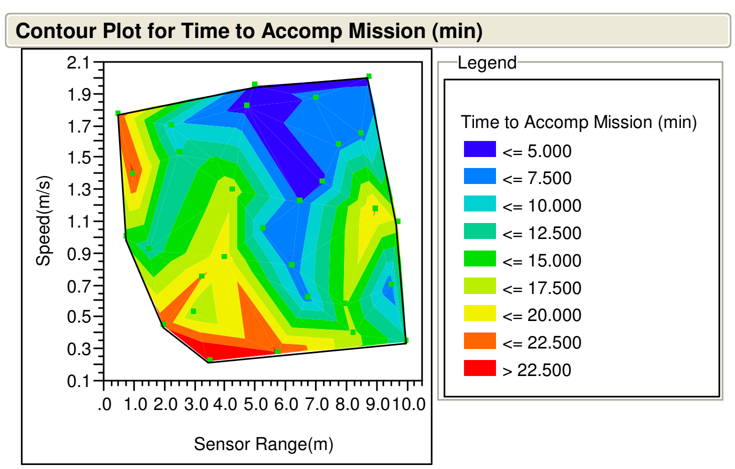


Figure 55. Contour plot for Speed vs Detector Capability

Figure 55 shows speed as being slightly more important than sensor range. As long as speeds are over 1.2m/s, the mission can be completed in a short time as long as sensor range is over 2m. The performance becomes more erratic when speed is less than 1.2m/s.

9. Summary of Findings for Non-Pheromone Robots

Non Pheromone Robots							
	Number of Robots	Sensor Range (m)	Speed (m/s)	Det Capab TOT (sec)	Det Reset Time (sec)	Mean Mission Time (min)	Std Dev Mission Time (min)
Setting 1	82	7.75	1.575	1.843	2.9	6.760	2.566
Setting 2	110	5.25	1.05	1.986	5.1	7.464	3.034
Setting 3	116	7	1.875	1.51	9.1	5.255	3.273
Setting 4	121	8.75	2	1.718	9.4	4.780	2.441
Setting 5	133	2.25	1.7	7.339	3.8	8.146	4.213
Setting 6	144	5	1.95	1.343	0.1	4.591	1.512
Setting 7	155	6.75	0.625	3.176	2.3	6.442	2.322
Setting 8	161	7.25	1.35	9.835	7.5	4.543	1.929
Setting 9	166	6.5	1.225	14.83	8.1	5.063	1.807
Setting 10	172	6.25	0.825	5.84	1.6	6.032	1.658
Setting 11	178	4.75	1.825	1.206	0.4	3.813	1.487
Setting 12	194	9.5	0.7	1.422	4.4	5.572	2.025

Table 4. Swarm settings for non-pheromone robots that produce 100% mission accomplishments

Table 4 shows the design points that produce 100% mission accomplishment for all 100 replications. This gives a direct reference as to which settings for the robot swarm are likely to yield mission completion times within the 30min, as well as how fast (on average) the mission will be completed. The standard deviation of mission completion time is also provided to assist in understanding the range of potential outcomes.

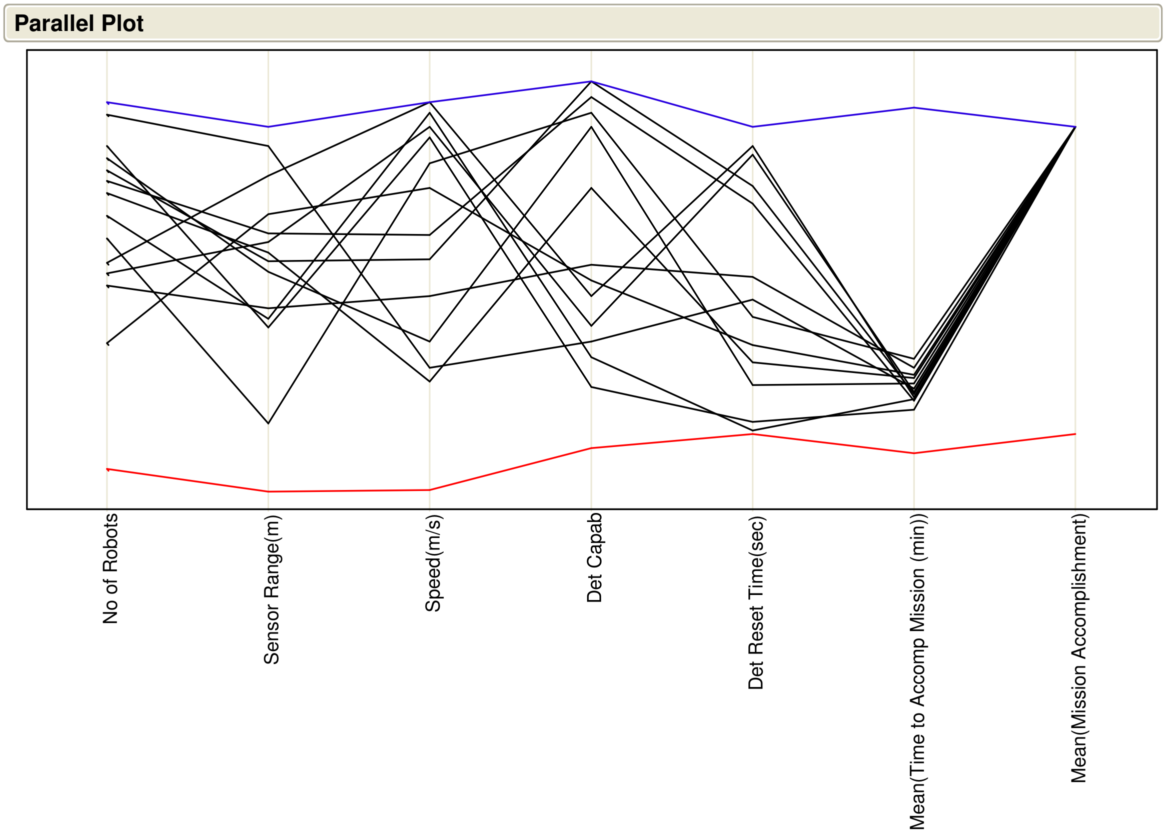


Figure 56. Parallel plot of mission completion times of design points with 100% successful mission (blue and red lines are dummy lines for bounds)

Another overview of the swarm settings with 100% mission accomplishment is shown in the parallel plot in Figure 56. It can be seen that the number of robots and speed clearly have a minimum threshold for the occurrence of these responses. Sensor range has a minimum threshold as well, except for one particular instance with a low sensor range setting that was made up for by having good speed and a large number of robots.

In summary, the behaviors and impacts of various decision factors on the performance of the non-pheromone robots has been explored. It would be misleading to propose optimal factor setting levels, as the settings will only be good for a swarm with this type of algorithm for movement and this type of routine in detection. The main qualitative conclusion drawn is that number of robots, sensor range and speed are the three main decision factors. It is also observed that one has to be wary of possible quadratic effects of number of robots, speed and TOT on the performance of the swarm.

Earlier in this section, it is also determined that instances of drastic failures where very few IEDs are detected are attributed to low speed settings. In addition, it is important to realize that while some factors seem to be more forgiving and can be made up for by having other factors at their high levels, the number of robots and speed have a minimum threshold that must be crossed before the swarm becomes reliable. It turns out that these thresholds are realistic levels from the perspective of currently available technologies.

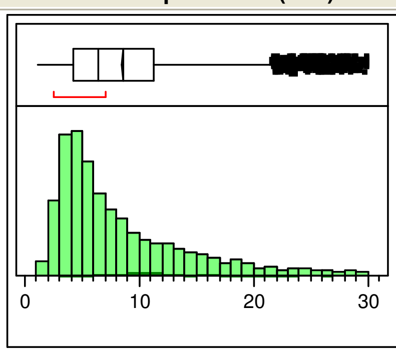
Further analysis using clustering and outlier techniques are performed and are found to agree with the findings here. Details can be found in Appendix C.

B. PHEROMONE ROBOTS

There are two aspects that are of interest for the pheromone robots. First and foremost, it is essential to determine whether swarm robots using virtual pheromones as a shared memory map are more effective than swarm robots without this capability. The second is whether the terms that explain the behavior of the MOEs for the pheromone robots change after the incorporation of two new decision factors and one new noise factor. The following analyses will focus on these aspects.

1. Overview of MOEs

Time to Accomp Mission (min)



Quantiles

100.0%	maximum	29.997
99.5%		28.858
97.5%		25.368
90.0%		17.529
75.0%	quartile	11.234
50.0%	median	6.349
25.0%	quartile	4.198
10.0%		3.120
2.5%		2.179
0.5%		1.623
0.0%	minimum	1.020

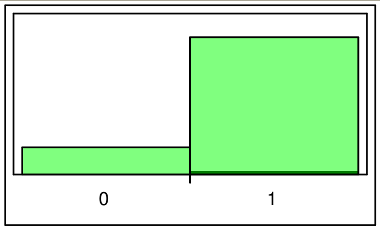
Moments

Mean	8.5480158
Std Dev	6.0227429
Std Err Mean	0.066413
upper 95% Mean	8.6782019
lower 95% Mean	8.4178296
N	8224

Figure 57. Distribution of MOE 1 (conditioned on mission accomplishment)

It can be seen that the time to accomplish the mission (given the mission is accomplished under 30min) is lower than that of the non-pheromone robot case. The mean time to complete mission is now 8.55min, down from 10.26min previously (seen in Figure 57).

Mission Accomplished



Frequencies

Level	Count	Prob
0	1676	0.16929
1	8224	0.83071
Total	9900	1.00000

N Missing 0

2 Levels

Figure 58. Distribution of MOE 2

The probability of mission accomplishment for pheromone robots has also improved, up from 0.738 to 0.831 (seen in Figure 58). This difference of 0.0929 (roughly 12%) is input into a two-sample proportion test to see if it is significantly different.

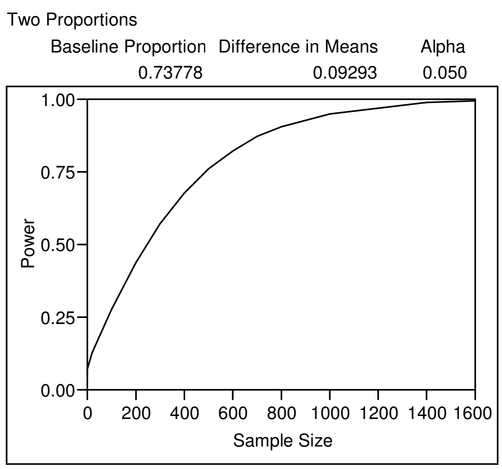
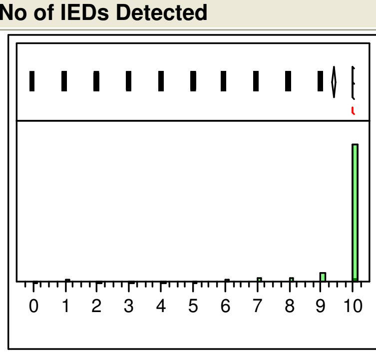


Figure 59. Two-sample proportion test of MOE 2

It can be seen from Figure 59 that MOE 2 for pheromone robots is statistically better than the non-pheromone robots, as the power approaches 1 at a sample size of around 1,600 (we have a sample size of 19,800).

No of IEDs Detected



Quantiles

100.0%	maximum	10.000
99.5%		10.000
97.5%		10.000
90.0%		10.000
75.0%	quartile	10.000
50.0%	median	10.000
25.0%	quartile	10.000
10.0%		8.000
2.5%		2.000
0.5%		1.000
0.0%	minimum	0.000

Moments

Mean	9.4059596
Std Dev	1.7342205
Std Err Mean	0.0174296
upper 95% Mean	9.4401251
lower 95% Mean	9.3717941
N	9900

Figure 60. Distribution of MOE 3

The distribution of MOE 3 for the pheromone robots (seen in Figure 60) also shows evidence that the performance is enhanced; the mean number of IEDs detected is up from 9.15 to 9.41. Even more interesting, the number of observations does not spike at zero IED detections, unlike the non-pheromone robots. In another words, there is no clear "falling off the cliff" phenomenon here where drastic failures occur for a particular setting with the incorporation of pheromones.

2. Logistic Regression of MOE 2 (over Non-Pheromone and Pheromone Robots)

Whole Model Test

Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	7392.851	29	14785.7	0.0000
Full	2932.568			
Reduced	10325.418			

RSquare (U)	0.7160
Observations (or Sum Wgts)	19800

Converged by Objective

Effect Wald Tests

Source	Nparm	DF	Wald ChiSquare	Prob>ChiSq
No of Robots	1	1	405.790403	0.0000
Sensor Range(m)	1	1	489.22281	0.0000
Speed(m/s)	1	1	325.663591	0.0000
Det Capab	1	1	32.4230776	0.0000
Det Reset Time(sec)	1	1	0.15599327	0.6929
Repel Friends	1	1	0.98408813	0.3212
Repel Cover	1	1	78.778362	0.0000
Precision Movt	1	1	41.662432	0.0000
Terrain{2-1&0}	1	1	296.438864	0.0000
Terrain{1-0}	1	1	84.733939	0.0000
Pheromones	1	1	126.810876	0.0000
No of Robots*Speed(m/s)	1	1	78.6039014	0.0000
No of Robots*Pheromones	1	1	40.1600689	0.0000
Sensor Range(m)*Repel Friends	1	1	10.13825	0.0015
Sensor Range(m)*Pheromones	1	1	5.83287744	0.0157
Speed(m/s)*Det Reset Time(sec)	1	1	12.1049385	0.0005
Speed(m/s)*Repel Cover	1	1	122.971429	0.0000
Speed(m/s)*Terrain{1-0}	1	1	15.6329995	0.0001
Speed(m/s)*Pheromones	1	1	25.0221393	0.0000
Det Capab*Pheromones	1	1	18.6958895	0.0000
Det Reset Time(sec)*Precision Movt	1	1	18.3857875	0.0000
Det Reset Time(sec)*Pheromones	1	1	7.02562627	0.0080
Repel Cover*Terrain{2-1&0}	1	1	4.60974412	0.0318
Repel Cover*Pheromones	1	1	9.50280042	0.0021
Precision Movt*Pheromones	1	1	12.4647697	0.0004
No of Robots*No of Robots	1	1	20.0008211	0.0000
Sensor Range(m)*Sensor Range(m)	1	1	27.7759583	0.0000
Speed(m/s)*Speed(m/s)	1	1	60.5034383	0.0000
Det Capab*Det Capab	1	1	12.183032	0.0005

Figure 61. Logistic regression of MOE 2 over non-pheromone and pheromone robots

It can be seen from Figure 61 that the binary predictor variable "pheromones" turns out to be a significant main effect in the model when regressed over the entire data set. From this and the distribution plots, it can be inferred that virtual pheromone capability makes a difference in determining mission accomplishment. This fit gives an R-square of 0.716.

Now that the impact of pheromone capability is established, it is necessary to further explore the pheromone capable robots.

3. Logistic Regression of MOE 2 for Pheromone Robots

Whole Model Test

Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	3433.9537	11	6867.907	0.0000
Full	1068.2028			
Reduced	4502.1566			

RSquare (U)	0.7627
Observations (or Sum Wgts)	9900

Converged by Objective

Effect Wald Tests

Source	Nparm	DF	Wald ChiSquare	Prob>ChiSq
No of Robots	1	1	8.29094003	0.0040
Sensor Range(m)	1	1	73.4479661	0.0000
Speed(m/s)	1	1	13.9196418	0.0002
Det Capab	1	1	7.01280465	0.0081
Det Reset Time(sec)	1	1	4.72652662	0.0297
Pheromone Sensor Range (m)	1	1	22.1331042	0.0000
Pheromone Persistence (sec)	1	1	6.25631106	0.0124
Sensor Range(m)*Pheromone Sensor Range (m)	1	1	14.0970668	0.0002
Speed(m/s)*Pheromone Persistence (sec)	1	1	48.6027346	0.0000
Det Reset Time(sec)*Pheromone Sensor Range (m)	1	1	6.33306345	0.0119
Pheromone Persistence (sec)*Pheromone Persistence (sec)	1	1	3.83545564	0.0502

Figure 62. Logistic regression of MOE 2 without noise factors for pheromone robots

The logistic regression for MOE 2 of the pheromone robots produces a model (seen in Figure 62) that is relatively simpler than that for the non-pheromone robots. With virtual pheromones incorporated, two decision factors are introduced, namely pheromone sensor range and pheromone persistence. Many of the interaction and quadratic terms of the initial five decision factors do not appear in this model. This is due to the behavior being dominated by the effects of virtual pheromones, such that the swarm has **become more robust to small interactions between other decision factors**. This is a key impact that the virtual pheromones are making to the system. It can be seen that sensor range, speed, pheromone sensor range are the main effects that explain most part of the variability in performance. Speed and pheromone persistence appear to have significant interaction as well.

4. Conditional Regression of MOE 1 for Pheromone Robots

Summary of Fit

RSquare	0.670855
RSquare Adj	0.670132
Root Mean Square Error	3.459108
Mean of Response	8.548016
Observations (or Sum Wgts)	8224

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
No of Robots	1	1	135305.33	11308.02	0.0000
Sensor Range(m)	1	1	12845.27	1073.532	<.0001
Speed(m/s)	1	1	42673.17	3566.373	0.0000
Det Capab	1	1	9856.96	823.7865	<.0001
Det Reset Time(sec)	1	1	85.84	7.1736	0.0074
Pheromone Sensor Range (m)	1	1	318.91	26.6525	<.0001
Pheromone Persistence (sec)	1	1	1283.52	107.2695	<.0001
No of Robots*Speed(m/s)	1	1	403.16	33.6934	<.0001
No of Robots*Det Reset Time(sec)	1	1	4439.16	370.9990	<.0001
Sensor Range(m)*Pheromone Sensor Range (m)	1	1	140.41	11.7344	0.0006
Sensor Range(m)*Pheromone Persistence (sec)	1	1	514.54	43.0021	<.0001
Speed(m/s)*Pheromone Sensor Range (m)	1	1	50.90	4.2540	0.0392
Speed(m/s)*Pheromone Persistence (sec)	1	1	1244.84	104.0367	<.0001
Det Capab*Det Reset Time(sec)	1	1	486.81	40.6844	<.0001
Det Capab*Pheromone Sensor Range (m)	1	1	1146.66	95.8312	<.0001
Det Reset Time(sec)*Pheromone Sensor Range (m)	1	1	131.86	11.0197	0.0009
Sensor Range(m)*Sensor Range(m)	1	1	12938.35	1081.312	<.0001
Det Reset Time(sec)*Det Reset Time(sec)	1	1	11686.64	976.7009	<.0001

Figure 63. Regression of MOE 1 without noise factors for pheromone robots (conditioned on mission accomplishment)

With the conditional regression of MOE 1, more interaction terms show up in the model (see Figure 63). An attempt is then made to simplify the model in Figure 64.

Summary of Fit

RSquare	0.668601
RSquare Adj	0.668076
Root Mean Square Error	3.469874
Mean of Response	8.548016
Observations (or Sum Wgts)	8224

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
No of Robots	1	1	89083.104	7398.915	0.0000
Sensor Range(m)	1	1	25526.831	2120.165	0.0000
Speed(m/s)	1	1	21569.221	1791.46	0.0000
Det Capab	1	1	8994.318	747.0350	<.0001
Det Reset Time(sec)	1	1	466.251	38.7251	<.0001
Pheromone Sensor Range (m)	1	1	52.099	4.3272	0.0375
Pheromone Persistence (sec)	1	1	1603.453	133.1769	<.0001
No of Robots*No of Robots	1	1	19029.497	1580.52	0.0000
Speed(m/s)*Speed(m/s)	1	1	16885.966	1402.486	<.0001
Det Capab*Det Capab	1	1	439.814	36.5293	<.0001
Det Reset Time(sec)*Det Reset Time(sec)	1	1	40.588	3.3711	0.0664
Pheromone Sensor Range (m)*Pheromone Sensor Range (m)	1	1	1890.367	157.0069	<.0001
Pheromone Persistence (sec)*Pheromone Persistence (sec)	1	1	890.867	73.9922	<.0001

Prediction Profiler

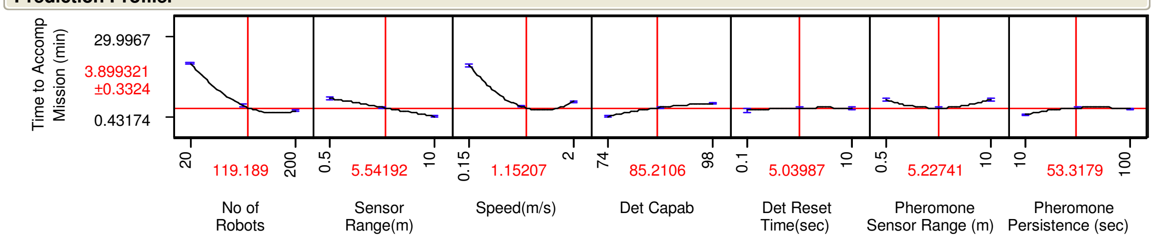


Figure 64. Simplified regression of MOE 1 without noise factors for pheromone robots (conditioned on mission accomplishment)

It is found that the R-square of the regression does not degrade by much when the model is simplified by taking out all interaction terms (0.669). It shows that the number of robots, speed and sensor range remain as the three main effects, with the number of robots and speed showing up as quadratic terms. The explanation for the number of robots being quadratic is as before, i.e., overcrowding hinders robots to move around because of inherent repulsion from one another. This phenomenon is more pronounced in the pheromone robots case because every robot leaves behind a trail that other robots repel. When the entire area of operations is overcrowded with robots and their trails, it becomes an environment that discourages movement and in turn, multiple unique detections. Finally, it can be observed that the pheromone-related factors do not appear to influence the response substantially, though we have established earlier on that the presence of pheromones is significant in the model.

5. Regression Tree of MOE 2 for Pheromone Robots

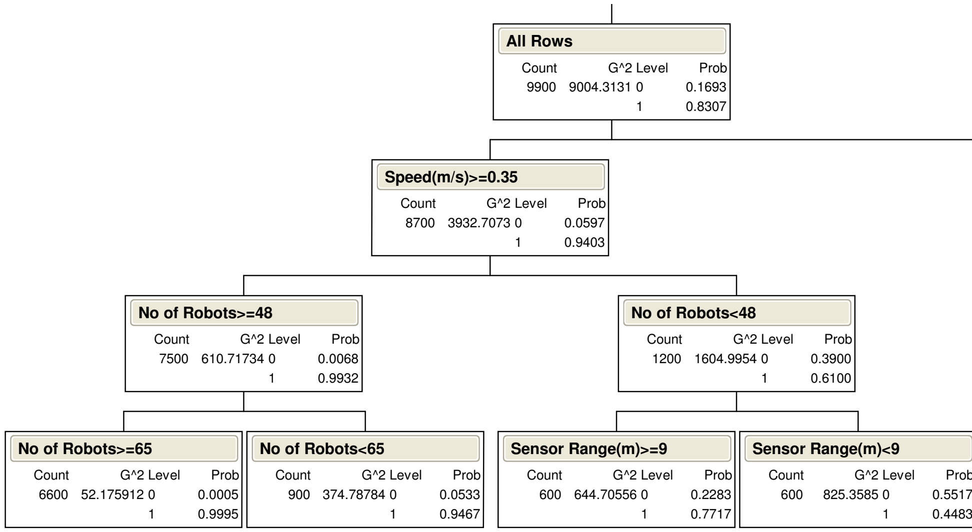


Figure 65. Regression tree of MOE 2 for pheromone robots

The regression tree of MOE 2 for the pheromone robots shows speed as being the most important factor, followed by the number of robots as shown in Figure 65. It can be seen that for a swarm of more than 65 robots and speeds of more than 0.35m/s, the probability of mission accomplishment is almost 100%. With the incorporation of pheromones, the requirements for other decision factors seem to be less stringent.

6. Regression Tree of MOE 2 (over Non-Pheromone and Pheromone Robots)

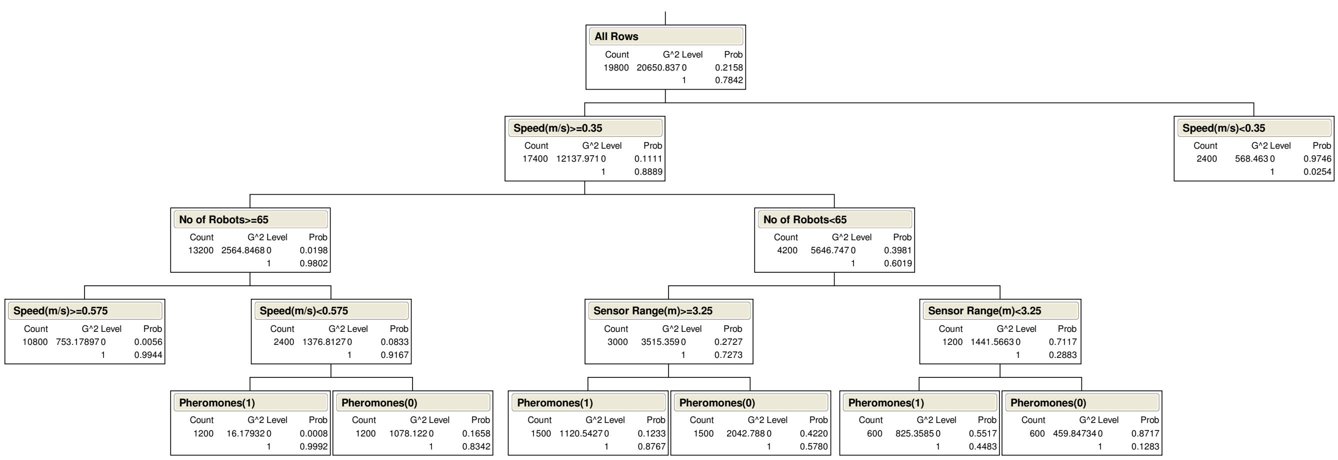


Figure 66. Regression tree of MOE 2 for non-pheromone and pheromone robots

As expected, in a regression tree over the entire data set as shown in Figure 66, speed turns out to be the most important factor in determining mission accomplishment followed by the number of robots. The importance of virtual pheromones then kicks in "across the board," suggesting that it provides a boost in performance for the robot swarm for all cases.

7. Summary of Findings for Pheromone Robots

Pheromone Robots									
	Number of Robots	Sensor Range (m)	Speed (m/s)	Det Capab TOT (sec)	Det Reset Time (sec)	Pheromone Sensor Range (m)	Pheromone Persistence (sec)	Mean Mission Time (min)	Std Dev Mission Time (min)
Setting 1	71	8	0.575	1.041	6.6	4	40.9	11.816	4.139
Setting 2	82	7.75	1.575	1.843	2.9	4.25	94.4	5.914	2.680
Setting 3	88	8.25	0.4	1.092	6.3	6.75	18.4	11.222	4.087
Setting 4	93	8.5	1.65	1.843	4.1	7.25	100	4.365	2.249
Setting 5	110	5.25	1.05	1.986	5.1	5.25	55	5.571	1.745
Setting 6	116	7	1.875	1.51	9.1	9.5	26.9	4.045	1.626
Setting 7	121	8.75	2	1.718	9.4	1.5	29.7	4.414	1.653
Setting 8	133	2.25	1.7	7.339	3.8	3.75	91.6	5.783	2.097
Setting 9	138	3	0.525	2.151	7.2	6.25	15.6	10.312	4.099
Setting 10	144	5	1.95	1.343	0.1	0.5	35.3	4.460	1.503
Setting 11	149	2.5	1.525	9.835	3.5	6.5	69.1	5.161	2.022
Setting 12	155	6.75	0.625	3.176	2.3	2.25	12.8	5.472	1.656
Setting 13	161	7.25	1.35	9.835	7.5	2	60.6	3.548	1.061
Setting 14	166	6.5	1.225	14.83	8.1	7.75	57.8	4.018	1.255
Setting 15	172	6.25	0.825	5.84	1.6	8	21.3	5.086	1.490
Setting 16	178	4.75	1.825	1.206	0.4	9.75	43.8	3.413	1.217
Setting 17	183	10	0.35	1.608	4.7	4.75	85.9	8.732	2.822
Setting 18	189	0.75	1	1.271	6.9	1.75	71.9	6.647	1.913
Setting 19	194	9.5	0.7	1.422	4.4	5.5	77.5	3.929	1.025
Setting 20	200	1.5	0.925	1.271	8.8	7	63.4	6.544	2.532

Table 5. Swarm settings for pheromone robots that produce 100% mission accomplishments

Compared to the non-pheromone robots, eight more settings give 100% mission accomplishment (as shown in Table 5). Pheromone sensor range and pheromone persistence seem to reveal no patterns in ranges or thresholds that must be met.

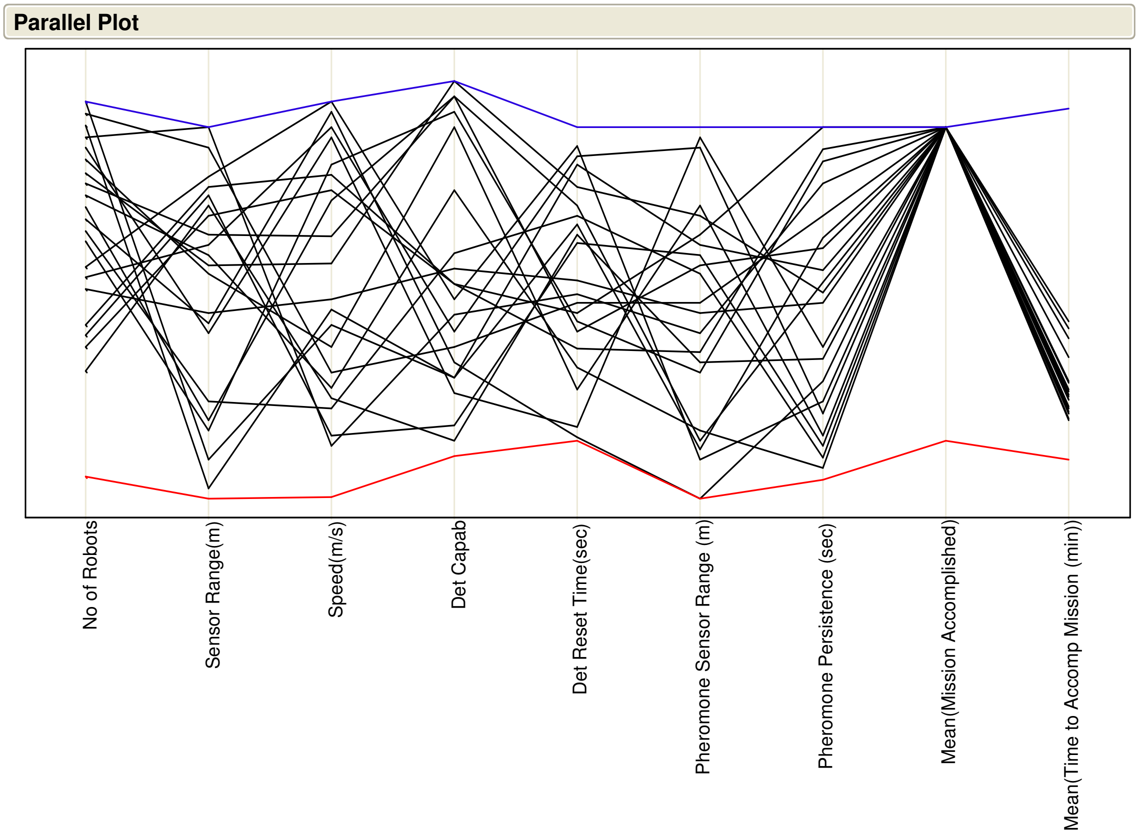


Figure 67. Parallel plot of mission completion times of design points with 100% successful mission (blue and red lines are dummy lines for bounds)

The parallel plot in Figure 67 shows that the minimum thresholds for each factor setting seem to have relaxed as compared to the non-pheromone robots. Some design points with 100% mission accomplishments happen when sensor range is quite close to its lower bound. The number of robots remains as the only factor that strongly suggests a minimum threshold.

Figure 68. Effects of pheromones on different levels of terrain

From the comparison of the distribution plots of MOE 2 broken down to the different levels of terrain in Figure 68, it is clear that virtual pheromones make a difference in all types of terrain in terms of effectiveness enhancement. In fact, as the terrain gets more challenging, the incorporation of virtual pheromones makes a larger difference.

Overall, the incorporation of virtual pheromones as part of the swarm robot capability improves the effectiveness of the swarm by about 12% in terms of mission completion. The main factors that influence the system the most remain as number of robots, sensor range and speed, with indications of number of robots and speed being quadratic. It is interesting that in general, the two new pheromone-related decision factors are not significant in the regression as main effects, which implies their presence alone is enough to make a difference, regardless of their setting levels. Virtual pheromones enhance the robustness of the swarm by making it less sensitive to interactions between various decision factors, and the presence of virtual pheromones tends to mitigate performance degradations that occur when other decision factors are set at low levels. In general, the findings agree with those from the analysis using clustering and outlier techniques, in which the details are laid out in Appendix C.

VI. LIMITATIONS, CONCLUSIONS AND FUTURE WORK

A. LIMITATIONS OF MODELING MOVEMENT AND DETECTION

The methodology in the behavior and detection by the swarm robots can be summarized in the following.

The movement and behavioral modeling is analogous to Batalin's Molecular approach:

- Repel from other robots to attain spread;
- Avoid obstacles; and
- Continuous movement and coverage is the desired **emergent behavior.**

The biggest limitation in modeling the movement and behavior of a swarm is how MANA performs the movement algorithm, which is hard-coded into the software. In swarm robotics, movements of robots are more definite and deliberate. There is a pre-defined reaction for each encounter with a fellow robot or an obstacle. For instance, this reaction could be turning 90 degrees to the right every time it senses and obstacle, or turning 180 degrees every time it senses another robot. When no objects are present within its sensor range, the robot simply moves in a straight line.

The movement algorithm in MANA does not work in this way. Rather, it models each agent (robot) deciding on its next move at every time step (Gill and Greiger, 2003). It calculates the penalty incurred of all nine possible moves (including staying put) it can make at every time step and eventually chooses the one with the least penalty. If there is a tie, it will be stochastically broken by means of a generated random number. Occurrences of ties can be partially controlled by the Precision Move parameter. A larger Precision Move value allows more ties to happen and hence the movement will be less predictable and vice versa. The problem lies in instances when a robot has no encounter with any objects. In practice, a robot moves in a straight line when there are no externalities. Unfortunately in MANA, this is when the movement of the robot is most unpredictable because the penalties incurred in all the possible moves are exactly the same and hence will be determined randomly. It is not possible to model a robot moving in a straight line unless waypoints are given. Waypoints will have to be pre-planned by the operator, which goes against the principles of autonomous behavior of swarm robots.

One way that is used to mitigate this is by activating the Navigate Obstacles (momentum) option in MANA, which encourages the robot to continue moving in its current direction. It is observed that with this option unchecked, the robot does what some term as a "random walk" and does not move away much from its original location when there are no objects in its surroundings. However, this option, as the name speaks, is designed to assist the agents in navigating around obstacles. For example, if a robot encounters a wall, it will keep going along the wall until it ends so that it can turn towards where it originally desired to go. MANA does some manipulation to the penalty function so the agent incurs a lower penalty if it keeps moving in the current direction. When this option is checked, the robot tends to sustain better in moving in a general direction, but still significantly deviates from a straight line.

Another limitation in the modeling robotic movement algorithms is priority. The method of putting tasks in order of priority whenever there is a conflict of interest is fundamental to all robotic movement algorithms. For example, Batalin's robots perform obstacle avoidance as first priority, followed by repelling from other robots and then moving in a straight line from the last chosen direction until its sensors pick up something again. Swarm robots are also typically programmed to stay within a certain distance from one another to prevent any from straying and losing contact. In MANA, priorities cannot be assigned and restrictions cannot be imposed on distances between robots; every move takes into consideration of everything a robot senses. A robot in reality will also have other parameters like refresh rates and lag time between sensor and actuator. These are not captured in the agent based model. Every time step, assumed to be 0.1sec in the model, the robot makes a movement choice. This is not a good representation of the actual behavior.

Despite the above limitations, MANA is able to replicate the desired emergent behavior quite well. With the right combination of parameter settings and movement options based on Batalin's algorithm principles, the agents exhibit the effects of spread, coverage, and ongoing patrol throughout the area so that multiple robots detect different IEDs. This is ultimately what one strives to get out of swarm robots in a search and detect mission.

The detection methodology is based approximately on how a FIDO detector works:

- The detector is in continuous active state;
- The IED is detected with a probability based on a time on target (TOT);
- The detector triggers into a dormant state for a fixed duration as the device takes time to reset back to its baseline;
- The coordinates where a detection occurs is sent to operator (not modeled); and
- The mission ends in 30min.

First and foremost, it should be highlighted that there are many ways that a detector can be modeled. The detector modeled is switched on from the beginning and only triggers into a dormant state when it has just made a detection. This is based on the FIDO. Other detection routines are possible, such as detectors that are good for only one detection, or detectors that require more than a few positive samples to acknowledge a detection. A detector may not have a fixed TOT, but instead the TOT may be dependent on the explosive vapor level emanated from the IED.

Furthermore, many enhancements can be introduced to the swarm to increase its effectiveness. The ability for a robot (that just detected a target) to get neighboring robots to swarm towards a target and interrogate it further is one way to improve multiple detections. However, this requires some form of communication ability, e.g., emitting an acoustic chirping to attract other robots when a detection is made. Another possibility is a scenario where robots use a primary base detector that is on continuously, and a more powerful detector as a secondary means used to interrogate the target further when the base detector picks up any explosive content. This requires the robot to stop in its track when the base detector is triggered; the secondary detector will then have to come online to confirm the detection. In this case, there may not be a requirement for multiple unique detections from different robots. This enhancement would require some modification to the models in this thesis.

The bottom line is, although every effort is made to setup the robot and detector routine as a general representation of how a robot swarm is envisaged to carry out a search and detect mission, the insights and findings are applicable only to this setup. One should not directly draw the same conclusions on the impact of the decision factors and the effect of various settings if the setup of the robots and detector differ from the one in this research. Modifications to the model will be necessary to get more accurate insights.

B. OTHER LIMITATIONS OF THE MODELS

There are other aspects of the models in this thesis that limit the extent of their generalization to the real world. As mentioned in the analysis section, one aspect that could not be differentiated was the sensor suite of the robot. The agents in MANA are limited to one set of sensor specification, when in practice up to three different types of sensors are equipped on the robot. Ideally, the IED detector should be separately modeled from other sensors, including visual and acoustic sensors that detect the presence of other robots as well as obstacles. One implication in the model is that both the IED detector and its sensor suite are triggered to the dormant state whenever a detection is made, which is not what is desired.

Another limitation is with the modeling of pheromone sensor range. Pheromone sensor range is not influenced by line of sight in the model. This is not realistic because pheromone sensors should not be able to detect through walls. In the simulation, because of the way it is modeled, there can be instances where the pheromone sensors pick up pheromone trails left behind by another robot that is on the other side of the wall.

Next, the attempt to quantify a difficulty level for different terrains is not a straightforward issue. In the terrain files, additional obstacles are added to the area of operations and this is quantified to be the “next level of difficulty.” This is a subjective approach in quantifying levels of terrain difficulty, and by no means complies to any standards. Another approach could be to zoom down on specific types of terrain of interest, collect some data about robot mobility over these terrains, and convert them into a terrain files for the simulation to be run on.

One aspect that the models did not explicitly capture is the communication of location information back to the operator whenever a detection is made by a swarm robot. This is a key capability that a swarm robot must be equipped with, and the timeliness, accuracy, and reliability of this communication should be captured as part of the model for a more complete analysis. It should not just be assumed as a given.

C. CONCLUSIONS

The objectives of this research are to explore ground swarm robotics using an agent based simulation approach, as well as to investigate how various decision and noise factors impact its effectiveness in a search and detect mission, in the hope of providing these insights to swarm robotics researchers and engineers to help realize this concept earlier. A by-product of the modeling is the assessment of the feasibility of using agent based simulation in modeling swarm robotics. Both objectives have been given the respective analysis. There are no definite quantitative conclusions from this research but qualitative ones are aplenty.

For a robot swarm to be effective on a search and detect mission and **assuming the presumed movement and detection routine is adopted**, speed appears to be the most vital factor that has a minimum threshold. It is encouraging to note that the results seem to suggest that this minimum threshold required of speed is attainable from the perspective of modern day technology. A robot that moves as fast as the Mini-Whegs™ seems to be sufficient to do the job. There also seems to be a minimum required number of robots needed. Results point to a minimum threshold of approximately 80 for a 50m by 50m area, but one should bear in mind that when tactics and initial dispositions of robots are manipulated with, the number of robots required will drop further. Sensor range (or detector range) seems to have a minimum threshold, but the threshold becomes be less stringent and can be compensated by other factors with the incorporation of virtual pheromones. As for the time on target and reset time aspect of the detector, they are not as critical as the other decision factors as a main effect, but do contribute via interactions with some other factors. Last but not least, equipping swarm robots with virtual pheromones not only improves the swarm's effectiveness, but also enhances its robustness and reliability. It is strongly recommended that virtual pheromones be incorporated as a capability of the swarm robot.

D. FUTURE WORK

Perhaps the most fitting follow-up to this research is to investigate the tactics, techniques, and procedures (TTPs) used in the deployment of a swarm. The scenarios in this research limit the starting location of the robot swarm to one possibility. Some test runs show that the spread and coverage are attained much faster when multiple starting locations were used. One should bear in mind that while the operator could choose to split the swarm into many different starting locations in an effort to gain an initial spread of the robots, the choices might be limited by the circumstances of the situation.

Experiments can be done to find the optimal robot density given other decision factors are fixed. Some results have suggested that the number of robots may have a quadratic effect on the performance of the swarm. In addition, there may be diminishing returns in the performance of the swarm as more robots are added to it given the area is fixed. The mix of TTPs and the knowledge of optimal robot density could provide the operator some flexibility and options in the deployment of the swarm robots.

Further versions of modeling swarm robots may consider overcoming the limitations of the model as stated. Analysis can also be done to find out combinations of parameters that give robustness to the swarm such that slight deviations from those settings do not produce large degradations. In addition, robot failures and redundancy are definitely fascinating characteristics to explore on swarm robotics. One could model robot failures or trapped robots in the progress of the mission, and assess how sensitive or robust the robot swarm is to these perturbations. External disruptions by hostile elements could also be modeled.

Other agent based software can also be explored to validate how well they are suited to model swarm robotics. For example, in MANA, priorities cannot be assigned and restrictions cannot be imposed on distances between robots, but this can easily done in Pythagoras, an agent based software platform developed by Northrop Grumman.

As mentioned before, the model is not complete without modeling the actual communication that takes place between the robots and the operator. This is an additional aspect to consider that will reveal more insights on the factors within communications such as message accuracy, comms link capacity and comms latency.

As for the design of experiment, it is recommended that future experiments explore more fidelity in terms of the design points since the simulations do not take a long time to run on the high performance computers. The NOLH for an 8-11 factor gives 33 design points, but a simple re-arrangement of columns in the matrix will yield new design points that can be explored later on for better resolution within the ranges of each factor (especially for contour plots and quadratic, interaction analyses). On another note, some of the analyses were hampered due to the simulation end time ceiling imposed at 30min. Relaxing this time limit can be to let the simulation to keep running until all IEDs are detected (with very high probability) so that conditional regressions do not need to be performed, and a single MOE of mission accomplishment time will be sufficient for analyses.

In a nutshell, agent based simulation is found to have huge potential as a means to investigate swarm robotics and obtain insights on the impact of various factors on the overall effectiveness. Swarm robots produce much uncertainty in terms of its emergent behavior from multiple dynamic interactions, which is what agent based simulations were designed to examine. With the incorporation of an efficient DOE and data farming methodologies, roboticists and engineers should consider leveraging on this tool to assist in the development and progress of swarm robots to be employed in the real world.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. RUBY SCRIPT FOR EXTRACTING MOE

```
def detection(target_set,target_id,detector_id,det_time,finalhash)
  target_set[target_id]=Hash.new unless target_set.has_key?(target_id)
  old_size=target_set[target_id].size
  target_set[target_id][detector_id]=1 unless
target_set[target_id].has_key?(detector_id)
  if target_set[target_id].size == 3 && target_set[target_id].size !=
old_size
    finalhash[target_id.to_i]=det_time.to_i
  end
end

f=File.new("S37boutput.txt","w")
d=File.open("C:/Documents and Settings/Terence Ho/My
Documents/NPS/Thesis/"+\
"Maui Results August/S37b/S37b_29Sep_2006_09_29_06_54_54.csv","r")
d.gets
f<<"Index,Excursion,No of Robots,Sensor Range,Speed,Det Capab,Det Reset
Time,"+\
"Repel Friends,Repel Cover,Precision Movt,Repel Pheromones,Pheromone
Sensor Range,Pheromone Persistence,B Kill,Red Kill,B Goal,R Goal,Time
Steps,"+\
"Sqd1Kill,Sqd2Kill,Sqd3Kill,Sqd1Inj,Sqd2Inj,Sqd3Inj,Tgt1,Tgt2,Tgt3,Tgt4
,Tgt5,Tgt6,"+\
"Tgt7,Tgt8,Tgt9,Tgt10"+"\n"

Dir["C:/Documents and Settings/Terence Ho/My Documents/NPS/Thesis/Maui
Results August"+\
"/S37b/m_detect/multidetect.*"].each do |currentFile|
  File.open(currentFile,"r") do |infile|
    target=Hash.new
    final=Hash.new
    while line = infile.gets do
      values = line.chomp.split(/,/)
      time = values[0]
      if time.to_i > 10
        detection(target,values[4],values[3],time,final)
      end
    end
  final2=final.to_a
  final3=final2.flatten #.values_at(1,3,5,7,9,11,13,15,17,19)
  designPt=d.gets.chomp #unless
d.gets.chomp.split[0]=="random_index,Excursion_Number,Squad"
  f<<designPt + final3[1].to_s + "," + final3[3].to_s + "," +
final3[5].to_s + "," + final3[7].to_s + \
  "," + final3[9].to_s + "," + final3[11].to_s + "," + final3[13].to_s
+ "," + final3[15].to_s + \
  "," + final3[17].to_s + "," + final3[19].to_s + "\n"
  end
end
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. OTHER REGRESSIONS

Whole Model Test

Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	3772.6503	18	7545.301	0.0000
Full	1923.4977			
Reduced	5696.1480			

RSquare (U)	0.6623
Observations (or Sum Wgts)	9900

Converged by Gradient

Lack Of Fit

Source	DF	-LogLikelihood	ChiSquare
Lack Of Fit	80	41.9844	83.96871
Saturated	98	1881.5133	Prob>ChiSq
Fitted	18	1923.4977	0.3590

Parameter Estimates

Term	Estimate	Std Error	ChiSquare	Prob>ChiSq
Intercept	7.54104657	0.9633634	61.27	<.0001
No of Robots	-0.050358	0.0019583	661.29	<.0001
Sensor Range(m)	-0.5680034	0.0272916	433.16	<.0001
Speed(m/s)	-3.8426623	0.2219552	299.73	<.0001
Det Reset Time(sec)	-0.0558565	0.0281064	3.95	0.0469
Repel Friends	-0.0162937	0.0074215	4.82	0.0281
Repel Cover	-0.0695414	0.0082295	71.41	<.0001
Precision Movt	-0.0182713	0.0019818	85.00	<.0001
Terrain{2-1&0}	0.60291883	0.0437775	189.68	<.0001
Terrain{1-0}	0.34320854	0.0535031	41.15	<.0001
(No of Robots-110.061)*(Speed(m/s)-1.05152)	0.0413949	0.0049566	69.75	<.0001
(No of Robots-110.061)*(Precision Movt-200.121)	-0.000103	0.0000437	5.56	0.0184
(Speed(m/s)-1.05152)*(Repel Cover+30.1212)	-0.1177821	0.0132347	79.20	<.0001
(Speed(m/s)-1.05152)*Terrain{1-0}	0.33400786	0.1257864	7.05	0.0079
(Det Reset Time(sec)-5.05152)*(Repel Friends+42.5152)	-0.035501	0.0037312	90.53	<.0001
(Det Reset Time(sec)-5.05152)*(Precision Movt-200.121)	-0.0017155	0.0005686	9.10	0.0026
(No of Robots-110.061)*(No of Robots-110.061)	0.00036857	0.0000411	80.41	<.0001
(Sensor Range(m)-5.25758)*(Sensor Range(m)-5.25758)	0.13218111	0.0150334	77.31	<.0001
(Speed(m/s)-1.05152)*(Speed(m/s)-1.05152)	3.01794056	0.5657881	28.45	<.0001

For log odds of 0/1

Effect Wald Tests

Source	Nparm	DF	Wald ChiSquare	Prob>ChiSq
No of Robots	1	1	661.291948	0.0000
Sensor Range(m)	1	1	433.155271	0.0000
Speed(m/s)	1	1	299.732503	0.0000
Det Reset Time(sec)	1	1	3.94946656	0.0469
Repel Friends	1	1	4.8200679	0.0281
Repel Cover	1	1	71.4073771	0.0000
Precision Movt	1	1	85.0038992	0.0000
Terrain{2-1&0}	1	1	189.677371	0.0000
Terrain{1-0}	1	1	41.148957	0.0000
No of Robots*Speed(m/s)	1	1	69.7479728	0.0000
No of Robots*Precision Movt	1	1	5.56067945	0.0184
Speed(m/s)*Repel Cover	1	1	79.200507	0.0000
Speed(m/s)*Terrain{1-0}	1	1	7.05091918	0.0079
Det Reset Time(sec)*Repel Friends	1	1	90.5263952	0.0000
Det Reset Time(sec)*Precision Movt	1	1	9.10265108	0.0026
No of Robots*No of Robots	1	1	80.4114366	0.0000
Sensor Range(m)*Sensor Range(m)	1	1	77.3074951	0.0000
Speed(m/s)*Speed(m/s)	1	1	28.4520572	0.0000

Figure 69. Logistic regression of MOE 2 with all factors for non-pheromone robots

Whole Model Test

Model	-LogLikelihood	DF	ChiSquare	Prob>ChiSq
Difference	3550.7055	19	7101.411	0.0000
Full	2145.4425			
Reduced	5696.1480			

RSquare (U)	0.6234
Observations (or Sum Wgts)	9900

Converged by Gradient

Lack Of Fit

Source	DF	-LogLikelihood	ChiSquare
Lack Of Fit	13	118.8872	237.7744
Saturated	32	2026.5553	Prob>ChiSq
Fitted	19	2145.4425	<.0001

Parameter Estimates

Term	Estimate	Std Error	ChiSquare	Prob>ChiSq
Intercept	0.31880694	1.5801656	0.04	0.8401
No of Robots	-0.0458664	0.001989	531.76	<.0001
Sensor Range(m)	-0.5745866	0.0360781	253.64	<.0001
Speed(m/s)	-3.6164504	0.1861493	377.44	<.0001
Det Capab	0.09045586	0.0158611	32.52	<.0001
Det Reset Time(sec)	-0.0615535	0.0529029	1.35	0.2446
(No of Robots-110.061)*(Sensor Range(m)-5.25758)	-0.0078685	0.0022949	11.76	0.0006
(No of Robots-110.061)*(Speed(m/s)-1.05152)	-0.3177235	0.0375808	71.48	<.0001
(No of Robots-110.061)*(Det Capab-86.1212)	-0.0123198	0.001846	44.54	<.0001
(No of Robots-110.061)*(Det Reset Time(sec)-5.05152)	-0.0174263	0.0034005	26.26	<.0001
(Sensor Range(m)-5.25758)*(Speed(m/s)-1.05152)	1.44554492	0.241651	35.78	<.0001
(Sensor Range(m)-5.25758)*(Det Capab-86.1212)	0.12644839	0.0160112	62.37	<.0001
(Sensor Range(m)-5.25758)*(Det Reset Time(sec)-5.05152)	0.94845113	0.10862	76.24	<.0001
(Speed(m/s)-1.05152)*(Det Capab-86.1212)	0.70075613	0.1655566	17.92	<.0001
(Speed(m/s)-1.05152)*(Det Reset Time(sec)-5.05152)	-1.4562294	0.2025899	51.67	<.0001
(Det Capab-86.1212)*(Det Reset Time(sec)-5.05152)	-0.0374135	0.0148923	6.31	0.0120
(No of Robots-110.061)*(No of Robots-110.061)	-0.0003776	0.0001537	6.04	0.0140
(Sensor Range(m)-5.25758)*(Sensor Range(m)-5.25758)	-0.2544672	0.0518858	24.05	<.0001
(Det Capab-86.1212)*(Det Capab-86.1212)	0.07056569	0.0131823	28.66	<.0001
(Det Reset Time(sec)-5.05152)*(Det Reset Time(sec)-5.05152)	0.11754178	0.0478356	6.04	0.0140

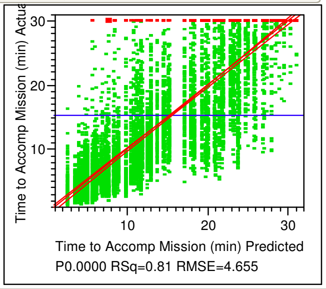
For log odds of 0/1

Effect Wald Tests

Source	Nparm	DF	Wald ChiSquare	Prob>ChiSq
No of Robots	1	1	531.756014	0.0000
Sensor Range(m)	1	1	253.643246	0.0000
Speed(m/s)	1	1	377.435057	0.0000
Det Capab	1	1	32.5241107	0.0000
Det Reset Time(sec)	1	1	1.3537723	0.2446
No of Robots*Sensor Range(m)	1	1	11.7558749	0.0006
No of Robots*Speed(m/s)	1	1	71.477179	0.0000
No of Robots*Det Capab	1	1	44.5368248	0.0000
No of Robots*Det Reset Time(sec)	1	1	26.2626556	0.0000
Sensor Range(m)*Speed(m/s)	1	1	35.783768	0.0000
Sensor Range(m)*Det Capab	1	1	62.3701127	0.0000
Sensor Range(m)*Det Reset Time(sec)	1	1	76.2448241	0.0000
Speed(m/s)*Det Capab	1	1	17.9159998	0.0000
Speed(m/s)*Det Reset Time(sec)	1	1	51.6682806	0.0000
Det Capab*Det Reset Time(sec)	1	1	6.3114926	0.0120
No of Robots*No of Robots	1	1	6.03662282	0.0140
Sensor Range(m)*Sensor Range(m)	1	1	24.0528627	0.0000
Det Capab*Det Capab	1	1	28.6551384	0.0000
Det Reset Time(sec)*Det Reset Time(sec)	1	1	6.03785929	0.0140

Figure 70. Logistic regression of MOE 2 without noise factors for non-pheromone robots

Actual by Predicted Plot



Summary of Fit

RSquare	0.80556
RSquare Adj	0.80483
Root Mean Square Error	4.655037
Mean of Response	15.43701
Observations (or Sum Wgts)	9900

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	37	885365.8	23928.8	1104.269
Error	9862	213703.3	21.7	Prob > F
C. Total	9899	1099069.2		0.0000

Parameter Estimates

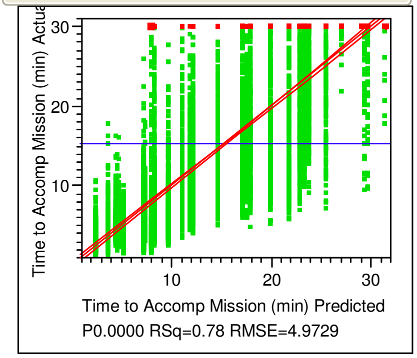
Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	26.568224	0.759206	34.99	<.0001
No of Robots	-0.119816	0.000907	-132	0.0000
Sensor Range(m)	-1.415914	0.018358	-77.13	0.0000
Speed(m/s)	-8.511497	0.084404	-100.8	0.0000
Det Capab	0.1423648	0.007798	18.26	<.0001
Det Reset Time(sec)	-0.036452	0.017141	-2.13	0.0335
Repel Friends	-0.004215	0.005027	-0.84	0.4018
Repel Cover	-0.055508	0.00555	-10.00	<.0001
Precision Movt	-0.015754	0.000805	-19.57	<.0001
Terrain{0&1-2}	-0.961914	0.049623	-19.38	<.0001
Terrain{0-1}	-0.626704	0.0573	-10.94	<.0001
(No of Robots-110.061)*(Speed(m/s)-1.05152)	-0.342027	0.062484	-5.47	<.0001
(No of Robots-110.061)*(Det Capab-86.1212)	-0.072291	0.00357	-20.25	<.0001
(No of Robots-110.061)*(Repel Friends+42.5152)	0.0403751	0.004009	10.07	<.0001
(No of Robots-110.061)*(Repel Cover+30.1212)	-0.014354	0.000919	-15.62	<.0001
(No of Robots-110.061)*(Terrain{0&1-2}-0.33333)	0.0068972	0.000927	7.44	<.0001
(No of Robots-110.061)*Terrain{0-1}	0.0035203	0.00107	3.29	0.0010
(Sensor Range(m)-5.25758)*(Det Capab-86.1212)	-1.397037	0.10405	-13.43	<.0001
(Sensor Range(m)-5.25758)*(Repel Cover+30.1212)	0.0611827	0.0052	11.77	<.0001
(Speed(m/s)-1.05152)*(Det Reset Time(sec)-5.05152)	6.414086	0.487048	13.17	<.0001
(Speed(m/s)-1.05152)*(Repel Cover+30.1212)	-1.46475	0.078219	-18.73	<.0001
(Speed(m/s)-1.05152)*(Precision Movt-200.121)	-0.144529	0.011449	-12.62	<.0001
(Speed(m/s)-1.05152)*Terrain{0-1}	-0.360007	0.10129	-3.55	0.0004
(Det Capab-86.1212)*(Det Reset Time(sec)-5.05152)	0.2761391	0.033992	8.12	<.0001
(Det Capab-86.1212)*(Terrain{0&1-2}-0.33333)	0.04333	0.006956	6.23	<.0001
(Det Capab-86.1212)*Terrain{0-1}	0.016927	0.008032	2.11	0.0351
(Det Reset Time(sec)-5.05152)*(Repel Cover+30.1212)	-0.265418	0.019033	-13.94	<.0001
(Det Reset Time(sec)-5.05152)*(Precision Movt-200.121)	-0.022081	0.001125	-19.64	<.0001
(Repel Friends+42.5152)*(Terrain{0&1-2}-0.33333)	0.0170581	0.004738	3.60	0.0003
(Repel Friends+42.5152)*Terrain{0-1}	0.0142564	0.005474	2.60	0.0092
(Repel Cover+30.1212)*Terrain{0-1}	0.0097091	0.004809	2.02	0.0435
(Precision Movt-200.121)*Terrain{0-1}	0.0019233	0.000963	2.00	0.0458
(Sensor Range(m)-5.25758)*(Sensor Range(m)-5.25758)	-1.484502	0.142586	-10.41	<.0001
(Speed(m/s)-1.05152)*(Speed(m/s)-1.05152)	43.909967	2.930587	14.98	<.0001
(Det Capab-86.1212)*(Det Capab-86.1212)	0.2819723	0.015814	17.83	<.0001
(Det Reset Time(sec)-5.05152)*(Det Reset Time(sec)-5.05152)	0.6939603	0.193006	3.60	0.0003
(Repel Cover+30.1212)*(Repel Cover+30.1212)	0.0688716	0.004789	14.38	<.0001
(Precision Movt-200.121)*(Precision Movt-200.121)	-0.006817	0.000626	-10.88	<.0001

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
No of Robots	1	1	377842.43	17436.7	0.0000
Sensor Range(m)	1	1	128904.63	5948.702	0.0000
Speed(m/s)	1	1	220359.19	10169.15	0.0000
Det Capab	1	1	7222.38	333.2992	<.0001
Det Reset Time(sec)	1	1	98.00	4.5225	0.0335
Repel Friends	1	1	15.23	0.7029	0.4018
Repel Cover	1	1	2167.21	100.0126	<.0001
Precision Movt	1	1	8299.76	383.0180	<.0001
Terrain{0&1-2}	1	1	8142.45	375.7582	<.0001
Terrain{0-1}	1	1	2592.20	119.6252	<.0001
No of Robots*Speed(m/s)	1	1	649.27	29.9628	<.0001
No of Robots*Det Capab	1	1	8887.77	410.1535	<.0001
No of Robots*Repel Friends	1	1	2198.05	101.4357	<.0001
No of Robots*Repel Cover	1	1	5290.21	244.1329	<.0001
No of Robots*Terrain{0&1-2}	1	1	1199.86	55.3712	<.0001
No of Robots*Terrain{0-1}	1	1	234.35	10.8150	0.0010
Sensor Range(m)*Det Capab	1	1	3906.38	180.2718	<.0001
Sensor Range(m)*Repel Cover	1	1	3000.12	138.4496	<.0001
Speed(m/s)*Det Reset Time(sec)	1	1	3758.14	173.4311	<.0001
Speed(m/s)*Repel Cover	1	1	7598.89	350.6743	<.0001
Speed(m/s)*Precision Movt	1	1	3452.98	159.3486	<.0001
Speed(m/s)*Terrain{0-1}	1	1	273.74	12.6325	0.0004
Det Capab*Det Reset Time(sec)	1	1	1430.05	65.9939	<.0001
Det Capab*Terrain{0&1-2}	1	1	840.85	38.8036	<.0001
Det Capab*Terrain{0-1}	1	1	96.24	4.4412	0.0351
Det Reset Time(sec)*Repel Cover	1	1	4213.88	194.4625	<.0001
Det Reset Time(sec)*Precision Movt	1	1	8355.46	385.5883	<.0001
Repel Friends*Terrain{0&1-2}	1	1	280.85	12.9606	0.0003
Repel Friends*Terrain{0-1}	1	1	147.00	6.7838	0.0092
Repel Cover*Terrain{0-1}	1	1	88.33	4.0764	0.0435
Precision Movt*Terrain{0-1}	1	1	86.48	3.9910	0.0458
Sensor Range(m)*Sensor Range(m)	1	1	2348.84	108.3945	<.0001
Speed(m/s)*Speed(m/s)	1	1	4864.78	224.5003	<.0001
Det Capab*Det Capab	1	1	6889.23	317.9246	<.0001
Det Reset Time(sec)*Det Reset Time(sec)	1	1	280.14	12.9280	0.0003
Repel Cover*Repel Cover	1	1	4481.69	206.8215	<.0001
Precision Movt*Precision Movt	1	1	2566.01	118.4166	<.0001

Figure 71. Regression of MOE 1 with all factors for non-pheromone robots

Actual by Predicted Plot



Summary of Fit

RSquare	0.777691
RSquare Adj	0.777264
Root Mean Square Error	4.972932
Mean of Response	15.43701
Observations (or Sum Wgts)	9900

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	19	854736.3	44986.1	1819.087
Error	9880	244332.9	24.7	Prob > F
C. Total	9899	1099069.2		0.0000

Parameter Estimates

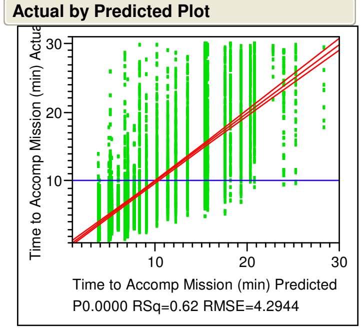
Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	31.023932	0.689826	44.97	0.0000
No of Robots	-0.117495	0.000937	-125.4	0.0000
Sensor Range(m)	-1.335113	0.018089	-73.81	0.0000
Speed(m/s)	-8.72699	0.088503	-98.61	0.0000
Det Capab	0.0725663	0.007019	10.34	<.0001
Det Reset Time(sec)	0.0507729	0.017023	2.98	0.0029
(No of Robots-110.061)*(Sensor Range(m)-5.25758)	0.0171814	0.001521	11.30	<.0001
(No of Robots-110.061)*(Speed(m/s)-1.05152)	-0.208137	0.028979	-7.18	<.0001
(No of Robots-110.061)*(Det Capab-86.1212)	-0.014619	0.001197	-12.22	<.0001
(No of Robots-110.061)*(Det Reset Time(sec)-5.05152)	0.013842	0.001896	7.30	<.0001
(Sensor Range(m)-5.25758)*(Speed(m/s)-1.05152)	1.7651416	0.158581	11.13	<.0001
(Sensor Range(m)-5.25758)*(Det Capab-86.1212)	0.1501486	0.011612	12.93	<.0001
(Sensor Range(m)-5.25758)*(Det Reset Time(sec)-5.05152)	0.6283349	0.07445	8.44	<.0001
(Speed(m/s)-1.05152)*(Det Capab-86.1212)	-1.317812	0.109899	-11.99	<.0001
(Speed(m/s)-1.05152)*(Det Reset Time(sec)-5.05152)	-2.196836	0.141421	-15.53	<.0001
(Det Capab-86.1212)*(Det Reset Time(sec)-5.05152)	-0.214056	0.011854	-18.06	<.0001
(Sensor Range(m)-5.25758)*(Sensor Range(m)-5.25758)	-0.249375	0.053558	-4.66	<.0001
(Speed(m/s)-1.05152)*(Speed(m/s)-1.05152)	17.895694	0.891499	20.07	<.0001
(Det Capab-86.1212)*(Det Capab-86.1212)	0.1884432	0.008612	21.88	<.0001
(Det Reset Time(sec)-5.05152)*(Det Reset Time(sec)-5.05152)	0.727452	0.040055	-18.16	<.0001

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
No of Robots	1	1	389042.94	15731.59	0.0000
Sensor Range(m)	1	1	134720.81	5447.656	0.0000
Speed(m/s)	1	1	240458.55	9723.334	0.0000
Det Capab	1	1	2643.02	106.8748	<.0001
Det Reset Time(sec)	1	1	219.99	8.8956	0.0029
No of Robots*Sensor Range(m)	1	1	3155.97	127.6167	<.0001
No of Robots*Speed(m/s)	1	1	1275.75	51.5872	<.0001
No of Robots*Det Capab	1	1	3690.83	149.2448	<.0001
No of Robots*Det Reset Time(sec)	1	1	1318.72	53.3245	<.0001
Sensor Range(m)*Speed(m/s)	1	1	3063.94	123.8953	<.0001
Sensor Range(m)*Det Capab	1	1	4134.59	167.1890	<.0001
Sensor Range(m)*Det Reset Time(sec)	1	1	1761.49	71.2289	<.0001
Speed(m/s)*Det Capab	1	1	3555.89	143.7881	<.0001
Speed(m/s)*Det Reset Time(sec)	1	1	5967.48	241.3046	<.0001
Det Capab*Det Reset Time(sec)	1	1	8064.17	326.0879	<.0001
Sensor Range(m)*Sensor Range(m)	1	1	536.14	21.6795	<.0001
Speed(m/s)*Speed(m/s)	1	1	9965.08	402.9542	<.0001
Det Capab*Det Capab	1	1	11842.03	478.8517	<.0001
Det Reset Time(sec)*Det Reset Time(sec)	1	1	8156.73	329.8305	<.0001

Figure 72. Regression of MOE 1 without noise factors for non-pheromone robots

Actual by Predicted Plot



Summary of Fit

RSquare	0.618952
RSquare Adj	0.618272
Root Mean Square Error	4.294442
Mean of Response	10.26101
Observations (or Sum Wgts)	7304

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	13	218382.46	16798.7	910.8794
Error	7290	134443.88	18.4	Prob > F
C. Total	7303	352826.34		0.0000

Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
Intercept	26.396328	1.203779	21.93	<.0001
No of Robots	-0.065682	0.002707	-24.26	<.0001
Sensor Range(m)	-0.963412	0.032116	-30.00	<.0001
Speed(m/s)	-4.647512	0.221254	-21.01	<.0001
Det Capab	-0.036194	0.013054	-2.77	0.0056
Det Reset Time(sec)	0.1122617	0.024847	4.52	<.0001
(No of Robots-126.004)*(Det Reset Time(sec)-5.0376)	0.002669	0.000723	3.69	0.0002
(Sensor Range(m)-5.71985)*(Speed(m/s)-1.16514)	0.5007533	0.080371	6.23	<.0001
(Speed(m/s)-1.16514)*(Det Capab-85.3449)	-0.409879	0.057711	-7.10	<.0001
(No of Robots-126.004)*(No of Robots-126.004)	0.0008048	0.000089	9.08	<.0001
(Sensor Range(m)-5.71985)*(Sensor Range(m)-5.71985)	0.114081	0.028055	4.07	<.0001
(Speed(m/s)-1.16514)*(Speed(m/s)-1.16514)	11.16551	0.770526	14.49	<.0001
(Det Capab-85.3449)*(Det Capab-85.3449)	0.0320816	0.002529	12.68	<.0001
(Det Reset Time(sec)-5.0376)*(Det Reset Time(sec)-5.0376)	0.222202	0.029826	-7.45	<.0001

Effect Tests

Source	Nparm	DF	Sum of Squares	F Ratio	Prob > F
No of Robots	1	1	10857.346	588.7218	<.0001
Sensor Range(m)	1	1	16595.806	899.8805	<.0001
Speed(m/s)	1	1	8137.155	441.2239	<.0001
Det Capab	1	1	141.772	7.6874	0.0056
Det Reset Time(sec)	1	1	376.473	20.4136	<.0001
No of Robots*Det Reset Time(sec)	1	1	251.337	13.6284	0.0002
Sensor Range(m)*Speed(m/s)	1	1	715.918	38.8195	<.0001
Speed(m/s)*Det Capab	1	1	930.284	50.4431	<.0001
No of Robots*No of Robots	1	1	1519.470	82.3908	<.0001
Sensor Range(m)*Sensor Range(m)	1	1	304.939	16.5348	<.0001
Speed(m/s)*Speed(m/s)	1	1	3872.538	209.9820	<.0001
Det Capab*Det Capab	1	1	2966.695	160.8642	<.0001
Det Reset Time(sec)*Det Reset Time(sec)	1	1	1023.599	55.5030	<.0001

Figure 73. Conditional regression of MOE 1 without noise factors for non-pheromone robots

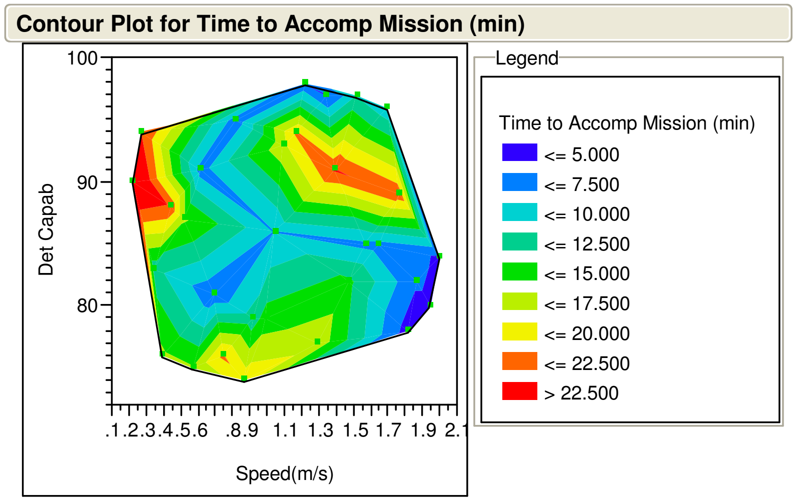


Figure 74. Contour plot for Detector Capability vs Speed

Figure 74 shows a large variance in performance. Only with a large speed (>1.2m/s) and a low det capab of less than <86 (=2 sec TOT), are we confident of achieving mission completion in a short time.

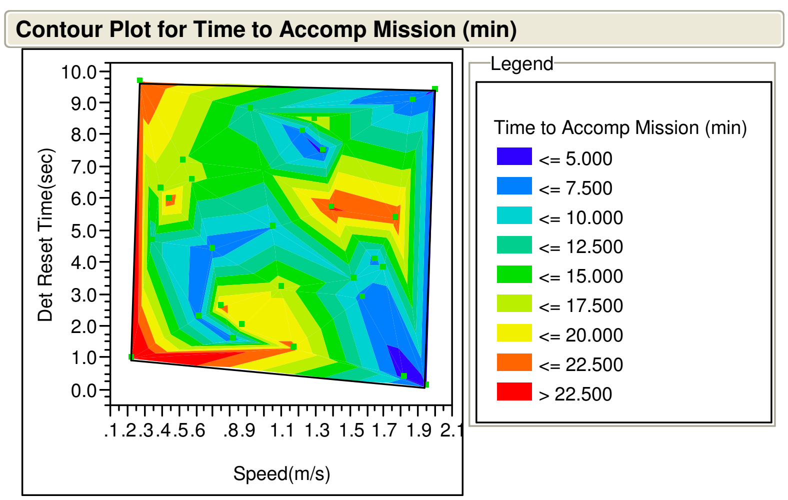


Figure 75. Contour plot for Detector Reset Time vs Speed

Figure 75 shows that a combination of high speed (>1.2m/s) and low detector reset time (<3sec) is necessary to boost the chances of a short mission completion time of the swarm.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. CLUSTERING AND OUTLIER ANALYSIS

A. INTRODUCTION

This appendix is a supplement to the statistical analysis done in the main section of the thesis, using the software package, Clustering & Outlier Analysis Data Mining Tool (COADM). It is written by DSO National Laboratories in Singapore. The bulk of the analysis is performed by the second reader of this thesis, Mr Choo Chwee Seng.

B. OBJECTIVE

The motivation behind the following analysis is to complement the statistical analysis and provide additional insights using COADM. This analysis allows the user to have a quick overview of the "good" and "bad" clusters, as well as outliers within each cluster, grouped according to the MOEs. This translates to the user having not only the ability to quickly identify the general trends of factor settings that attribute to good and bad performances in terms of MOEs, but also the ability to zoom down on the "bad" outliers in "good" clusters, input the parameters back into the simulation and examine what went wrong in those runs.

C. CORRELATION PLOTS FOR NON-PHEROMONE ROBOTS

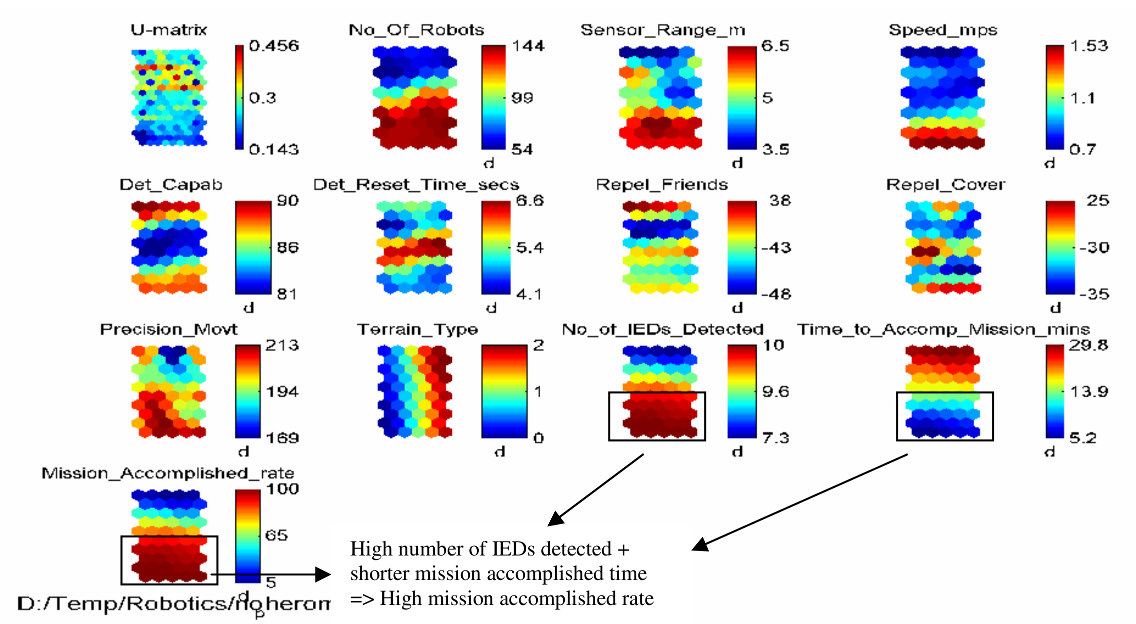


Figure 76. Overview of correlation plots for non-pheromone robots

Figure 76 shows the overview of the factor settings in the DOE as well as the distribution of the MOEs over all runs. It is expected that there is little correlation between the input factors, which can be observed from their different colored patterns from one another. The MOEs show high correlation with each another, which is also expected as a high mission accomplishment rate translates to a high number of IEDs detected and a short time to accomplish mission.

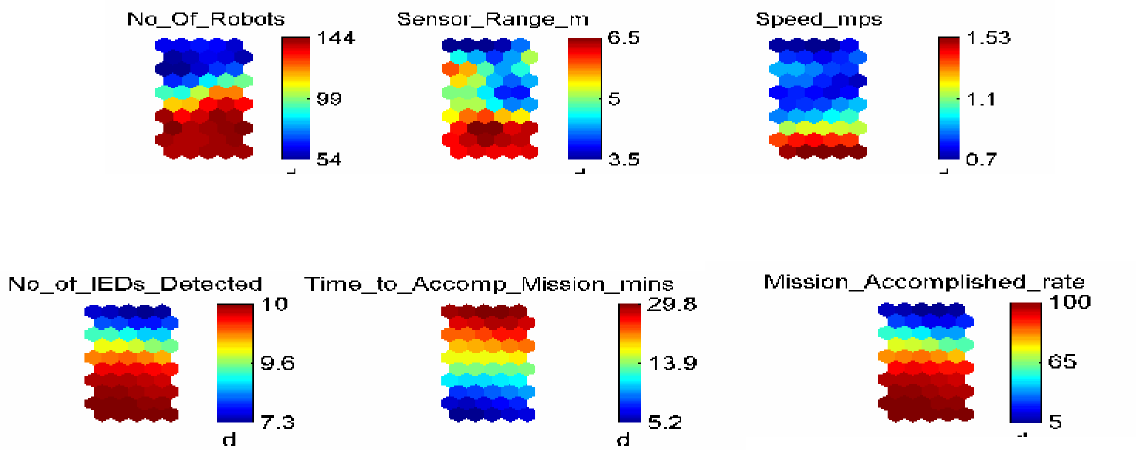


Figure 77. High correlation between factors and MOEs for non-pheromone robots

Figure 77 singles out the three factors that are observed to have high correlation with the MOEs, i.e., their patterns vary closely with the MOEs. The factors identified are number of robots, sensor range and speed of robot, and are consistent with the findings from the statistical analysis that these factors form the main effects. The other factors have no obvious trends or patterns and have no significant impact on the MOEs. This is again consistent with previous findings from statistical analysis.

D. CLUSTERING ANALYSIS FOR NON-PHEROMONE ROBOTS

With consideration given to all three MOEs, the clustering tool generates a total of eight clusters, represented by the different colors as show in the following figure. The figures in parentheses represent the number of design points in each cluster while the black hexagons are the outliers within the cluster. Outliers are found in Cluster 4 and 8 in this case as shown in Figure 78.

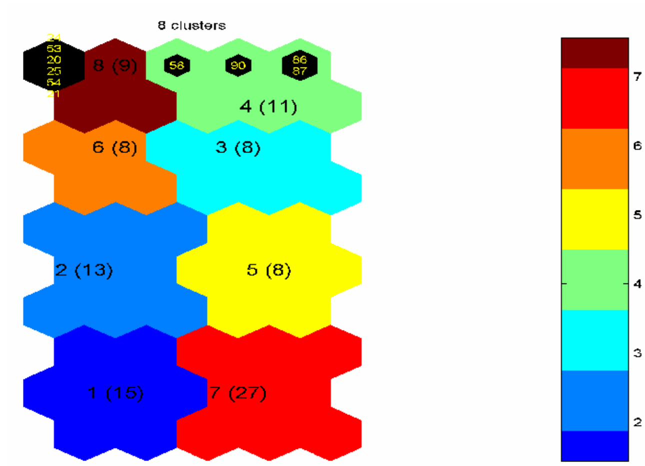


Figure 78. Clustering analysis for case of non-pheromone robots

The cluster table shown in Table 6 summarizes the range of each factor and MOE that each cluster represents. It can be observed that Cluster 1 and 7 are the best clusters in terms of overall performance of the three MOEs, while Cluster 4 and 8 are the worst clusters. The cluster table provides some insight on the "optimal" settings for the non-pheromone robots.

CLUSTER: 1 1: No_of_IEDs_Detected 9.990 +/- 0.011 2: Time_to_Accomp_Mission_mins 5.748 +/- 1.569 3: Mission_Accomplished_rate 99.000 +/- 1.121 4: Speed_mps 1.350 +/- 0.465 5: No_Of_Robots 144.000 +/- 32.570 6: Sensor_Range_m 6.250 +/- 2.093 7: Precision_Movt 213.000 +/- 67.787 8: Det_Capab 87.000 +/- 7.103 9: Repel_Friends -43.000 +/- 10.990 10: Repel_Cover -30.000 +/- 12.578 11: Det_Reset_Time_secs 4.700 +/- 3.177 12: Terrain_Type 0.000 +/- 0.258	CLUSTER: 2 1: No_of_IEDs_Detected 9.940 +/- 0.063 2: Time_to_Accomp_Mission_mins 12.586 +/- 4.016 3: Mission_Accomplished_rate 94.000 +/- 6.036 4: Det_Reset_Time_secs 6.000 +/- 1.165 5: Repel_Cover -28.000 +/- 12.193 6: Precision_Movt 206.000 +/- 48.685 7: No_Of_Robots 116.000 +/- 51.903 8: Sensor_Range_m 5.250 +/- 3.454 9: Repel_Friends -43.000 +/- 12.141 10: Speed_mps 0.875 +/- 0.403 11: Terrain_Type 0.000 +/- 0.506 12: Det_Capab 81.000 +/- 4.778	CLUSTER: 3 1: No_of_IEDs_Detected 9.470 +/- 0.203 2: Time_to_Accomp_Mission_mins 24.090 +/- 2.109 3: Mission_Accomplished_rate 60.000 +/- 12.270 4: Terrain_Type 2.000 +/- 0.463 5: Det_Reset_Time_secs 5.100 +/- 2.695 6: Precision_Movt 188.000 +/- 42.576 7: Sensor_Range_m 4.250 +/- 1.863 8: No_Of_Robots 76.000 +/- 36.249 9: Repel_Cover -33.000 +/- 13.169 10: Speed_mps 0.825 +/- 0.352 11: Det_Capab 82.000 +/- 5.410 12: Repel_Friends -47.000 +/- 11.463
CLUSTER: 4 1: No_of_IEDs_Detected 7.830 +/- 2.592 2: Time_to_Accomp_Mission_mins 29.713 +/- 1.604 3: Mission_Accomplished_rate 8.000 +/- 16.608 4: Terrain_Type 2.000 +/- 0.467 5: Det_Capab 88.000 +/- 1.954 6: Repel_Friends -41.000 +/- 8.029 7: Repel_Cover -30.000 +/- 10.304 8: Det_Reset_Time_secs 5.100 +/- 3.748 9: Precision_Movt 181.000 +/- 73.826 10: Sensor_Range_m 4.000 +/- 3.174 11: Speed_mps 0.750 +/- 0.641 12: No_Of_Robots 59.000 +/- 33.833	CLUSTER: 5 1: No_of_IEDs_Detected 9.930 +/- 0.071 2: Time_to_Accomp_Mission_mins 13.589 +/- 3.448 3: Mission_Accomplished_rate 93.000 +/- 6.719 4: Det_Reset_Time_secs 6.000 +/- 1.219 5: No_Of_Robots 133.000 +/- 57.410 6: Terrain_Type 2.000 +/- 0.518 7: Repel_Friends -41.000 +/- 6.739 8: Precision_Movt 200.000 +/- 36.597 9: Repel_Cover -30.000 +/- 11.801 10: Sensor_Range_m 4.750 +/- 2.542 11: Speed_mps 0.875 +/- 0.295 12: Det_Capab 83.000 +/- 4.027	CLUSTER: 6 1: No_of_IEDs_Detected 9.600 +/- 0.192 2: Time_to_Accomp_Mission_mins 21.866 +/- 2.283 3: Mission_Accomplished_rate 65.000 +/- 12.048 4: Sensor_Range_m 5.500 +/- 2.967 5: Precision_Movt 194.000 +/- 21.889 6: Det_Capab 84.000 +/- 9.812 7: Speed_mps 0.875 +/- 0.189 8: Repel_Cover -33.000 +/- 7.396 9: Terrain_Type 0.000 +/- 0.518 10: Det_Reset_Time_secs 4.400 +/- 2.338 11: No_Of_Robots 59.000 +/- 15.937 12: Repel_Friends -47.000 +/- 5.398
CLUSTER: 7 1: No_of_IEDs_Detected 9.990 +/- 0.027 2: Time_to_Accomp_Mission_mins 6.655 +/- 2.323 3: Mission_Accomplished_rate 99.000 +/- 2.674 4: Speed_mps 1.350 +/- 0.497 5: No_Of_Robots 144.000 +/- 33.026 6: Sensor_Range_m 6.500 +/- 2.264 7: Terrain_Type 1.000 +/- 0.509 8: Det_Capab 88.000 +/- 7.136 9: Precision_Movt 206.000 +/- 63.535 10: Repel_Friends -44.000 +/- 11.240 11: Repel_Cover -30.000 +/- 12.791 12: Det_Reset_Time_secs 4.400 +/- 2.904	CLUSTER: 8 1: No_of_IEDs_Detected 8.080 +/- 3.503 2: Time_to_Accomp_Mission_mins 29.500 +/- 0.977 3: Mission_Accomplished_rate 17.000 +/- 10.404 4: Det_Capab 90.000 +/- 2.351 5: Repel_Friends -40.000 +/- 7.546 6: Precision_Movt 200.000 +/- 82.517 7: Det_Reset_Time_secs 4.700 +/- 3.901 8: Repel_Cover -31.000 +/- 10.686 9: Terrain_Type 1.000 +/- 0.500 10: Sensor_Range_m 4.000 +/- 2.268 11: Speed_mps 0.750 +/- 0.742 12: No_Of_Robots 59.000 +/- 31.548	

Table 6. Cluster table for non-pheromone robots

As shown earlier, the outliers fall in Cluster 4 and 8, which Table 6 reveals are the worst clusters. Closer examination of the individual outliers reveals that these outliers perform worse than the mean of their respective clusters, i.e., “bad” outliers in “bad” clusters, which is of low value for our analysis. However, it is both reassuring and important to note that there are no “bad” outliers in the “good” clusters, which implies that the robot swarm should maintain its effectiveness in most, if not all circumstances, if the factor settings are set appropriately.

E. CORRELATION PLOTS FOR PHEROMONE ROBOTS

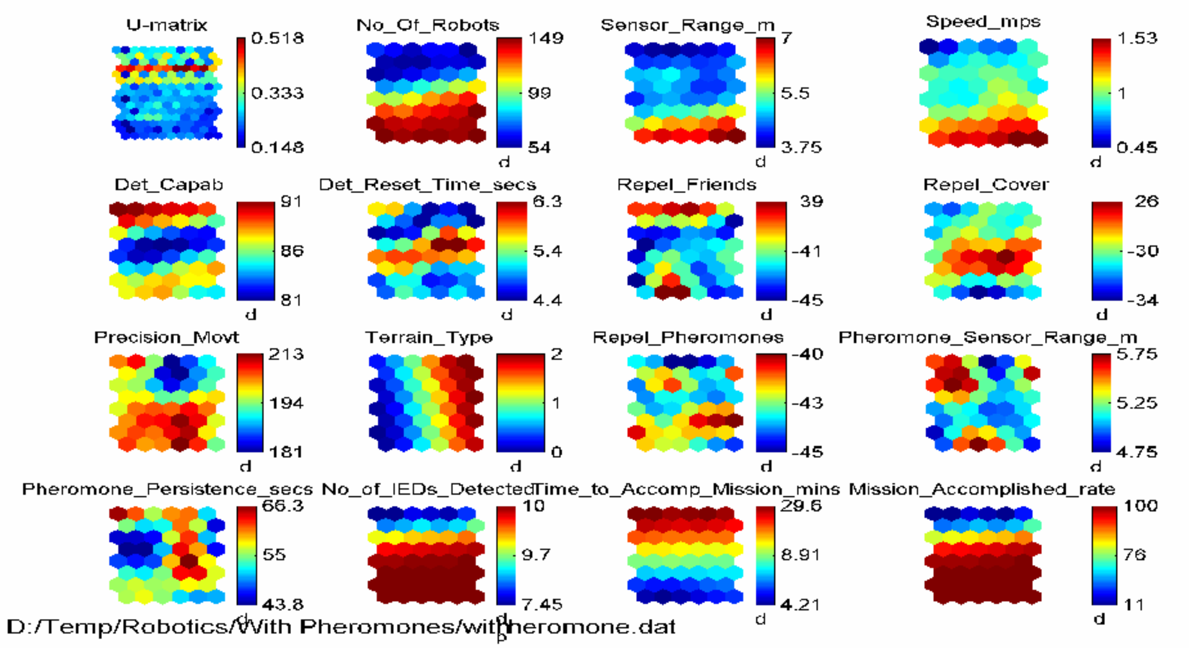


Figure 79. Overview of correlation plots for pheromone robots

Similar to the non-pheromone robots case, the overview of the correlation plots for pheromone robots are provided in Figure 79. Again, no correlation shows up between input factors, but high correlation shows up between the MOEs, as expected.

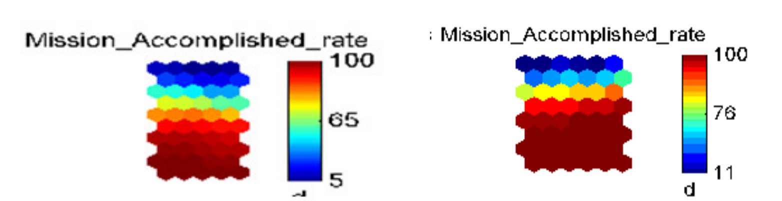


Figure 80. Comparison of mission accomplishment rate between non-pheromone (left) and pheromone robots (right)

A comparison between of the mission accomplishment rate for non-pheromone robots and pheromone robots made in Figure 80 reveals that there is more "redness" in the plot for the pheromone robots. This translates to more design points accomplishing the mission in the case of the pheromone robots, i.e., pheromone robots perform better than non-pheromone robots in terms of mission accomplishment. This is consistent with the findings from the statistical analyses in Chapter V.

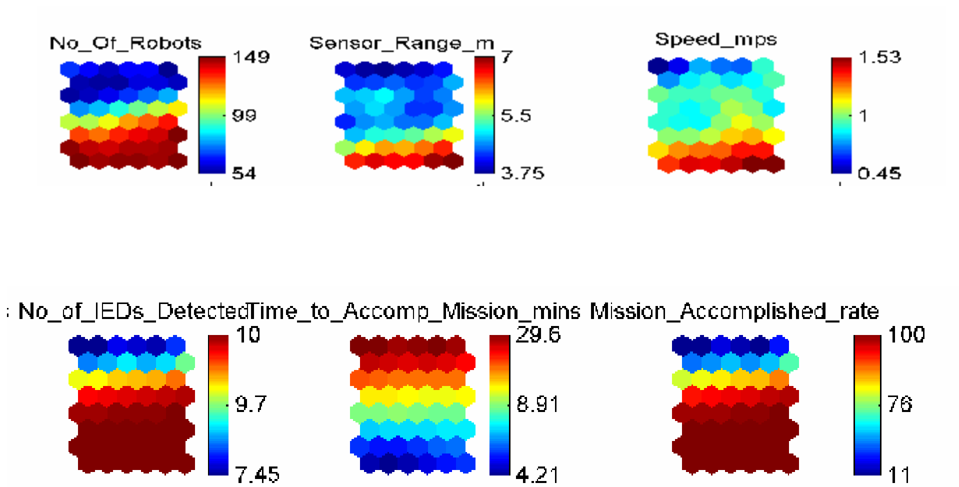


Figure 81. High correlation between factors and MOEs for pheromone robots

Figure 81 tells the same story as the non-pheromone robots case, that the number of robots, sensor range and speed, are highly correlated with the MOEs and hence have a significant impact on the MOEs. With the introduction of pheromone capability, the main factors that determine the outcome remain unchanged. In addition, it is observed that there is no obvious correlation for other factors with the MOEs.

F. CLUSTERING ANALYSIS FOR PHEROMONE ROBOTS

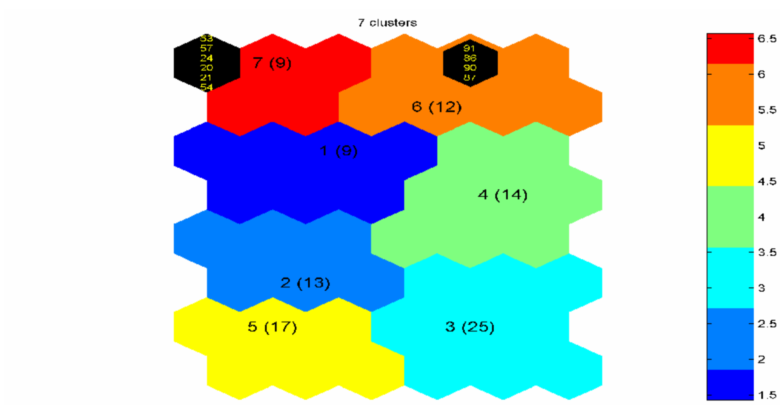


Figure 82. Clustering analysis for case of pheromone robots

COADM generates a total of 7 clusters for the case of the pheromone robots when considering all three MOEs as shown in Figure 82. The outliers found are in Cluster 6 and Cluster 7.

CLUSTER: 1	CLUSTER: 2	CLUSTER: 3
1: No_of_IEDs_Detected 9.910 +/- 0.080	1: No_of_IEDs_Detected 10.000 +/- 0.000	1: No_of_IEDs_Detected 10.000 +/- 0.000
2: Time_to_Accomp_Mission_mins 13.815 +/- 3.573	2: Time_to_Accomp_Mission_mins 6.710 +/- 2.733	2: Time_to_Accomp_Mission_mins 5.072 +/- 1.466
3: Mission_Accomplished_rate 91.000 +/- 8.043	3: Mission_Accomplished_rate 100.000 +/- 0.000	3: Mission_Accomplished_rate 100.000 +/- 0.000
4: Pheromone_Sensor_Range_m 5.500 +/- 2.545	4: Det_Reset_Time_secs 5.700 +/- 1.140	4: No_Of_Robots 144.000 +/- 32.866
5: Repel_Cover -30.000 +/- 5.094	5: Repel_Cover -29.000 +/- 14.754	5: Sensor_Range_m 6.250 +/- 2.413
6: Repel_Pheromones -43.000 +/- 13.426	6: Precision_Movt 206.000 +/- 48.677	6: Speed_mps 1.350 +/- 0.560
7: Speed_mps 0.925 +/- 0.381	7: No_Of_Robots 116.000 +/- 46.255	7: Terrain_Type 1.000 +/- 0.507
8: Precision_Movt 194.000 +/- 28.697	8: Repel_Pheromones -43.000 +/- 8.921	8: Precision_Movt 206.000 +/- 67.996
9: Sensor_Range_m 5.000 +/- 3.446	9: Speed_mps 0.925 +/- 0.359	9: Repel_Pheromones -43.000 +/- 9.742
10: Det_Reset_Time_secs 5.100 +/- 1.645	10: Repel_Friends -43.000 +/- 10.419	10: Det_Capab 87.000 +/- 7.023
11: Terrain_Type 1.000 +/- 0.527	11: Sensor_Range_m 5.000 +/- 2.573	11: Pheromone_Persistence_secs 57.800 +/- 29.490
12: Det_Capab 84.000 +/- 9.121	12: Pheromone_Sensor_Range_m 5.000 +/- 1.951	12: Repel_Cover -30.000 +/- 13.085
13: No_Of_Robots 71.000 +/- 20.606	13: Pheromone_Persistence_secs 52.200 +/- 30.035	13: Pheromone_Sensor_Range_m 5.250 +/- 2.986
14: Repel_Friends -44.000 +/- 12.228	14: Det_Capab 83.000 +/- 5.251	14: Repel_Friends -44.000 +/- 10.480
15: Pheromone_Persistence_secs 49.400 +/- 12.432	15: Terrain_Type 0.000 +/- 0.480	15: Det_Reset_Time_secs 4.700 +/- 3.005
CLUSTER: 4	CLUSTER: 5	CLUSTER: 6
1: No_of_IEDs_Detected 9.980 +/- 0.009	1: No_of_IEDs_Detected 10.000 +/- 0.000	1: No_of_IEDs_Detected 9.210 +/- 2.570
2: Time_to_Accomp_Mission_mins 11.037 +/- 3.826	2: Time_to_Accomp_Mission_mins 4.367 +/- 1.210	2: Time_to_Accomp_Mission_mins 26.839 +/- 3.521
3: Mission_Accomplished_rate 98.000 +/- 0.929	3: Mission_Accomplished_rate 100.000 +/- 0.000	3: Mission_Accomplished_rate 40.000 +/- 30.696
4: Det_Reset_Time_secs 5.700 +/- 1.809	4: No_Of_Robots 144.000 +/- 33.645	4: Terrain_Type 2.000 +/- 0.452
5: Terrain_Type 2.000 +/- 0.469	5: Sensor_Range_m 6.250 +/- 2.173	5: Det_Capab 88.000 +/- 4.896
6: Repel_Cover -29.000 +/- 13.352	6: Speed_mps 1.300 +/- 0.494	6: Repel_Friends -41.000 +/- 9.434
7: Pheromone_Persistence_secs 60.600 +/- 29.591	7: Precision_Movt 206.000 +/- 62.578	7: Pheromone_Persistence_secs 57.800 +/- 21.658
8: No_Of_Robots 104.000 +/- 59.769	8: Repel_Pheromones -41.000 +/- 9.584	8: Repel_Cover -30.000 +/- 10.189
9: Precision_Movt 200.000 +/- 39.815	9: Repel_Friends -41.000 +/- 12.063	9: Pheromone_Sensor_Range_m 5.250 +/- 3.333
10: Speed_mps 1.000 +/- 0.361	10: Det_Capab 87.000 +/- 7.361	10: Repel_Pheromones -43.000 +/- 13.256
11: Repel_Pheromones -43.000 +/- 11.605	11: Pheromone_Persistence_secs 57.800 +/- 28.573	11: Speed_mps 0.825 +/- 0.595
12: Repel_Friends -43.000 +/- 9.354	12: Pheromone_Sensor_Range_m 5.250 +/- 2.939	12: Sensor_Range_m 4.250 +/- 3.333
13: Pheromone_Sensor_Range_m 5.000 +/- 2.403	13: Det_Reset_Time_secs 5.100 +/- 2.973	13: Precision_Movt 188.000 +/- 69.635
14: Sensor_Range_m 4.750 +/- 2.365	14: Repel_Cover -31.000 +/- 12.683	14: Det_Reset_Time_secs 4.400 +/- 3.334
15: Det_Capab 83.000 +/- 4.438	15: Terrain_Type 0.000 +/- 0.393	15: No_Of_Robots 59.000 +/- 33.510
CLUSTER: 7		
1: No_of_IEDs_Detected 8.410 +/- 2.928		
2: Time_to_Accomp_Mission_mins 27.711 +/- 2.390		
3: Mission_Accomplished_rate 17.000 +/- 22.749		
4: Det_Capab 90.000 +/- 2.315		
5: Pheromone_Sensor_Range_m 5.750 +/- 4.037		
6: Repel_Friends -40.000 +/- 8.141		
7: Pheromone_Persistence_secs 57.800 +/- 21.551		
8: Precision_Movt 200.000 +/- 88.658		
9: Det_Reset_Time_secs 5.100 +/- 4.196		
10: Repel_Pheromones -43.000 +/- 10.829		
11: Repel_Cover -31.000 +/- 10.764		
12: Terrain_Type 1.000 +/- 0.527		
13: Speed_mps 0.625 +/- 0.630		
14: Sensor_Range_m 4.000 +/- 2.221		
15: No_Of_Robots 59.000 +/- 30.232		

Table 7. Cluster table for pheromone robots

The cluster table in Table 7 for the pheromone robots shows that, when compared to the non-pheromone robots cluster table, there are more "best" clusters (Cluster 2, 3 and 5) while there are fewer "worst" clusters (Cluster 7). Overall, the values of the MOEs of the clusters have improved as compared to the non-pheromone case, consistent with previous findings. Figure 82 shows that the outliers for the case of pheromone robots are found in Cluster 6 and 7, which are the two worst clusters among all. Further investigations reveal that these outliers are worse than the mean of their respective clusters. This is again the case of "bad" outliers in "bad" clusters, which is of not much value for our purpose.

G. CONCLUSION

COADM allows a swift analysis of the general behavior and trends of the factor inputs and MOEs. It also allows the user to quickly identify correlations within factor inputs and within MOEs, as well as between them. In addition, outliers are siphoned out right away and examined closely by re-entering the parameters and random seed back into the simulation to find out what went wrong. This is an extremely important aspect to be investigated in instances of "bad" outliers in "good" clusters. With closer examination of these outliers, insights can be gained out factor deviations or conditions that will cause a drastic impact on performance, especially when it is expected that the system will function well.

Overall, the COADM analysis is meant to complement the statistical analysis performed in the main section of this thesis. The findings gained from this clustering and outlier analysis **agrees** with those from the statistical analysis in Chapter V. It is also revealed that the system has no instances of "bad" outliers in "good" clusters, perhaps suggesting the robustness of the performance of the robot swarm when it operates at the settings for it is expected to do well.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Ayers, J., Davis, J. and Rudolph, A., "*Neurotechnology for Biomimetics*," MIT Press, 2000.

Babilot, M., "*Comparison of a Distributed Operations Force to a Traditional Force in Urban Combat*," M.S. Thesis, Operations Research Department, Naval Postgraduate School, Monterey, CA, Sep 2005.

Batalin, M. and Sukhatme, G., "*Spreading Out: A Local Approach to Multi-robot Coverage*," Proceedings of the 6th International Symposium on Distributed Autonomous Robotics Systems, pp 373.382, Fukuoka, Japan, Jun 25-27, 2002.

Beni, G., "*From Swarm Intelligence to Swarm Robotics*," SAB 2004 International Workshop, pp1-9, Santa Monica, CA, Jul 2004.

Bonabeau, E., Dorigo, M. and Theraulaz, G., "*Swarm Intelligence: From Natural to Artificial Systems*," Oxford University Press, 1999.

Bruemmer, D., Dudenhoeffer, D., McKay, M. and Anderson, M., "*A Robotic Swarm for Spill Finding and Perimeter Formation*," Spectrum, Aug 2002.

Cassinis, R., "*Landmines Detection Methods using Swarms of Simple Robots*," in: Pagello, E. et al. (Ed.), Intelligent Autonomous Systems 6, IOS Press, 2000.

Centibot project website. http://www.ai.sri.com/centibots/index.html. Last accessed Nov 2006.

Cioppa, T., "*Efficient Nearly Orthogonal and Space-filling Experimental Designs for High-dimensional Complex Models*," Ph.D. Dissertation, Operations Research Department, Naval Postgraduate School, Sep 2002.

Cioppa, T. M. and Lucas, T. W., "*Efficient Nearly Orthogonal and Space-filling Latin Hypercubes*," Technometrics, forthcoming, 2006.

CWRU Mini-Whegs. http://biorobots.cwru.edu/projects/whegs/miniwhegs.html. Last accessed Nov 2006.

D'Andrea, R., "*The Cornell RoboCup Robot Soccer Team: 1999-2003*," Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY, 2003.

Dickie, A. "*Modeling Robot Swarms Using Agent Based Simulation*," M.S. Thesis, Operations Research Department, Naval Postgraduate School, Monterey, CA, Jun 2002.

Dorigo, M., "*The SWARM-BOTS Project*," Swarm Robotics, SAB 2004 International Workshop, pp70-83, Santa Monica, CA, Jul 2004.

Dudenhoeffer, D. and Jones, M., "*A Formation Behavior for Large-Scale Micro-Robot Force Deployment*", in: Joines, J. A., Barton, R. R., Kang, K. and Fishwick, P. A. (Eds.), Proceedings of the 2000 Winter Simulation Conference, Piscataway, NJ: Institute of Electrical and Electronics Engineers, pp972-982, 2000.

FIDO detector integrated with Packbot news article. http://www.forbes.com/forbes/2005/0314/056_print.html. Last accessed Nov 2006.

FIDO XT Series Explosives Detector Brochure.

FCS website. http://www.globalsecurity.org/military/systems/ground/fcs.htm. Last accessed Nov 2006.

FY 2005 Joint Robotics Program Master Plan by OUSD (AT&L) Defense Systems/ Land Warfare and Munitions, 2005.

Gill, A. and Greiger, D., "*Comparison of Agent Based Distillation Movement Algorithms*," Military Operations Research, Vol 8 No 3, pp5-16, 2003.

Howard, A., Mataric, A. J., Sukhatme, G., "*Mobile Sensor Network Deployment using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem,*" Proceedings of the 6th International Symposium on Distributed Autonomous Robotic Systems, pp299-309, Fukuoka, Japan, Jun 2002.

Icosystems news article. http://www.newscientist.com/article.ns?id=dn3661. Last accessed Nov 2006.

ICx Nomadics Inc. Website (FIDO detector website). http://www.icxt.com/products/. Last accessed Nov 2006.

Interview with Col. Edward M. Ward, logistics chief of the Robotic Systems Joint Project Office. http://www.decaturdaily.com/decaturdaily/news/050927/robots.shtml. Last accessed Nov 2006.

iRobot integration with FIDO. http://robotstocknews.blogspot.com/2006/08/irobot-unveils-explosive-sniffing.html. Last accessed Nov 2006.

iRobot swarm robotics website. http://irobot.com/sp.cfm?pageid=149. Last accessed Nov 2006.

iRobot Wayfarer project website. http://www.robotfrontier.com/. Last accessed Nov 2006.

James McLurkin Personal Website. http://people.csail.mit.edu/jamesm/swarm.php. Last accessed Nov 2006.

Kratochvil, B., Burt, I. T., Drenner, A., Goerke, D., Jackson, B., McMillen, C., Olsen, C., Papanikolopoulos, N., Pfeifer, A., Stoeter, S. A., Stubbs, K. and Waletzko, D., "*Heterogeneous Implementation of an Adaptive Robotic Sensing Team*", Proceedings of the 2003 IEEE International Conference on Robotics and Automation, Taipei, Taiwan, Sep 2003.

Kennedy, J. and Eberhart, R., "*Swarm Intelligence*," Academic Press, 2001.

MANA Users Manual v3.0, Jul 2004

MARCBot news article. http://www.estripes.com/article.asp?section=104&article=30233&archive=true. Last accessed Nov 2006.

MARCBot manufacturer website. http://www.exponent.com//practices/techdev/marcbot.html. Last accessed Nov 2006.

MARCBot news article. http://www.defenseindustrydaily.com/2006/06/mighty-mites-marcbots-add-exponent-to-ied-landmine-detection-updated/index.php#more. Last accessed Nov 2006.

Morlok, R. and Gini, M., "*Dispersing robots in an unknown environment*," DARS 2004, Toulouse, France, Jun 2004.

National Defense Authorization Act for Fiscal Year 2001. H.R.4205, Sec. 217.

Operation Iraqi Freedom Statistics. http://icasualties.org/oif_a/CasualtyTrends.htm. Last accessed Nov 2006.

Payton, D., Estkowski, R., Howard, M., "*Pheromone Robotics and the Logic of Virtual Pheromones*," Swarm Robotics, SAB 2004 International Workshop, pp45-57, Santa Monica, CA, Jul 2004.

Rekleitis, I., Lee-Shue, V., New, A. P. and Choset, H., "*Limited Communication, Multi-Robot Team Based Coverage*," Proceedings of the 2004 IEEE International Conference on Robotics and Automation, New Orleans, LA, Apr 2004.

Sahin, E., "*Swarm Robotics: From Sources of Inspiration to Domains of Application*," Swarm Robotics, SAB 2004 International Workshop, Santa Monica, CA, Jul 2004.

Sanchez, S., "*Work Smarter, Not Harder: Guidelines for Designing Simulation Experiments*", in: Proceedings of the 2005 Winter Simulation Conference, Kuhl, M. E., Steiger, N. M., Armstrong, F. B. and Joines, J. A. (Eds), Piscataway, NJ: Institute of Electrical and Electronic Engineers, pp 69-82, 2005.

Sauter, J., Matthews, R., Van Dyke Parunak, H. and Brueckner, S., "*Performance of Digital Pheromones for Swarming Vehicle Control*," Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems, The Netherlands, 2005.

Schroer, R. T., Boggess, M. J., Bachmann, R. J., Quinn, R. D., and Ritzmann, R. E. "*Comparing Cockroach and Whegs Robot Body Motions*," IEEE Conference on Robotics and Automation, New Orleans, LA, Apr 2004.

Stormont, D., "*Modeling a Swarm of Search and Rescue Robots using Star Logo*," 7th Annual Swarm Users/ Researchers Conference (SWARMFEST 2003), Notre Dame, IN, Apr 2003.

Tarasewich, P. and McMullen, P., "*Swarm Intelligence: Power in Numbers*", Communications of the ACM, Vol 45, No 8, pp62-67, Aug 2002.

Van Dyke Parunak, H., Brueckner, S. and Sauter, J., "*Digital Pheromone Mechanisms for Coordination of Unmanned Vehicles*," International Conference on Autonomous Agents, Bologna, Italy, Jul 2002.

Wagner, I., Lindenbaum, M. and Bruckstein, A., "*Distributed Covering by Ant-Robots Using Evaporating Traces*," IEEE Transactions on Robotics and Automation, Vol 15, No 5, pp918-933, Oct 1999.

Yamauchi, B., "*Autonomous Urban Reconnaissance Using Man-Portable UGVs*," Proceedings of SPIE Vol. 6230: Unmanned Systems Technology VIII, Orlando, FL, Apr 2006.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Professor Susan M. Sanchez
   Naval Postgraduate School
   Monterey, California

4. Mr Choo Chwee Seng
   DSO National Laboratories
   Singapore

5. Professor Yeo Tat Soon
   Director, Temasek Defence Systems Institute (TDSI)
   National University of Singapore
   Singapore

6. Captain Sze-Tek Terence Ho
   Singapore Armed Forces
   Singapore