NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

STATE SPACE ANALYSIS OF THE GENERAL ANALYTIC MINEFIELD EVALUATION TOOL: IN PURSUIT OF THE ADVANCED UNDERSEA WEAPON SYSTEM

by

Brian C. Flick

March 2019

Thesis Advisor:	Thomas A. Hamrick
Co-Advisor:	Paul J. Sanchez
Second Reader:	Steven E. Pilnick

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 March 2019	3. REPORT TYPE AND DATES COVERED Master's thesis

4. TITLE AND SUBTITLE STATE SPACE ANALYSIS OF THE GENERAL ANALYTIC MINEFIELD EVALUATION TOOL: IN PURSUIT OF THE ADVANCED UNDERSEA WEAPON SYSTEM	5. FUNDING NUMBERS
6. AUTHOR(S) Brian C. Flick	
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)

Naval mine warfare planning tools will grow increasingly important as maritime mines grow more sophisticated, permitting their use in less restrictive environments and for a wider variety of missions. The purpose of this study is to explore the state space of General Analytical Minefield EvaluationTool (GAMET)in order to evaluate simulation functionality and identify significant factors and interactions relevant to a traditional minefield and advanced undersea warfare system (AUWS) design. A design of experiments (DOE) is applied using a nearly orthogonal, nearly balanced Latin hypercube (NOB). A survey of the simulation is made to identify inputs and response variables. Two scenarios are created with justification of factor level settings and DOE process. A Visual Basic for Applications (VBA) user tool is developed to assist in creating files required to run 512 design points (DP) for the simulation to process. During the course of the study, GAMET simulated over 100 million transits, processing over 51,000 unique DPs. The analysis confirms that GAMET produces results consistent with real world experience and can be used to explore various AUWS designs. Metamodels based on regression analysis can be satisfactory for predicting number of casualties and detonations per transitor. These findings will advance AUWS design, help improve GAMET, and support broader GAMET usage within the Mine Warfare community.

14. SUBJECT TERMS GAMET, MW, simulation, mining, AUWS, DOE	15. NUMBER OF PAGES 97
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

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.

STATE SPACE ANALYSIS OF THE GENERAL ANALYTIC MINEFIELD EVALUATION TOOL: IN PURSUIT OF THE ADVANCED UNDERSEA WEAPON SYSTEM

Brian C. Flick
Commander, United States Navy
BSEE, Virginia Tech, 1993
MBA, George Washington University, 2002
MA, Air University, 2005

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 2019

Approved by: Thomas A. Hamrick
Advisor

Paul J. Sanchez
Co-Advisor

Steven E. Pilnick
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Naval mine warfare planning tools will grow increasingly important as maritime mines grow more sophisticated, permitting their use in less restrictive environments and for a wider variety of missions. The purpose of this study is to explore the state space of General Analytical Minefield EvaluationTool (GAMET) in order to evaluate simulation functionality and identify significant factors and interactions relevant to a traditional minefield and advanced undersea warfare system (AUWS) design. A design of experiments (DOE) is applied using a nearly orthogonal, nearly balanced Latin hypercube (NOB). A survey of the simulation is made to identify inputs and response variables. Two scenarios are created with justification of factor level settings and DOE process. A Visual Basic for Applications (VBA) user tool is developed to assist in creating files required to run 512 design points (DP) for the simulation to process. During the course of the study, GAMET simulated over 100 million transits, processing over 51,000 unique DPs. The analysis confirms that GAMET produces results consistent with real world experience and can be used to explore various AUWS designs. Metamodels based on regression analysis can be satisfactory for predicting number of casualties and detonations per transitor. These findings will advance AUWS design, help improve GAMET, and support broader GAMET usage within the Mine Warfare community.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

1 Introduction **1**
1.1 Background and Motivation 1
1.2 Study Objectives 3
1.3 Benefits 4
1.4 Literature Review 4
1.5 Methodology 5

2 GAMET Overview **7**
2.1 Background 7
2.2 Inputs 9
2.3 Outputs 12
2.4 Simulation Execution Logic 12

3 Experiment Design and Implementation **17**
3.1 Case Study Selection and Construction 17
3.2 Selection of Response Variables 22
3.3 Design Methodology 22
3.4 VBA Construction 25
3.5 Running the Experiment 28
3.6 Summary 30

4 Data Analysis **31**
4.1 Introduction 31
4.2 Analytical Techniques 31
4.3 Case I 32
4.4 Case II 38

5 Conclusion **45**
5.1 Results 45

5.2 Recommendations 47

Appendix A Listing of Input Parameters 51

Appendix B Listing of Output Parameters 59

Appendix C Parameters Varied in Case I 61

Appendix D Parameters Held Constant in Case I 65

Appendix E Parameters Varied in Case II 67

Appendix F Parameters Held Constant in Case II 71

Appendix G Supplementals 73

List of References 75

Initial Distribution List 77

List of Figures

Figure 2.1 Screenshot during execution. 8

Figure 2.2 Sample `Mine` input file. 11

Figure 2.3 GAMET transitor logic flow for a non-cooperative minefield. . . 14

Figure 2.4 GAMET transitor logic flow for a cooperative minefield. 16

Figure 3.1 Case I screenshots of DP 2 (left) and 16 (right). 18

Figure 3.2 AUWS large diameter unmanned underwater vehicle (LD-UUV) concept of operation (CONOP) used to construct Case Study II. . 20

Figure 3.3 Case II screenshot of DP 10. 21

Figure 3.4 `Detonation` output file for Case I DP 2 displaying output used for calculating MOEs. 22

Figure 3.5 Partial NOB design matrix for Case I. 24

Figure 3.6 Scatterplot matrix for 12 (of 44) Case I simulation input variables showing the space-filling properties of the NOB design. 25

Figure 3.7 Sensor matrix for Case I DP 2. 26

Figure 3.8 Sensor matrix for Case I DP 227. 27

Figure 3.9 `Warhead` matrix for Case I design point (DP) 2. 28

Figure 3.10 VBA user form. 29

Figure 4.1 Residual by predicted plot for Y_{cas} fitted model in Case I. 33

Figure 4.2 Stepwise regression of Y_{cas} in Case I. 34

Figure 4.3 Prediction profiler for Y_{cas} in Case I. 34

Figure 4.4 Stepwise regression of Y_{det} in Case I. 36

Figure 4.5 Prediction profiler for Y_{det} in Case I. 37

Figure 4.6 Interaction profiler of Y_{det} in Case I. 37
Figure 4.7 Stepwise regression of Y_{cas} in Case II. 39
Figure 4.8 Prediction profiler for Y_{cas} in Case II. 40
Figure 4.9 Stepwise regression of Y_{det} in Case II. 41
Figure 4.10 Prediction profiler of Y_{det} in Case II. 42
Figure 4.11 Interaction profiler of Y_{det} in Case II. 43

List of Tables

Table 2.1 GAMET input files. Adapted from Belton (2013). 10

Table 2.2 GAMET output files table. Adapted from Belton (2013). 12

Table A.1 Probability of successful communications matrix found in example `Comm` file. 55

Table A.2 Probability of detection matrix found in example `Sensor` file. . . 56

Table A.3 Probability of transitor casualty matrix found in example `Warhead` file. 56

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

AIC Akailke's information criterion
ANOVA analysis of variance
AUWS advanced undersea weapon system
A2/AD anti-access/area denial
BDA battle damage assessment
BIC Bayesian information criterion
CCD central composite design
CMM Capability Maturity Model
CNMAWC Commander, Naval Mine and ASW Command
CONOP concept of operation
CPA closest point of approach
DoD Department of Defense
DOE design of experiments
DP design point
EC enabling capability
ENWGS Enhanced Naval Wargaming System
GAMET General Analytical Minefield Evaluation Tool
GOTS government off-the-shelf
GUI graphical user interface
ISR intelligence, surveillance, and reconnaissance
LCS Littoral Combat Ship
LD-UUV large diameter unmanned underwater vehicle
MCM mine countermeasures
MOE measure of effectiveness
MW mine warfare
NMAWC Naval Mine and Anti-Submarine Warfare Command
NOB nearly orthogonal and nearly balanced
NOLH nearly orthogonal Latin-hypercube
NPS Naval Postgraduate School

NSWC-PCD Naval Surface Warfare Center-Panama City Division

NWP Naval Warfare Publication

ONR Office of Naval Research

OPNAV Office of the Chief of Naval Operations

POF probability of firing

PRNG pseudorandom number generator

SC ship count

SEED Simulation Experiments and Efficient Designs

SPAWAR Space and Naval Warfare Systems Command

UMPM Uncountered Minefield Planning Model

USN United States Navy

UUV unmanned underwater vehicle

VBA Visual Basic for Applications

Executive Summary

Flexible and robust naval mine warfare (MW) planning tools will grow increasingly important as maritime mines grow more sophisticated, permitting their use in more restrictive environments and for a wider variety of missions. This thesis analyzes one of the tools that the Navy uses to evaluate existing and future minefields. General Analytical Minefield Evaluation Tool (GAMET), already in use by Naval Surface Warfare Center-Panama City Division (NSWC-PCD), holds the greatest promise for learning more about the potentially complex relationship between a ship and a minefield given GAMET's ability to model distributed netted sensors and pre-positioned mobile mines. Over the course of this study, through the use of a custom Visual Basic for Applications (VBA) user tool and a design of experiments (DOE) approach, GAMET was run more than 100 million times. Both a traditional minefield and a non-traditional cooperative minefield were studied using an advanced undersea weapon system (AUWS) concept proposal. The cooperative minefield allows separate sensor and warhead nodes to communicate and work together to target a contact of interest.

Although GAMET is available to the fleet, this study concurs with the user guide's recommendation that GAMET is best used by a trained analyst. Even a user who is familiar with MW would be challenged to properly use GAMET. The simulation's program-specific terminology, large number of input variables, lack of documentation, and requirement for creation of multiple and dependent text files hinders casual use. GAMET shows value primarily as a tool to evaluate more complex minefield and mine systems. Multiple improvements to both the user guide and the simulation are recommended as a result of this study.

The statistically significant factors utilized in GAMET that contribute to the efficacy of a traditional minefield inflicting a casualty on a transitor (i.e., ship or submarine) are the transitor's keel depth, the mine's damage and sensor range, the number of mines, the reliability of the mine, and the depth of the mine. For an AUWS cooperative minefield, significant factors for inflicting a casualty on a transitor include the number of detections required prior to launching a weapon, the ability to replenish (i.e., reload) a weaponized node, the total power available, number of sensor nodes, the replenishment criteria, the

sensor's range, and the reliability of warhead, sensor, and modem.

In summary, this study's exploration of GAMET yields insights into its operation and the factors that are significant to producing output. GAMET is confirmed to operate as documented and produces results consistent with intuition and real world experience. Additionally, GAMET is used to model an AUWS concept proposal and provides insights that can further AUWS development. A significant benefit of this study is that the VBA tool created provides a foundation to conduct further DOEs of more complicated mines and sensor technologies. Areas for further research include an evaluation of GAMET's planning mode, an analysis of the shortest path algorithm used for cooperative node communications, a study of additional AUWS models, a DOE incorporating more input factors, a sensitivity analysis of significant factors in the current study, and development of metamodels using different response (i.e., dependent) variables.

Acknowledgments

I am indebted to my wife, Tawni, and my daughter, Sloane, who give me something to look forward to each day. Additionally, a thank you to my father, former Hospital Corpsman 2nd Class James Flick, whose love for the Navy and sea stories inspired my desire for Naval service. Additionally, this thesis could not be possible without the counsel and recommendations of my advisors. I also would like to acknowledge LCDR McLemore's early programming assistance with Excel's Visual Basic for Applications, Dr. Tom Lucas's statistical analysis advice, and Dr. Gerald Brown's cogent editorial critique. Next, my thesis processor, Rebecca Pieken, who processed my thesis more than once and always with a smile! Lastly, a thank you to the Graduate Writing Center team, especially Chloe Woida, who spent countless hours and dozens of sessions helping me improve the structure and clarity of this thesis. Her technical writing knowledge, dedication to students, and sincere professionalism are a credit to the university.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1: Introduction

As technological innovations erode barriers to mine usage and the United States Navy (USN) force structure changes to reflect a focus on the littorals, mine warfare (MW) is becoming increasingly relevant. Current mine technology incorporates digital signal processing algorithms allowing mines to selectively target ships utilizing a combination of acoustic, electromagnetic, and water pressure measurements (Garrold 1998). Technological advances in the areas of unmanned systems, command and control, component miniaturization, and sensor improvements permit mines of increased sophistication to operate in areas and situations not previously considered. Future systems, such as advanced undersea weapon system (AUWS), will have distributed, netted sensors and offset weapon pods containing mobile "torpedo-like" mines (Everhart 2013).

Simulation is important to the MW community because it allows modeling and testing of many factors simultaneously without risk to personnel, and at a lower cost than live testing. The General Analytical Minefield Evaluation Tool (GAMET) is an object-oriented event-driven stochastic simulation used to evaluate minefield effectiveness, system performance trade-offs, and vessel vulnerability. This thesis adds to the body of literature on GAMET and its utility to MW by performing a state space analysis of GAMET, and by modeling a future mine warfare concept, hereinafter referred to as AUWS. The objectives in this thesis are to conduct a state space analysis (all possible factor settings and configurations within a system) and to model an AUWS concept proposal. In the process of conducting the analysis the following questions are answered: 1) Does GAMET function as expected? 2) What factors (i.e., inputs) contribute to a vessel's success in crossing a minefield, or conversely a minefield's success at causing a casualty within GAMET? 3) Can alternate, simpler tools (i.e., regression models) created based on GAMET output be useful to evaluate the efficacy of minefields?

1.1 Background and Motivation

New technology presents new challenges. In light of that, this section provides background information.

1.1.1 Mine Warfare

Mines are a pervasive concern for a ship and her crew in enemy waters, owing to their destructive capability, longevity, and detection difficulty. Joint Publication 3-15 defines MW as "the strategic, operational, and tactical use of mines and mine countermeasures either by emplacing mines to degrade the enemy's capabilities to wage land, air, and maritime warfare or by the countering of enemy-emplaced mines to permit friendly maneuver or use of selected land or sea areas" (Department of Defense 2011, p. GL-8). A mine is essentially a weapon that is isolated, immobile, and explodes in place (Glisson 2015). Naval mines have remained virtually unchanged for over a half century.

A minefield's purpose is to use deterrence or destruction to prevent the enemy from transiting a defined body of water. Without knowledge of a minefield, a vessel may attempt to transit mined waters, in which case the minefield's effectiveness is described by Hartmann as the ability to inflict damage thereby preventing the vessel from transiting through it (Hartmann 1979). With knowledge of a minefield, a vessel may avoid it or attempt to pass through it. In the real world, it would be impossible to determine whether a vessel avoided a minefield or was destroyed by it without overwatch, (i.e., continued monitoring of the minefield for battle damage assessment (BDA)).

The understanding of MW is changing as technological advances enable mines to be more than a floating or submerged bomb. Interconnected sensors providing targeting data to an unmanned underwater vehicle (UUV) modified to carry explosives forces an expansion of the definition of the mine and minefield. The MW community faces new MW defensive and offensive challenges in light of these new concepts. One such concept is AUWS.

1.1.2 Advanced Undersea Weapon System (AUWS)

First introduced in 2009, the concept of AUWS was driven by Admiral Drennan, Commander, Naval Mine and ASW Command (CNMAWC), who noted that "mines should not be thought of as dumb, immobile weapons waiting for unsuspecting mariners to run into them, but rather that existing developing technologies could transform the historic naval mine into a system of distributed sensors and moveable weapons with vastly improved capability" (Glisson 2015). Traditional minefields that today are comprised of non-cooperative mines (i.e., mines that do not communicate with one another) will in the future be com-

posed of cooperative mines (i.e., mines that do share targeting data). Everhart (2013) describes AUWS as a concept based on new technologies that will provide anti-access/area denial (A2/AD) in a littoral combat scenario. An AUWS minefield will be non-traditional – having distributed, netted sensors rather than static mines – and "will provide persistent, real-time autonomous intelligence, surveillance, and reconnaissance (ISR) and firing solutions for submerged, surface, shore, and airborne targets" (Everhart 2013). AUWS would allow U.S. forces to deploy a minefield that can detect, classify, and engage targets both in water and on land and can be remotely activated and deactivated.

1.1.3 MW simulation

Naval planners utilize a variety of mathematical models, and more recently simulation software to analyze minefields; simulation offers three benefits versus real world exercises. First, mining exercises are time consuming and difficult to conduct on a large scale. Simulation allows for mines, minefields, and plans to be tested without great expenditure of limited resources. Second, data collection is difficult given the overwatch requirement and the destruction of both mine and target. Third, the use of live mines is impractical and dangerous.

As mine sophistication escalates the complexity of the MW mission, MW simulation planning tools are of increased importance to the researcher, planner, and tactician. Simulation allows one to model and test many factors simultaneously. Therefore, it is prudent to explore avenues by which the planner and operator may better integrate this warfare area into planning and exercises. GAMET is one such avenue.

1.2 Study Objectives

The objectives of this thesis are to conduct a state space analysis and to model an AUWS concept proposal in GAMET. The intent is to learn more about the simulation and the factors that are significant to its output and by extension enhance the MW community's understanding of GAMET. First, a state space analysis of GAMET is undertaken utilizing a design of experiments (DOE) approach. DOE is a systematic process for designing a controlled experiment to evaluate the inputs (or factors) that contribute to changes in the outputs (or the response measures). Second, this thesis demonstrates that GAMET can

model a future mine warfare system (e.g., AUWS concept proposal).

1.3 Benefits

This thesis adds to the existing literature on GAMET, highlights areas for GAMET improvement, increases understanding of GAMET input factors and their relationship to simulation output, and provides insights into factors critical to an AUWS concept proposal. Identifying factors significant to a vessel defeating a minefield or conversely a minefield defeating an adversary is important to naval planners, strategists, and engineers involved in MW research and development. Additionally, identifying statistical significance of factors may decrease the number of inputs required to yield meaningful results and allows for creation of metamodels. A metamodel, in the context of this thesis, is a formulaic representation of a computer simulation. Metamodels enable a tactician or planner to estimate the effectiveness of a minefield in lieu of GAMET. Given the small number of dedicated MW simulation tools and increasing relevancy of MW, it is important to understand GAMET's limitations and identify areas for improvement. To understand whether the simulation is processing the input as expected, a thorough exploration of the simulation's factors is required. The requirement to create a file creation tool to apply the selected DOE can be applied to future research into GAMET, saving researchers countless hours. Finally, lessons learned in this study based on applying an AUWS case study can aid in the research and development of AUWS.

1.4 Literature Review

The author conducted a literature review focused on establishing a baseline understanding of naval mine warfare, current naval mine warfare tools, efforts in the development of persistent, distributed undersea weapons systems, and academic application of various DOEs.

There is a dearth of GAMET literature. Bard (2013) examines several models of varying degrees of complexity developed to analyze minefields. Bard concludes that GAMET provides output consistent with two other tools, Enhanced Naval Wargaming System (ENWGS) and Uncountered Minefield Planning Model (UMPM), and provides modeling flexibility to simulate advanced mining concepts.

DOE is a technique that allows an analyst to examine many more factors than would otherwise be possible and is a change from costly trial-and-error practices or sampling factors one at a time to cost-effective, versatile, and interactive tools based on statistical methods (Sanchez and Wan 2009, p. 60). As Sanchez and Wan (2012) explain, "Among the various designs, the use of Latin hypercubes has emerged as a good candidate since it provides an excellent compromise between resolution and efficiency" (Sanchez and Wan 2012, p.7). Inspired by the nearly orthogonal Latin-hypercube (NOLH) designs from Cioppa's PhD dissertation (2002), and follow-on work by Cioppa and Lucas (2007), the nearly orthogonal and nearly balanced (NOB) design allows for 100 continuous and 200 categorical factors to be varied in such a way that factors are nearly orthogonal to one another. Orthogonality ensures that a particular factor's effect on the response variable's variance can be isolated, and results in unconfounded estimates of the regression coefficients. The benefit of unconfounded estimates is it allows an experimenter to make more rigorous statements about a particular factor's impact on the response.

An exploration of all possible factor settings or configurations within the simulation (i.e., state space) is impossible due to combinatorial explosion. Just a two-level factorial experiment looking at each combination of GAMET's 65 factors, assuming a single replication of GAMET takes 1/200 seconds, would take 580 billion years to conduct, or over 414 times the estimated age of the universe! A DOE utilizing a NOB design would provide more information about GAMET's response than traditional DOE approaches.

1.5 Methodology

The following methodology, presented in four parts, was used in this thesis. First, it was necessary to become familiar with GAMET through hands-on usage because there was a lack of GAMET literature and due to the large number of ill-defined inputs. Input files were the vehicles for implementing a scenario while output files with response data were the result. I manipulated input text files supplied with GAMET and ran dozens of design points (DPs) to increase my understanding of the role of the factors, help inform the boundaries of the DOE, and confirm veracity of GAMET user's guide (Belton 2013).

Second, a state space analysis of GAMET was conducted using a DOE approach. The process was two-fold since the state space analysis of GAMET is dependent on 1) creating

the text files required to execute GAMET and 2) analyzing the output data produced. Two cases were defined: a traditional (i.e., non-cooperative) minefield case and a non-traditional (i.e., cooperative) minefield case. This decision was based on the twin objectives of learning as much as possible about the simulation by maximizing the number of factors examined and modeling an AUWS concept proposal. GAMET software architecture supports two distinct types of minefield and as a result utilizes unique factors depending on which minefield type is modeled.

Response variables were selected from the output files to act as appropriate minefield measures of effectiveness (MOEs). To learn as much as possible about factors significant to GAMET's response, it was necessary to run a broad set of DPs within each case and analyze the outcome of each DP. The DPs were created using a design blueprint provided by the Naval Postgraduate School (NPS) Simulation Experiments and Efficient Designs (SEED) center (SEED Center 2015). The design blueprint allowed for an efficient exploration of the simulation's state space in order to learn as much as possible about the simulation.

Third, a Visual Basic for Applications (VBA) program was written to automatically create the input files using the DPs, execute GAMET, and to extract the output data from the output files. Automating file creation was necessary because of the high number of DPs necessary to implement the DOE and the associated high number of text files required to run GAMET. Also, manual file creation, besides being time-intensive, is subject to data-entry error.

Fourth, analysis of the output data was primarily conducted using an interactive statistical analysis tool, JMP (2015), developed by the SAS Institute Inc. JMP allowed for creation of metamodels using regression, analysis of variance (ANOVA), and partition trees. Observations are based on my experience using GAMET and the analysis of GAMET data.

In the material that follows, Chapter 2 provides an overview of GAMET that is necessary for executing the DOE and understanding GAMET's input and output. Chapter 3 discusses the two cases that are developed, the selection of response variables, and VBA construction; all necessary steps for applying the DOE. Chapter 4 covers the data analysis and Chapter 5 reviews results and recommendations for follow-on research and improvements to GAMET.

CHAPTER 2:
GAMET Overview

This chapter provides additional background on GAMET, identifies inputs and outputs, and reviews simulation logic diagrams for both non-cooperative and cooperative minefields. Familiarity with these elements of GAMET is necessary for implementing the DOE, creating the VBA user tool, and later output analysis. This information is drawn primarily from my exploration of the simulation, its user's guide (Belton 2013), and a previous NPS thesis (Bard 2013). Additionally, correspondence with GAMET program manager Belton (2015a) clarified some aspects of GAMET operation. I lack specific knowledge of the simulation's algorithms for calculating many of its output data. Because it is distributed as a binary executable, and given the lack of clarifying documentation, GAMET is a blackbox.

2.1 Background

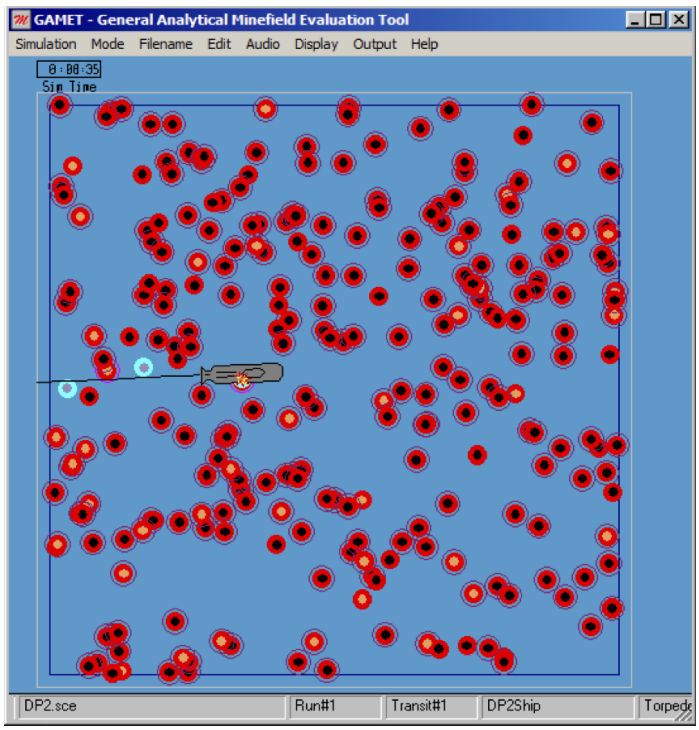
GAMET is a terminating stochastic simulation (i.e., one that has a clearly defined stopping condition) and utilizes an object-oriented and event-driven approach to simulation. It is used to evaluate and statistically analyze minefield effectiveness. It permits analysts to evaluate trade-offs in minefield system performance and the threat to vessels (i.e., transitors) (Bard 2013). GAMET can model scenarios that include standalone mines, cooperative networked fields of sensors, and remote weapons (Belton 2013). It is government off-the-shelf (GOTS) software consisting of compiled C++ code, developed by Naval Surface Warfare Center-Panama City Division (NSWC-PCD) and designed to be run on Windows PC-based systems.

The simulation uses a pseudorandom number generator (PRNG) and a seed file for generating a repeatable sequence of numbers. This allows GAMET scenarios to be reproduced (Belton 2015a). However, the PRNG seeding is restricted to a pre-selected set of values. Those values are automatically advanced through for each replication within a run. Since the first replicate's seed can only be changed utilizing the graphical user interface (GUI), I could not automate it as part of the VBA script. Hence, I cannot guarantee independent runs as the PRNG resets to the first seed for each DP. Nonetheless, based on the simulation's logic flow, GAMET utilizes the PRNG multiple times throughout a replication which according to Sánchez (2018) should "de-synchronize the use of randomness and reduce

between-run correlation."

The simulation was observed to exhibit stochastic variation discussed in Chapter 4. The GUI consists of pull-down menus, selectable icons, on-screen animation, and audio output. The GUI facilitates scenario observation, input data verification, and briefing support (see Figure 2.1). Since its release in 1998, GAMET has been used to support numerous studies, assessments, and war games (Belton 2015b).

Figure 2.1. Screenshot during execution.



A submarine passes through a randomly laid minefield. Undetonated mines are depicted as red circles with dark or light-colored centers depending on warhead reliability. Detonated mines are depicted as blue circles with light-colored centers.

GAMET has three modes: analysis, planning, and test. The analysis mode allows the user to create scenarios involving transitor(s) and mine(s) for the simulation to run, and provides output data. The planning mode allows the user to select a target MOE value and have the simulation increment the number of mines until the desired MOE is reached (Belton 2013). The test mode is not selectable by the user. I used the analysis mode exclusively for this

study.

2.2 Inputs

To conduct a DOE it is necessary to identify, define, and categorize all of the possible GAMET input parameters. Factors (or independent variables) are those parameters that are being considered for assessment in this study. This survey of the inputs not only aided me in understanding how GAMET functions but also assisted with setting realistic values for each factor. Because factors may be assigned a variety of values (or levels) and their characteristics can influence the choice of DOE, Sanchez et al. (2014) provide the following factor classifications:

- Quantitative or qualitative [categorical]. Quantitative factors naturally take on numerical values; qualitative factors do not (though they might be assigned numeric codings).
- Discrete or continuous (quantitative factors only). Discrete factors can have levels only at certain separated values; an example would be the number of x-ray machines in a hospital, which would have to be a non-negative integer, presumably with some upper bound. Continuous factors can assume any real value, perhaps within some range, such as the speed at which a vehicle is operated.
- Binary or not. Binary factors are naturally constrained to just two levels, like the classification of a part as either defective or non-defective. Non-binary factors could take on more than two values, but might still be tested at only two levels, typically "low" and "high," or might be allowed to assume (many) more than two levels in the experiment.
- Controllable or uncontrollable. In a simulation experiment all factors are manipulated and controlled, but in reality factors might be controllable or not. For example, the degree or nature of enemy jamming of a communications system would be controlled in a simulation, but not in an actual fight. This can affect how the experimenter interprets the estimates of the effects of factors. (Sanchez et al. 2014, p.5)

To use GAMET, a user must first create scenarios, described through inputs. The factor

values are captured through manually creating or editing of 9 to 13 input file types, shown in Table 2.1, for the specific DP to run. A DP may require more than one file of a particular type. For instance, if there are two different transitors, then two `Transitor` and two `Track` files must be created. GAMET may use dozens of input files depending on the complexity of the DP.

Table 2.1. GAMET input files. Adapted from Belton (2013).

Batch	*.scb	Batch processing of scenario files when >1 scenario
Boundary	*.fld	Defines minefield dimensions
Comms	*.cms	Defines comm/modem parameters
Comms Table	*.cmt	Sets the probability of successful comms vs. range
Mine	*.min	Defines mine laydown parameters
Planning	*.pln	Defines planning parameters. Not used in this study
Scenario	*.sce	Sets the number of runs
Sensor Table	*.sen	Sets the probability of detection vs. transitor at a water depth
Sequence	*.seq	Defines the sequence of transitor types per run
Transitor	*.trn	Defines transitor parameters
Track	*.trk	Defines transitor track parameters through minefield
Warhead Table	*.whd	Sets the probability of damage vs. transitor at a water depth
Waypoint	*.wyp	Defines waypoint transit parameters

The user guide does not provide a comprehensive list of inputs, therefore I created a list of GAMET parameters by examining each of the 13 input file types. A list of GAMET input parameters, grouped by file type, is found in Appendix A. The file type is listed with the associated filename extension in carets (e.g., `Batch` file <`*.scb`>). Additionally, the parameters in Appendix A are titled as they are in the GAMET input files for ease of reference and are identified as quantitative (continuous or discrete), or qualitative (categorical) variables.

An example of one of the input files is the `Mine` file presented in Figure 2.2. This file requires 32 inputs, more than any other single file type.

Figure 2.2. Sample `Mine` input file.

```
Node file for GAMET sim

NodesCooperative(TRUE,FALSE): FALSE
Require2DetectionsToLaunch(TRUE,FALSE): FALSE
ReplenishCriteria(ADJACENT,LOWPOWER,BOTH,NONE): LOWPOWER
NumberOfNodeGroups:  1


NodeGroupId: 1
NodeTypeName:   Torpedo
Weaponized(TRUE,FALSE): TRUE
NodePlacement(LINES,RANDOM,LIST): RANDOM
TriggerType(POF,SC): POF
NumberOfPOFsettings: 1
POFsettings(0-1): 0.98
NumberOfShipCountSettings: 1
ShipCountSettings: 1
InterShipDeadPeriod(min): 5
PowerTotal: 87
PowerDetect: 13
PowerTransmitLow: 12
PowerTransmitHi: 32
PowerReceive: 9
PowerReplenishThreshold: 7
Sensor&ModemReliability(0-1): 0.9
NumberOfWarheads: 1
WarheadReplenishTreshold(-1=NONE): 1
ReplenishTimeDelay(min): 42
WarheadReliability(0-1): 0.88
ProbOfSensorSurvival(0-1): 0.92
NumberOfNodesInGroup: 265
NodeDepth(ft): 360
WaterDepth(ft): 1000
```

Within the 13 input file types, 72 inputs were identified and 7 were determined to be uncontrollable. The remaining 65 total factors were considered to be controllable decision factors, and were broken into the following groups: 43 continuous, 11 discrete, and 11 categorical.

2.3 Outputs

GAMET outputs include many MOEs described in Naval Warfare Publication (NWP) 3-15 (Belton 2013). Identifying and defining GAMET outputs helped me determine the data necessary to calculate the MOEs. Once a DP is entered via input files, GAMET performs a simulation in which statistics are recorded over the course of a user-selected number of runs (Belton 2013). A run is a single replication of the DP. GAMET records the events, creating up to 16 different output files at the conclusion of each DP run. Each output file, along with its purpose, is listed in Table 2.2. In the analysis mode, 14 output files are generated. Two more output files are created: one for the planning and one for the test mode. Drawn from the 14 analysis mode output files, 47 output data fields are identified and listed in Appendix B.

Table 2.2. GAMET output files table. Adapted from Belton (2013).

Casualty	Logs the casualty density distribution
Casualty Density	Logs the casualty density distribution by transitor type
Cum Casualty Density	Logs the cumulative casualty density distribution by transitor type
Comms Stat	Logs successful end-to-end message statistics (generated for cooperative nodes)
Depletion	Logs the effects of mine depletion over a run of transitors
Detonation	Logs the mine detonation statistics
Distance to Casualty	Logs for each transitor distance to casualty statistics
Node Stats Average	Logs the location and statistics for nodes, averaged over the set of runs
Node Stats by Run	Logs the node lcoation and statistics for each run (generated in test mode)
Planning Results	Logs planning results (generated in planning mode)
Scenario	Logs the number of runs and the case files selected
Stopping Criteria	Logs the probability of X penetrators for an Y casualty stopping criteria
Threat	Logs the threat profile and expected casualties per transitor number
Threat Type	Logs the threat profile and expected casualties by transitor type
Time to Casualty	Logs for each transitor time to casualty statistics
Transitor Frequency	Logs the time between transistors

2.4 Simulation Execution Logic

This section reviews GAMET's simulation execution logic; this logic is fundamental to understanding how the program functions. The following logic description relies heavily

on the GAMET user's guide (Belton 2013). Later analysis of the data and conclusions regarding GAMET operation rely on the programmer's intent.

The study was bifurcated by the user's decision to model non-cooperative or cooperative minefields. A non-cooperative minefield is a conventional minefield composed of mines that are static and do not communicate with other mines. A cooperative minefield is a future minefield which is composed of mines that communicate targeting information and may be mobile. Therefore, two logic diagrams are depicted with explanatory text for each. The terms "node" and "mine" are interchangeable within GAMET documentation, however, in the context of the simulation, mines are either weaponized nodes (i.e., sensor with co-located warhead) or sensor-only nodes (i.e., sensor with offset warhead). A weaponized node can be static (i.e., explode at its position) or mobile (i.e., a torpedo that is launched).

The simulation distinguishes between a non-cooperative and cooperative minefield through the setting of the `NodesCooperative` factor. When `NodesCooperative = FALSE`, mines are represented by co-located sensors and warheads that do not communicate with one another and are destroyed upon actuation. Conversely, when `NodesCooperative = TRUE`, mines are broken into sensor and weapon nodes. This factor setting requires five new files, as each node type requires its own `Sensor` and `Warhead` matrices and three new files are required to simulate communication between nodes: a `Comm` "sensor" matrix, a `Comm` "warhead" matrix, and a `Comm` "parameter" file. Additionally, as a result of `NodesCooperative = TRUE`, other factors may be enabled that are related to the replenishment of power, the replenishment of warheads, sensor survival, and the number of required remote detections prior to launch.

At the onset of each run, GAMET uses input factor values from the `Mine` file to create a laydown of mines by "statistically perturbing the `Mine` file inputs" (Belton 2013). Some of the defining characteristics include the dimensions of the minefield, number of mines, reliability of the mines, mine power, sensor detection range, and warhead damage range. The mine positions remain constant throughout the run of transits.

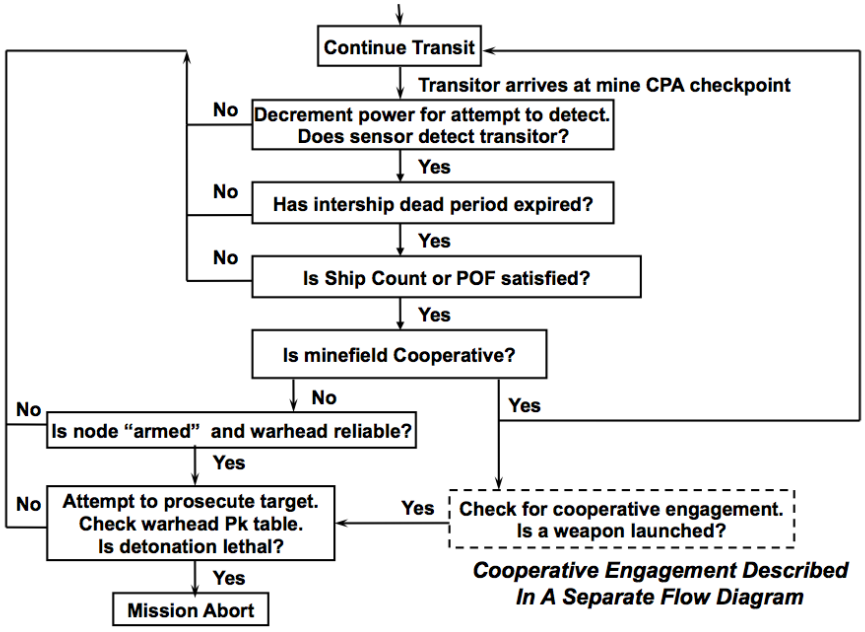
Transitor attributes are assigned as specified in `Transitor` and `Track` files and applied according to the `Scenario` and `Sequence` files. For each transit, a track is calculated using the `Track` file. Then, based on the user guide, as the transitor moves through the minefield, the simulation calculates the track's closest point of approach (CPA) to each mine with a

`SensorRange` greater than zero.

When the transitor arrives at its CPA checkpoint for a given mine, the mine attempts to detect the transitor. The maximum detection range is determined from the `Sensor` file, which also lists the probability of detection at various ranges. This probability is compared with a random performance draw to determine if a detection has occurred. If a detection is made, then the mine checks the `InterShipDeadPeriod` timer and then decrements its ship counter (if `TriggerType = SC` (ship count)) and its `PowerTotal` value. If these criteria are satisfied then the mine tells its warhead to detonate. Although it is not explicitly documented, analysis revealed that the `Warhead` file is referenced for the probability of damaging the transitor at various ranges, and the probability is then compared with a random performance draw to determine if the transitor has been destroyed.

See Figure 2.3 for a depiction of GAMET logic flow for a transitor entering a non-cooperative minefield. The diagram is incomplete as it does not show logic flow for an answer of "No" for the dashed box asking "Is a weapon launched?" If the answer to the preceding decision point "Is minefield cooperative?" is "Yes" then one looks at the cooperative engagement logic diagram, Figure 2.4.

Figure 2.3. GAMET transitor logic flow for a non-cooperative minefield. Source: Belton (2015b).



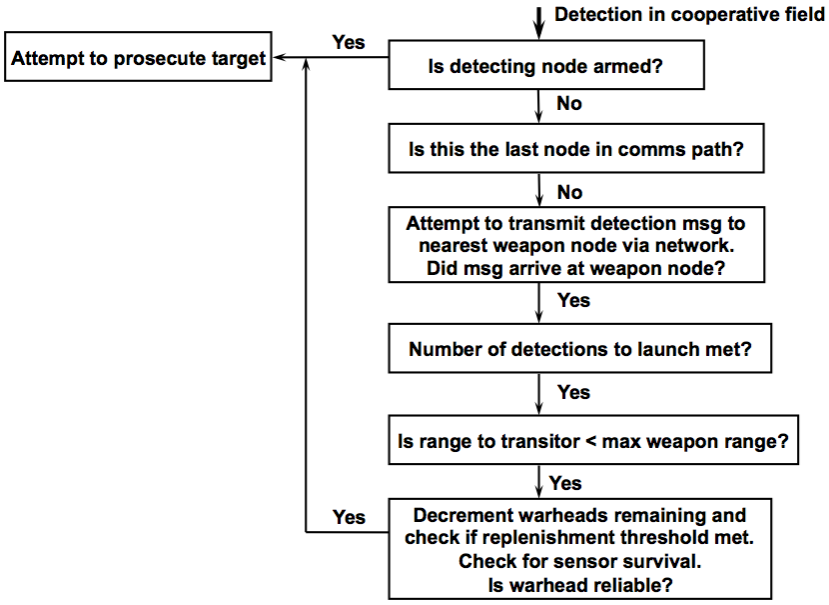
In the case of a cooperative minefield, a weaponized node will attempt to attack the transitor itself (i.e., launch warhead if armed and the transitor is within `WarheadRange`). By contrast, a sensor node will attempt to transmit the message to the nearest node (i.e., one hop at a time). If a node-to-node communication hop is unsuccessful, the transmitting node will wait the `TransmissionRetryDelay` time and then attempt to re-transmit, up to the specified number of `MaxTransmissionAttemptsLowPower`. If based on the `Comms` matrix the transmission is unsuccessful, the node then tries to use high power communications to skip a hop (i.e., transmit a message to a node two arcs away) and transmit the message to the following node. It tries this approach up to the `MaxTransmissionAttemptsHiPower` value. If the message communication to the weapon node is successful, GAMET attempts to attack the transitor and then uses the `ProbOfSensorSurvival` value to determine if the weapon launch operation has destroyed the sensor node.

If `Require2DetectionsToLaunch = TRUE`, then for remote detections two separate detection messages for a single transitor must be received by the weapon node before a warhead will be launched. Once a warhead is launched, the `Warhead` file matrix is used to determine if the transitor is damaged. If a weapon node reaches its `WarheadRepThreshold` it triggers field replenishment. The weapon and sensor nodes that meet the `ReplenishCriteria` will be scheduled for replenishment based on the `ReplenishTimeDelay`. Replenishment, however, will not occur until the target's transit is complete.

Throughout the process, power usage is tracked and is decremented from each node's `PowerTotal` value for transmission and reception of messages as well as node detections. Once a node's power level reaches zero, it is turned off.

See Figure 2.4 for a depiction of GAMET logic flow for a transitor entering a cooperative minefield. This logic flow diagram is also incomplete for the following decision points: "Is this the last node in the comms path?", "Did msg arrive at weapon node?", "Number of detections to launch met", "Is range to transitor < max weapon range?", and "Is warhead (i.e., weaponized node) reliable?".

Figure 2.4. GAMET transitor logic flow for a cooperative minefield. Source: Belton (2015b).



CHAPTER 3:
Experiment Design and Implementation

This chapter develops two cases, the simulation of non-cooperative and cooperative minefields, to support the thesis objectives. It identifies the response variables of interest. Then the application of the NOB design spreadsheet is explained with justification. A brief discussion of VBA's role with an examination of the probability matrices is introduced because scripting is critical to executing the DOE. Finally, the chapter discusses how the experiment was executed. Analysis of the output is presented in Chapter 4.

3.1 Case Study Selection and Construction

In order to apply the DOE and construct meaningful metamodels, cases were developed that represent potential real world uses of the simulation. As discussed in Chapter 2, the simulation is able to replicate a traditional minefield where mines do not communicate with one another, and a future cooperative minefield having sensor and weaponized nodes that are capable of communicating. This section includes the constraints that help frame each case. After introducing the two cases, factors that are varied between high and low level settings and those parameters held constant are identified for each case.

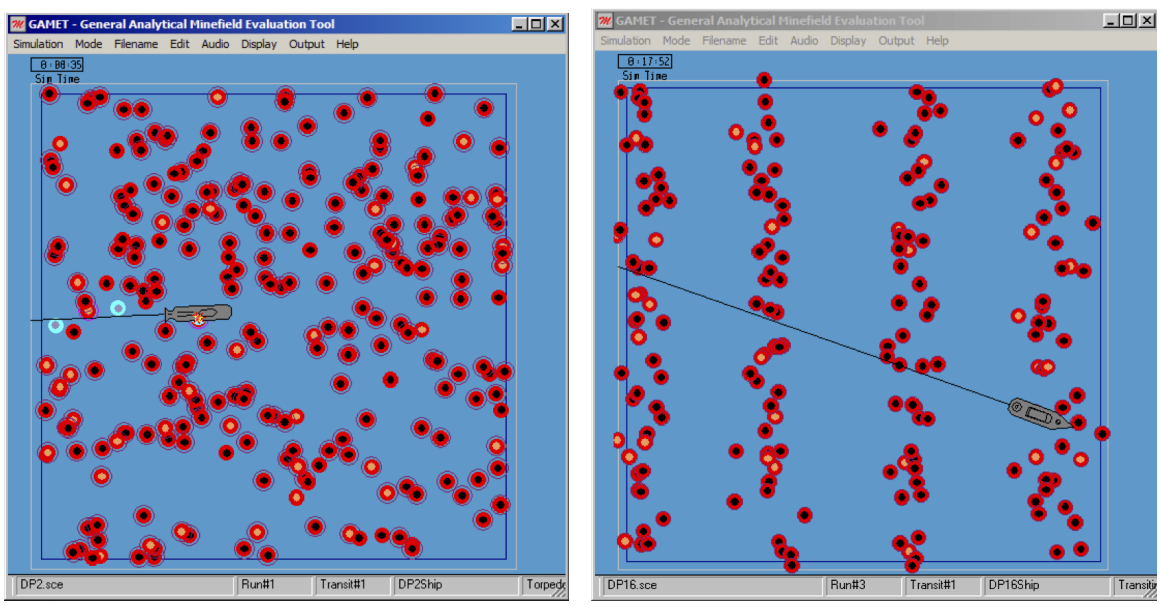
The simulation can utilize the 65 parameters previously described in Chapter 2. However, depending on the case that is being modeled, the simulation is not expected to use all the parameters. For example, Case I does not utilize cooperative nodes, thus parameters related to communication should not be used by the simulation based on available literature. However, the modeler's intent may not have been properly implemented. In order to ascertain the distinction between intent and correct implementation, this research varied as many inputs as feasible.

3.1.1 Case Study I: Non-Cooperative Minefield

This case consists of a single transitor attempting to cross both a randomly laid minefield and a mine line minefield. This case is simple allowing for a basic understanding of factor relationships with the outcomes. Constraints are listed and are demonstrated in Figure 3.1:

1. The transitor crosses minefield via a straight-line path (no waypoints).
2. Transitor parameters remain constant during transit (e.g., speed, keel depth, type, etc.).
3. Mine parameters remain constant within the minefield (e.g., mine depth, sensor range, warhead range, etc.).
4. A non-cooperative minefield is created through random or mine line placement.
5. Mines have a maximum sensing radius between 100 - 400 yards.
6. Mines have a maximum damage radius between 100 - 300 yards.
7. The density of the minefield is between 3 - 16 mines per square mile.
8. The minefield water depth is fixed at 1000 feet.

Figure 3.1. Case I screenshots of DP 2 (left) and 16 (right).



Two of the 512 DPs demonstrate two Case I constraints: 1) transitor crosses minefield via a straight-line and 2) the use of both random and mine line mine placement. On the left, a submarine passes through a randomly laid minefield, while on the right, a surface ship passes through a 4-mine lines minefield.

Choice of Factors and Levels

The DOE for Case I varies 47 factors. Factor values are primarily set at realistic ranges based on my naval experience and research to construct reasonable case studies. The factors,

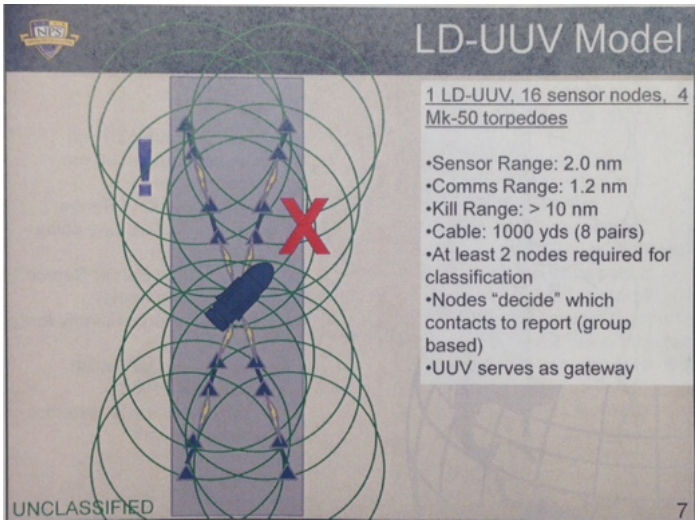
with my justification for their low and high level settings, can be found in Appendix C.

Eighteen parameters remained constant and their level settings are specified with justification in Appendix D. As noted earlier, an attempt was made to vary as many parameters as possible. However, some parameters could not be implemented in the VBA user form or were outside the scope of Case I.

3.1.2 Case Study II: Cooperative Minefield

The second case explores more of the state space of GAMET through additional factors. Since the simulation models cooperative fields, it may be valuable in evaluating various AUWS models. After reviewing four original AUWS concept of operation (CONOP) proposals, the author chose to model the one shown in Figure 3.2 in GAMET. According to Admiral Williams, USN retired, it is the model on which Office of Naval Research (ONR) is basing an AUWS prototype (Williams 2015). The model consists of a large diameter unmanned underwater vehicle (LD-UUV) acting as a mobile weapon pod with the ability to utilize non-organic sensors. The UUV contains torpedoes and communicates with sensors for transitor detection. In GAMET the UUV is modeled as a weaponized node with six warheads.

Figure 3.2. AUWS LD-UUV CONOP used to construct Case Study II.
Source: Everhart(2013).



The cooperative minefield depicted consists of two intersecting "sensor lines" and a bullet-like shape representing the LD-UUV or weapon pod containing four to six mobile mines. The triangles represent the 16 sensor nodes while the ring centered on each sensor indicates the detection radius.

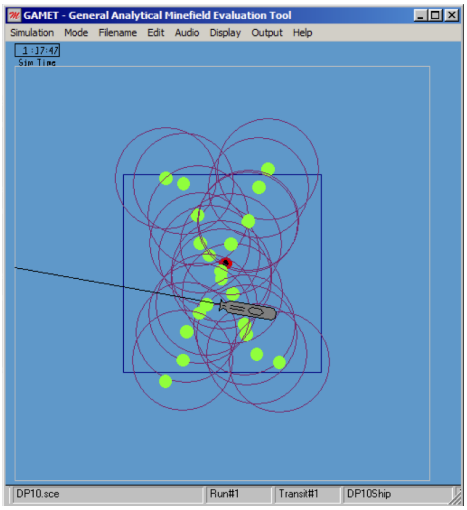
The content of Figure 3.2 was used to set some of the factor levels as well as formulation of the constraints listed as follows:

1. The transitor crosses minefield via a straight-line path (no waypoints).
2. Transitor parameters remain constant during transit (e.g., speed, keel depth, type, etc.).
3. Mine parameters remain constant within the minefield (e.g., mine depth, sensor range, warhead range, etc.).
4. A cooperative minefield created with two intersecting sensor lines and a single weapon pod.
5. The weapon pod has a capacity of 6 mobile mines.
6. The sensors have a sensing radius between 3600 - 4000 yards.
7. The mobile mines have a range between 16000 - 20000 yards.
8. The sensor communication range is between 1600 - 2400 yards.
9. The number of sensors per line varies between 7 and 11.

10. The minefield water depth is fixed at 1000 feet.

Figure 3.3 displays how Case II is graphically represented in GAMET and highlights the representation of the minefield as two intersecting sensor lines with a central weapons pod.

Figure 3.3. Case II screenshot of DP 10.



Case II DP demonstrating constraints: 1) transitor crossing minefield via a straight-line and 2) cooperative minefield created with two intersecting sensor lines. The single central weapon pod (i.e., static weaponized node) is depicted as red circle with dark center.

Choice of Factors and Levels

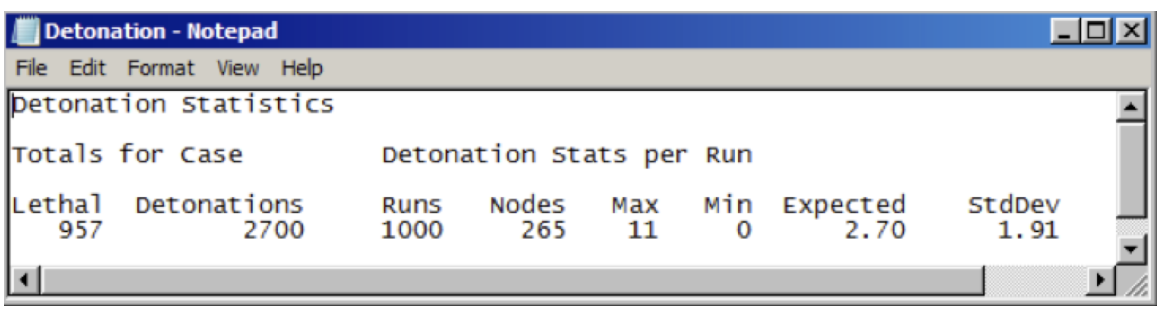
Factors were held at Case I levels to allow for comparisons to be made between non-cooperative and cooperative minefields. Although direct comparison of the two cases is impossible, it is possible to make inferences based on MOEs and resource commitments. For instance, if both Case I and Case II result in a similar MOE value with mine detection range held constant in both cases, and Case II requires fewer sensors, then one can infer cost, energy requirements, and other attributes about an AUWS concept. A list of the 44 factors that are varied and their level settings is found in Appendix E. Twenty-one parameters are held constant as dictated by the case and are listed in Appendix F.

3.2 Selection of Response Variables

Appendix B lists GAMET output parameters. One output field, the standard deviation of the detonations, Y_{sd} is used to confirm the stochastic nature of the simulation. Two others, serve as appropriate response variables to measure the effectiveness of a minefield. The first, Y_{cas}, is the MOE for the rate of casualties per transitor per transit. It can be calculated by dividing lethal detonations by the number of transits. Both outputs are found in the `Detonation` output file shown in Figure 3.4. The higher the probability of a casualty, the more effective the minefield will be at preventing transitors from crossing it.

A second MOE is Y_{det}, the number of detonations per transitor per transit. This can be calculated by dividing the number of detonations by the number of transitor runs. Although a ship may not suffer a casualty, a mine detonating has a non-zero probability of deterring continued crossing by the transitor and can deter other transitors from entering the minefield. While the deterrence value cannot be quantified, a minefield's effectiveness relies partially on knowledge of its existence (Hartmann 1979). Thus a mine detonating does contribute to a minefield's effectiveness.

Figure 3.4. `Detonation` output file for Case I DP 2 displaying output used for calculating MOEs.



```
Detonation - Notepad
File  Edit  Format  View  Help
Detonation Statistics

Totals for Case          Detonation Stats per Run

Lethal  Detonations      Runs   Nodes   Max   Min  Expected    StdDev
   957         2700      1000     265    11     0      2.70      1.91
```

3.3 Design Methodology

The thesis objectives outlined in Chapter 1 guided the choice of experimental design. (NOB) and NOLH designs are documented as useful in advancing a broad understanding of simulations without any prior knowledge about the response surface (Sanchez and Wan 2012). Also, the Latin hypercube's space-filling property (i.e., DPs are scattered throughout

the design space) means that much of the possible scenario range is captured (Sanchez and Wan 2012).

After the experimenter identifies input factors and response parameters of interest, the NOB design allows many user-selected factors to be varied that are discrete, continuous, and categorical. Factorial or gridded designs cannot be applied given the high number of factors and the broad range of value (Sanchez and Wan 2012). A coarse grid (2^k factorial) would not provide sufficient depth of coverage of a non-linear response surface and most importantly yields too many DPs to process. Likewise, a fine grid (more than two levels for factors) would be even more constrained by computer resources given the large size of the factor space (DeHollan 2015).

The NOB design spreadsheet is available for download off of the SEED Center's website (Vieira 2012). The NOB spreadsheet tool generates the input matrix, once all the factors with their associated low and high levels are entered.

Figure 3.5 is a snapshot of the Case I design spreadsheet with factor values set as detailed in Appendix C. The maximum magnitude of pairwise correlations between columns in the design matrix is 0.06 with the average equal to 0.02. The row entries represent each unique DP that is run, while each column represents one of the factors that are varied. Due to space constraints, Figure 3.5 shows the first 20 of the 512 DPs and 10 of the 44 factors that are varied in the design. Note that three of the ten factors shown are categorical followed by seven continuous factors, allowed by the flexibility of the NOB design. For the complete design matrix with factor level settings, see Appendix G for information on obtaining supplementals.

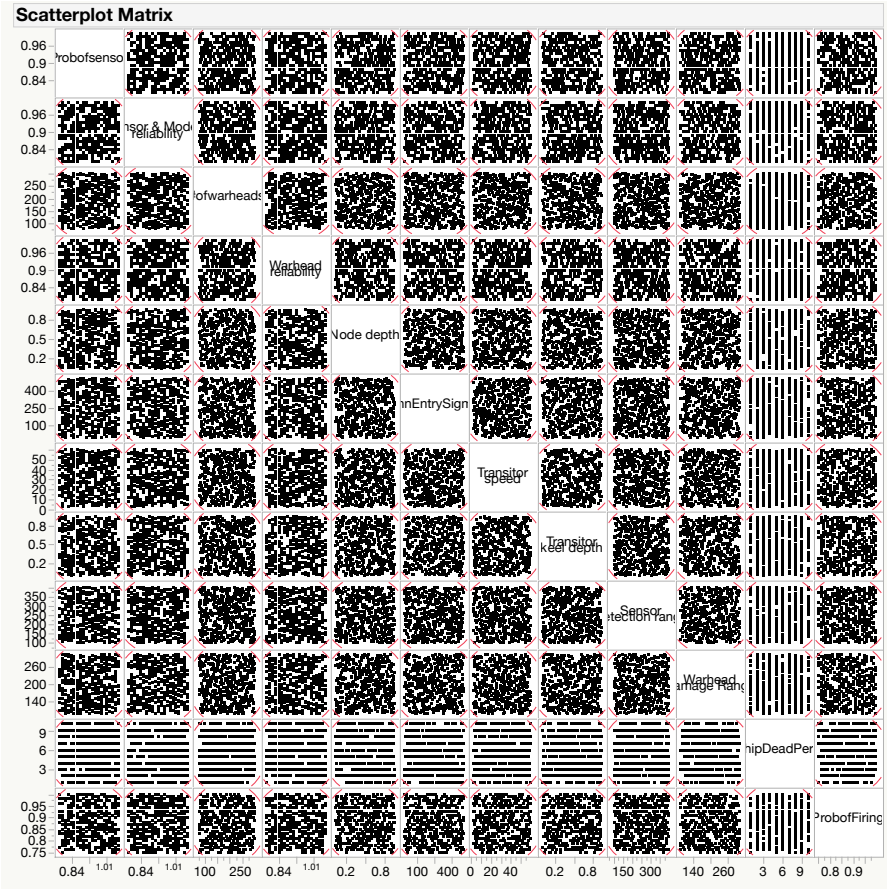
Figure 3.5. Partial NOB design matrix for Case I.

low	**1**	**1**	**1**	**0.8**	**0.8**	**80**	**0.8**	**0.05**	**0**	**3**
hi	**2**	**4**	**4**	**1**	**1**	**300**	**1**	**0.95**	**500**	**60**
decimals	**0**	**0**	**0**	**2**	**2**	**0**	**2**	**2**	**0**	**1**
discrete levels	**2**	**4**	**4**							
Design Pt #	**NodePlacemen**	**RepCriteria**	**EntryType**	**SensSurv**	**Sens&ModRe**	**#OfWarheads**	**WarheadRelia**	**NodeDepth**	**ChnEntrySig**	**TransSpeed**
1	2	3	1	0.81	0.99	117	0.99	0.77	27.4	53.2
2	1	2	2	0.9	0.87	265	0.83	0.37	111.5	33.3
3	1	2	4	0.87	0.97	89	0.85	0.12	342.5	23.2
4	2	4	1	0.89	0.81	290	0.96	0.67	119.4	23.5
5	2	1	1	0.96	1	118	0.82	0.31	229.9	40.6
6	2	4	2	1	0.86	279	0.99	0.52	162.4	25
7	1	1	1	0.98	0.89	109	0.81	0.63	369.9	42
8	1	3	2	0.89	0.99	143	0.84	0.33	195.7	43.9
9	1	2	4	0.83	0.85	281	0.97	0.91	366.9	42.9
10	1	3	2	0.89	0.84	291	0.98	0.43	277.9	7.2
11	2	3	1	0.81	0.8	82	0.84	0.72	409	19.6
12	2	2	4	0.91	0.97	120	0.99	0.72	56.7	22.2
13	2	1	2	0.88	0.88	236	0.96	0.82	93.9	38.5
14	1	2	3	0.98	0.87	183	0.97	0.18	275.9	26.3
15	2	4	1	0.96	0.94	282	0.86	0.44	53.8	42.3
16	2	4	2	0.95	1	192	0.84	0.28	461.8	43.8
17	2	1	2	,98	0.88	108	0.81	0.62	136	24.6
18	2	3	3	0.86	0.94	278	0.93	0.59	491.2	42.6
19	2	2	1	0.84	0.96	230	0.9	0.06	284.5	4.1
20	2	3	3	0.8	0.84	154	0.81	0.25	341.5	40

In column 1, DP 16 factor settings are (read left to right): `NodePlacement` = 2 (`LINES`), `RepCriteria` = 4 (`NONE`), `EntryType` = 2 (`BOUNDED`), `ProbOfSensorSurvival` = 0.95, `Sensor&ModemReliability` = 1.0, `NumberOfWarheads` = 192, `WarheadReliability` = 0.84, `NodeDepth` = 0.28 (280'), `EntrySigma(CHANNEL)` = 461.8, and `Speed` = 43.8.

The choice of a NOB design allows for a more in-depth analysis of GAMET by producing DPs that cover the input space range more completely. The design's space-filling property is visible in the scatterplot matrix of the variables shown in Figure 3.6. Each pane (above or below the factor diagonal) in the scatterplot shows a different variable-pair combination for all 512 DPs. Due to space constraints, only 12 of the 44 factors are shown. Therefore, only 66 of the 946 unique pairs appear.

Figure 3.6. Scatterplot matrix for 12 (of 44) Case I simulation input variables showing the space-filling properties of the NOB design.



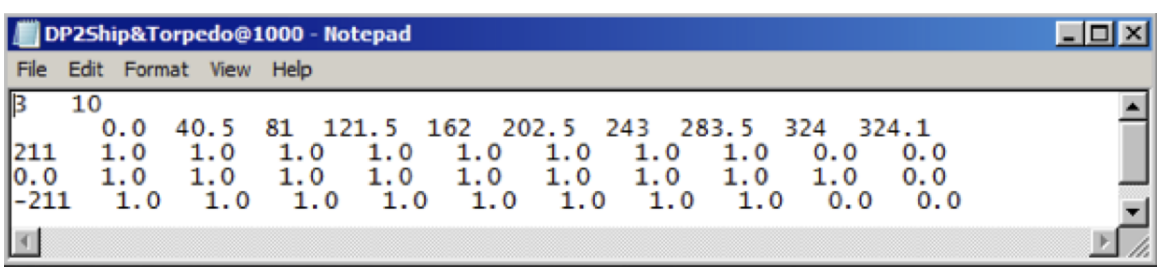
3.4 VBA Construction

In order to apply the NOB design, it was necessary to write a program to create the thousands of files necessary to run the simulation millions of times. The creation of a VBA program required me to understand each line of each file and aided my comprehension of each input. The program creates the files and provides unique file names so that GAMET can reference the files. Files that the VBA program creates must have the significant factors listed in a standard order to ensure the factor levels are read by GAMET. This section examines two examples of the 13 total file types that can be created within GAMET.

The creation of the sensor detection and warhead damage probability matrices (or tables) are critical to modeling the relationship between mine and transitor and the subsequent MOEs. GAMET will not provide accurate data or will fail to run altogether without their proper creation. The probabilities found in each table are based on the depth difference between the transitor and the mine. The simulation allows application of a standard deviation to the transitor's keel depth in the form of the input variable: `DepthSigma`. Because the tables must contain the full range of the probabilities based on the depth difference of the transitor and the mine, if the user applies a non-zero `DepthSigma`, then a table must be created that may be referenced by the simulation with the maximum depth difference.

Figure 3.7 shows the `Sensor` file that is created by the VBA script for DP 2. The first line shows the matrix dimensions: three rows by ten columns. The second line is comprised of increasing sensor ranges between 0 and a maximum range of 324 yards. Because GAMET requires a number beyond the maximum range to read and interpret the matrix correctly a final column in each matrix is added that sets the probability to zero for the maximum range plus 0.1 yards. The third, fourth, and fifth lines are indexed by the depth difference between the transitor and the mine. In this case, `DepthSigma = 3` feet. Therefore, to ensure coverage of 99.7% of keel depth variance, an additional 9 feet (3 sigmas) is added to the depth difference (202 feet). Because GAMET can be used to simulate low-flying aircraft over a minefield, the matrices all are created with positive and negative lines of depth difference. This gives the user the ability to provide different probabilities associated with a mine sensing an airborne transitor vice a surface or subsurface transitor.

Figure 3.7. Sensor matrix for Case I DP 2.



Associated tables are created for each unique transitor for each unique DP. When an individual DP is run thousands of times, the track of a transitor has its depth varied by a

normal random variable with standard deviation `DepthSigma`. The tables are referenced to interpolate horizontally and vertically in the water column. More columns result in increased fidelity. For instance, the Sensor files for both Case I and Case II model a "cookie-cutter" sensor. If a transitor is within the radius of a mine then there is a 100% probability of transitor detection. The only place this is not true is at the edges of the sensor. For instance, if a transitor's CPA is between 283.5 and 324 yards in Figure 3.7 then the probability of detection will be interpolated between the columns. GAMET requires that all matrices are constructed with a final column with all probabilities set to zero in order for the matrices to be read correctly.

In another example of a Case I `Sensor` file, DP 227 (Figure 3.8), the sensor's maximum range (193 yards) is less than the depth difference between transitor and mine (606 feet). Thus matrix row 1's (i.e., third line) probability of detection is zero from 0 to 193 horizontal yards. Nonetheless, the VBA script properly labels the "cookie cutter" probability at a depth difference equal to zero. This allows the simulation to interpolate between the third and fourth lines.

Figure 3.8. Sensor matrix for Case I DP 227.

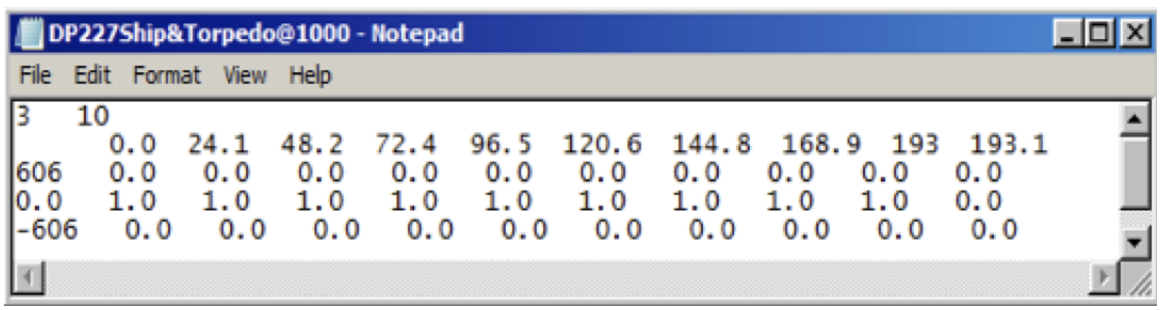
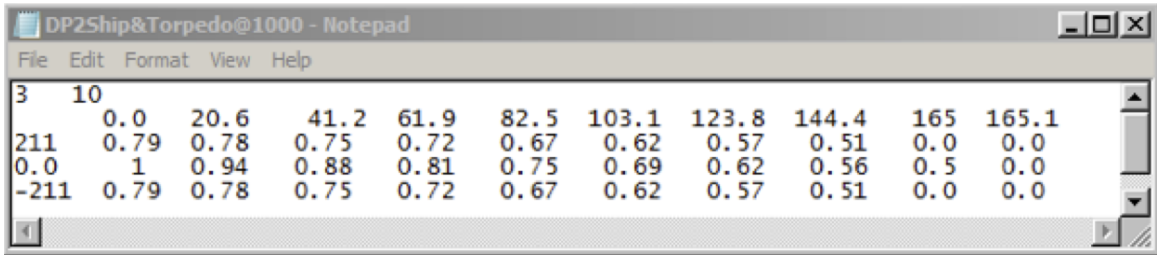


Figure 3.9 shows the `Warhead` file that is created by the VBA script for DP 2. Like the `Sensor` file, the first line shows the matrix dimensions. The second line is comprised of increasing warhead ranges between 0 and 165 yards. The matrix row indices match DP 2's `Sensor` file matrix since both the `Sensor` and the `Warhead` matrices are based on probabilities associated with the depth difference between the transitor and the mine. For both Case I and Case II, the `Warhead` files are modeled to create a probability distribution with warhead damage probability of 100% at 0 yards and decreasing to 50% at maximum

warhead range. The probabilities were calculated based on linear distance, thus a 100% probability of damage at a depth difference equal to zero will not equate to 100% at the actual depth difference.

Figure 3.9. `Warhead` matrix for Case I DP 2.



```
3   10
      0.0   20.6    41.2  61.9   82.5  103.1  123.8  144.4   165  165.1
211   0.79  0.78   0.75   0.72   0.67   0.62   0.57   0.51   0.0   0.0
0.0     1   0.94   0.88   0.81   0.75   0.69   0.62   0.56   0.5   0.0
-211  0.79  0.78   0.75   0.72   0.67   0.62   0.57   0.51   0.0   0.0
```

Looking at matrix row 1 (i.e., third line), indexed by the 211 foot depth difference between transitor and mine, there are ten columns with decreasing probabilities. These probabilities represent the likelihood that upon actuation a mine will inflict a casualty on the transitor, and they decrease as the horizontal distance increases from 0 to 165.1 yards.

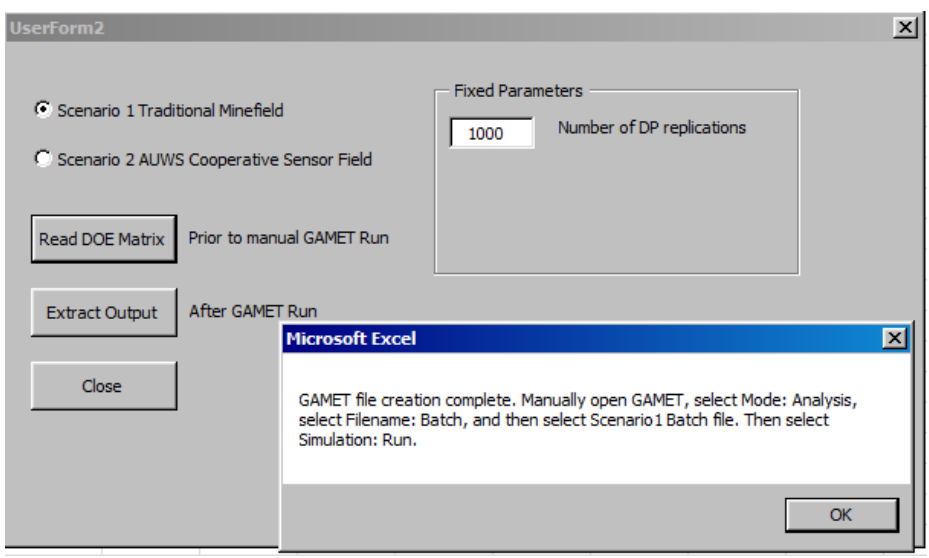
3.5 Running the Experiment

To run Case I, the simulation requires nine files for each DP. This is the minimum number for a simple case. A user manually creating a single DP will edit nine file templates included with the software. At 60 seconds per file, one DP will take nine minutes to create. Creation of 512 DPs would take over 76 hours. As the user guide points out, in a more complicated case with two mine types at three water depths with four transitor types, each DP will require 62 input files. Manually editing 62 input files for a single DP would take an hour given no data input errors. Because of the inordinate amount of time to manually create thousands of files and to minimize data input errors, a script was created in VBA to read the DOE matrix and to create the 4,609 files for Case I and the 6,657 files for Case II. Next, the VBA program requires the user to manually open GAMET and select the `Batch` file. The `Batch` file allows the simulation to run one DP after another for the specified number of times rather than the user manually initiating GAMET for each DP. The simulation software also produces an additional `Scenario` file for each DP that contains identical information as the original DP `Scenario` file but has a different name. The addition of a second `Scenario`

file for each DP adds an additional 512 files. Thus a total of 4609 + 512 = 5,121 and 6657 + 512 = 7,169 input files are necessary for Case I and Case II implementation, respectively.

After the simulation completes running, the VBA user form allows the user to extract the response variables from the respective output files and creates an Excel worksheet with the DPs collated with the response variables. The user form is depicted in Figure 3.10, while the VBA code is available as a supplement per Appendix G. For Case I, 512 DPs are replicated 10,000 times. The simulation averages each DP's results so although 5,120,000 simulation runs are made, GAMET produces 512 responses — one for each DP. For Case 2, 10 transitors attempt to cross during each run (or replication) in order to evaluate replenish-related factors that are documented as not used for a non-cooperative minefield. Additionally, these factors are expected to be utilized for a case study with a node capable of "firing" six torpedoes during a single replication. However when I attempted to replicate each Case 2 DP 10,000 times, the simulation would halt prior to completing the replications. Therefore each DP is replicated 1,000 times encompassing the data of 10,000 transits. These responses are the basis for the analysis conducted in Chapter 4.

Figure 3.10. VBA user form.



VBA user form for selection of case, entry of number of replications per DP, automatic file creation, and automatic data extraction.

3.6 Summary

Although simulation allows replication of a set of experiments at reduced cost and time, it is important to proceed with care when implementing a DOE. The objective of the DOE affects the factors that are varied within the DOE as well as their factor level settings. Because one of this study's objectives was to develop usable regression metamodels, factor values were set based on my experience or historical data. For instance, maximum sensor range is greater than maximum warhead damage range and both are varied at levels consistent with floating contact mines in Case I. Two cases are studied using two separate DOEs because of a desire to incrementally increase complexity and facilitate comprehension of the factor space. The next chapter is my analysis of the output data produced by the simulation as a result of applying the DOE.

CHAPTER 4:
Data Analysis

4.1 Introduction

This chapter covers the analytical techniques used, the outcome of the analyses, and highlights the insights gained through this research. After conducting the two DOEs described in Chapter 3, the output data were added as columns to the NOB design sheet and then the resulting Excel spreadsheet was imported into JMP for analysis.

Regression metamodels were a result of the analyses and may be useful in situations where running a complex simulation may not be feasible.

4.2 Analytical Techniques

Descriptive statistics and stepwise regression were the primary analytical techniques used. An alternative technique to developing a metamodel is to perform partitioning. Partition-based models did not reveal further insights and due to the high number of splits, lacked the utility of simpler regression-based models; therefore they are not presented.

4.2.1 Descriptive Statistics

Initial analysis of the data was conducted utilizing graphical tools and techniques. Although not all tools are demonstrated in this study, tools used included prediction and interaction profilers, box and scatter plots, and histograms. Each DP's output consisted of three pieces of data: the sample mean casualties per DP determined across replications (Y_{cas}), the sample mean detonations per DP determined across replications (Y_{det}), and the standard deviation of detonations per DP determined across replications (Y_{sd}). Y_{sd} was not treated as an MOE and therefore was not analyzed. However, the simulation was confirmed to be stochastic by the non-zero Y_{sd} result across DPs. Outliers were not excluded from the data set since they are a legitimate product of the simulation.

4.2.2 Stepwise Regression

For both cases and with respect to each MOE, the regression analysis process was two-fold: using a stepwise technique followed by a standard least squares regression. Stepwise regression was conducted by utilizing all first-order main effects, second-order with interactions, and third-order polynomials. The concept of practical significance as described by Sanchez et al. (2012) was used to construct models with fewer predictors and facilitate interpretation.

4.3 Case I

This section discusses the analysis of the Case I data using descriptive statistics. Second, it describes the regression analysis of both response variables, Y_{cas} and Y_{det}.

4.3.1 Case I Descriptive Statistics

Initial analysis indicates the data produced by the simulation conforms to expectations. Y_{cas} ranged from 0 to 0.999. Y_{det} varied between 0.009 and 11.06. Analyzing the Y_{det} distribution, there were no DPs in which a zero detonation response occurs. Thus the possibility exists of a casualty occurring for all DPs. In fact, at least one casualty occurred in 10,000 replications of each DP.

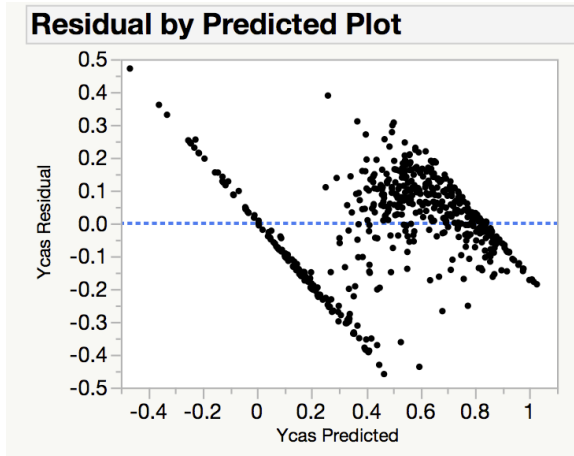
4.3.2 Case I Regression Analysis

Both response variables, Y_{cas} and Y_{det}, were analyzed using regression analysis. A prediction and an interaction profiler enabled understanding of factors and their contribution to the respective response variable.

Y_{cas} Analysis

The initial regression produced a model with an R^2 = 0.78 and consisted of 12 effects. However, the actual by predicted plot was in the shape of an s-curve indicating that a logistic transformation of the response variable may improve the model. Additionally, the residual by predicted plot in Figure 4.1 shows the residuals are not randomly scattered around zero indicating the model fits the data poorly.

Figure 4.1. Residual by predicted plot for Y_{cas} fitted model in Case I.



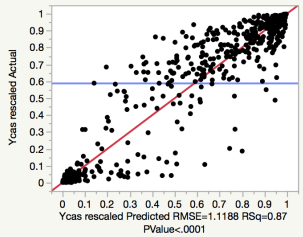
Therefore, a logit transformation on Y_{cas} was performed and as a result a more parsimonious model was produced. Additionally, I created a term by taking the absolute value of `KeelDepth - NodeDepth`. Independently, each factor is statistically insignificant but creating a term using the two factors, based on my experience that the distance between the ship's keel and the mine is important in the real world, resulted in a better fitting model when the regression analysis was conducted again.

Figure 4.2 displays the actual Y_{cas} value by predicted plot and summary data for the final model consisting of 10 effects: five single factors (or main effects), 2 two-factor interactions, and three quadratic terms. The summary of fit indicates that the model's R^2 = 0.87. The sorted parameter estimates show the terms that contribute to the prediction expression and their relative contribution to the total weight. Five main effects encompass six factors that are statistically significant. They are `WarheadRange, SensRange`, `NumberOfWarheads`, `WarheadReliability, KeelDepth`, and `NodeDepth`.

Figure 4.2. Stepwise regression of Y_{cas} in Case I.

Response Logit(Ycas rescaled)

Actual by Predicted Plot



Summary of Fit

RSquare	0.86571
RSquare Adj	0.863029
Root Mean Square Error	1.118814
Mean of Response	0.352905
Observations (or Sum Wgts)	512

Analysis of Variance

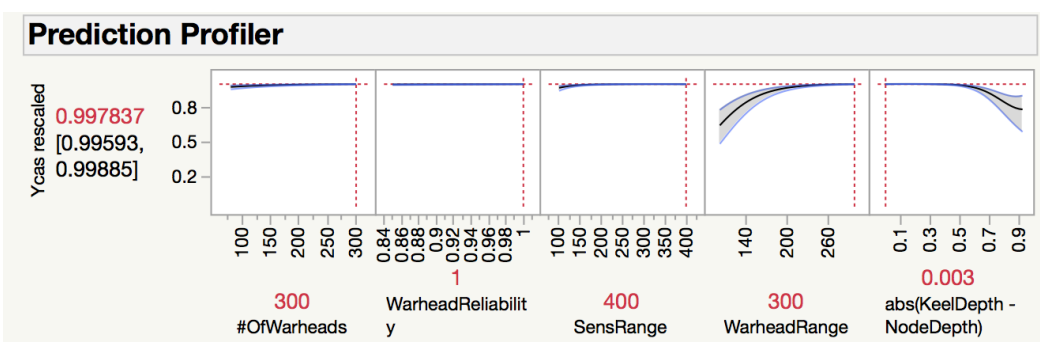
Source	DF	Sum of Squares	Mean Square	F Ratio
Model	10	4042.7872	404.279	322.9724
Error	501	627.1236	1.252	Prob > F
C. Total	511	4669.9108		<.0001*

Sorted Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
abs(KeelDepth - NodeDepth)	-10.45736	0.419774	-24.91	<.0001*
WarheadRange	0.0189106	0.00086	21.98	<.0001*
SensRange	0.0098377	0.000571	17.24	<.0001*
#OfWarheads	0.0113378	0.000779	14.55	<.0001*
(abs(KeelDepth - NodeDepth)-0.33185)*(abs(KeelDepth - NodeDepth)-0.33185)	-14.97824	1.381076	-10.85	<.0001*
(SensRange-250)*(abs(KeelDepth - NodeDepth)-0.33185)	0.0195351	0.002347	8.32	<.0001*
(SensRange-250)*(SensRange-250)	-5.56e-5	7.396e-6	-7.52	<.0001*
(abs(KeelDepth - NodeDepth)-0.33185)*(abs(KeelDepth - NodeDepth)-0.33185)*(abs(KeelDepth - NodeDepth)-0.33185)	23.47858	3.793975	6.19	<.0001*
(SensRange-250)*(WarheadRange-200)	5.8436e-5	0.00001	5.86	<.0001*
WarheadReliability	6.0130527	1.137801	5.28	<.0001*

The prediction profiler in Figure 4.3 shows the relationship of the main effects to the response for a particular set of factor settings. Both Y_{cas} and Y_{det} are values that vary between 0 - 1 (0 - 100%). Since a minefield planner desires a high Y_{cas} value, settings that maximize Y_{cas} are desired. The following factor value settings provide a predicted Y_{cas} = 1.0, `NumberOfWarheads` = 300, `WarheadReliability` = 1.0, `SensorRange` = 400, `WarheadRange` = 300, and abs [`KeelDepth` - `NodeDepth`] = 0.

Figure 4.3. Prediction profiler for Y_{cas} in Case I.



`WarheadReliability` is a linear effect that as it increases, increases the value of Y_{cas}. `NumberOfWarheads, SensorRange`, and `WarheadRange` are increasing functions with diminishing returns to scale. These effects match my intuition that maximizing the number of mines, detection and damage range, and mine reliability increases the likelihood that a transitor would be damaged crossing the minefield. `SensorRange` tapers off as `WarheadRange` becomes greater than it. Per GAMET user's guide, if a detection occurs then the mine will detonate. If the ability to detect a mine is greater than the ability to inflict damage, then it follows that fewer casualties will occur. Lastly, the abs [`KeelDepth` - `NodeDepth`] effect is a decreasing function as the distance between the transitor's keel and the mine increases. This matches my intuition that as the distance between a transitor and a mine increase, Y_{cas} should decrease.

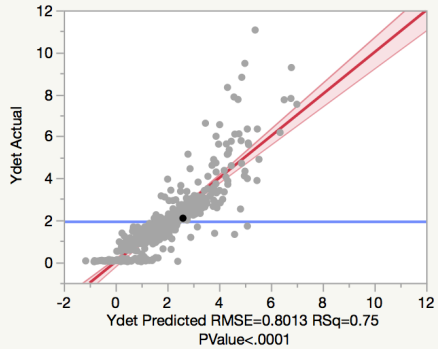
Y_{det} Analysis

Y_{det} values, unlike Y_{cas}, are not constrained to a bounded range so no transformation of the response was needed. The regression output in Figure 4.4 is for the detonations per transitor response variable, Y_{det}. The summary of fit indicates that the model's $R^2 = 0.75$. Six effects explain this variability: 4 main effects and 2 two-factor interactions. The main effects include the following factors, `SensorRange`, `WarheadRange`, `NumberOfWarheads`, `KeelDepth`, and `NodeDepth`.

Figure 4.4. Stepwise regression of Y_{det} in Case I.

Response Ydet

Actual by Predicted Plot



Summary of Fit

RSquare	0.748184
RSquare Adj	0.745192
Root Mean Square Error	0.801338
Mean of Response	1.911377
Observations (or Sum Wgts)	512

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	6	963.4900	160.582	250.0718
Error	505	324.2819	0.642	Prob > F
C. Total	511	1287.7718		<.0001*

Sorted Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
SensRange	0.011306	0.000408	27.68	<.0001*
WarheadRange	-0.010097	0.000613	-16.47	<.0001*
(SensRange-250)*(abs(KeelDepth - NodeDepth)-0.33185)	0.023027	0.001674	13.75	<.0001*
#OfWarheads	0.0058655	0.000558	10.52	<.0001*
(SensRange-250)*(WarheadRange-200)	-7.268e-5	7.119e-6	-10.21	<.0001*
abs(KeelDepth - NodeDepth)	0.4883891	0.151348	3.23	0.0013*

The prediction profilers for Y_{det} seen in Figure 4.5 indicate that the following factor values maximize Y_{det}, `NumberOfWarheads` = 300, `SensorRange` = 400, `WarheadRange` = 100, and abs [`KeelDepth` - `NodeDepth`] = 928. Intuitively, the number of detonations should increase, given an increase in `NumberOfWarheads` and `SensorRange`. Since the sensor is a "cookie cutter" sensor, when a transitor is within sensor range there is a 100% probability of detection and subsequent detonation. Given a low warhead damage range or a greater distance between the transitor and mine, it is more likely the transitor will survive, continuing across the minefield creating and surviving more detonations. It is important for a warfare planner to consider the mine damage range and mine density when planning a minefield.

Figure 4.5. Prediction profiler for Y_{det} in Case I.

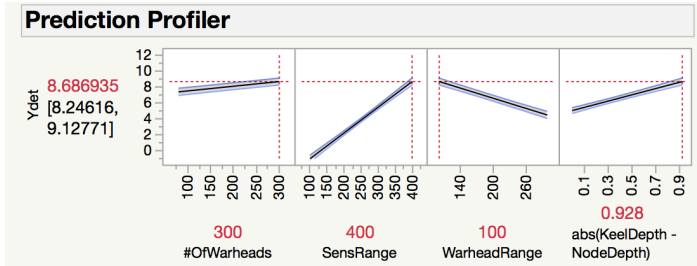
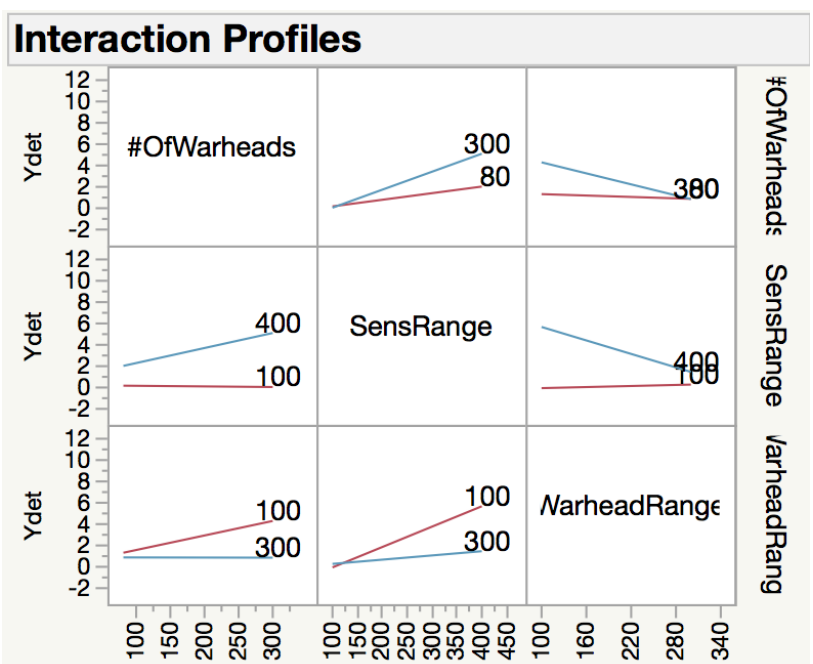


Figure 4.6 is the interaction profiler for Y_{det}. Increasing `SensorRange` yields a higher Y_{det}, but the effect is further amplified when you have more warheads. Likewise, when `SensorRange` is at its low level, increasing `WarheadRange` does not result in more detonations. However, when `SensorRange` is at its high level, increasing the `WarheadRange` results in a linear decrease in Y_{det}. This confirms expectations, because a high `SensorRange` with low `WarheadRange` will result in transitors surviving detonations and subsequently causing more detonations during their transit. A planner should balance sensor range with damage range to avoid detonations that cannot inflict damage on a vessel.

Figure 4.6. Interaction profiler of Y_{det} in Case I.



4.3.3 Case I Results

The Case I analysis confirms that GAMET provides output that is consistent with the GAMET user guide. The relationships between factors also match intuition and MW literature.

Transitor success is defined as a transitor crossing the minefield without inducing a detonation or becoming a casualty as a result. In Case I, `KeelDepth` is an input that is a characteristic of the transitor, and is expected to impact a vessel's success at crossing a traditional minefield in GAMET. `KeelDepth` is significant for predicting the number of casualties and detonations within the abs [`KeelDepth - NodeDepth`] main effect.

A minefield is considered effective when a transitor's passage through it causes a casualty to the transitor. In Case 1, there are five significant factors that impact the minefield's effectiveness as measured by Y_{cas}. In order of explaining variance, they are: `NodeDepth` within the abs [`KeelDepth - NodeDepth`] main effect, WarheadRange, SensorRange, NumberOfWarheads, and `WarheadReliability`. Four significant factors explain the number of detonations, Y_{det}: `SensorRange, WarheadRange`, `NumberOfWarheads`, and `NodeDepth` within the abs [`KeelDepth - NodeDepth`] main effect.

4.4 Case II

In this section, the Case II data is analyzed using descriptive statistics. Second, both response variables, Y_{cas} and Y_{det}, are analyzed using regression analysis.

4.4.1 Case II Descriptive Statistics

Initial analysis indicates the Case II data produced by the simulation conforms to expectations. Y_{cas} ranges from 0 to 0.999. Y_{det} varies between 0 and 1.94.

4.4.2 Case II Regression Analysis

Next, regression analysis of both response variables is conducted. As with Case I analysis, a prediction and an interaction profiler assist in understanding factors and their impact on the respective response variable.

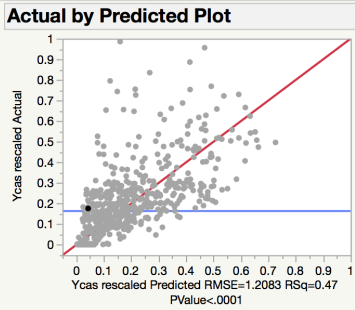
Y_{cas} Analysis

As in Case I, Y_{cas} values are bounded between 0 and 1 and therefore the regression underwent a logit transformation. The regression output in Figure 4.7 displays the Y_{cas} actual versus Y_{cas} predicted (by the model). The summary of fit indicates that 12 effects account for the model's R^2 of 0.47. The model includes 8 main effects and 4 two-factor interactions.

Figure 4.7. Stepwise regression of Y_{cas} in Case II.

Response Logit(Ycas rescaled)

Actual by Predicted Plot



Summary of Fit

RSquare	0.466169
RSquare Adj	0.451131
Root Mean Square Error	1.208253
Mean of Response	-1.62764
Observations (or Sum Wgts)	512

Analysis of Variance

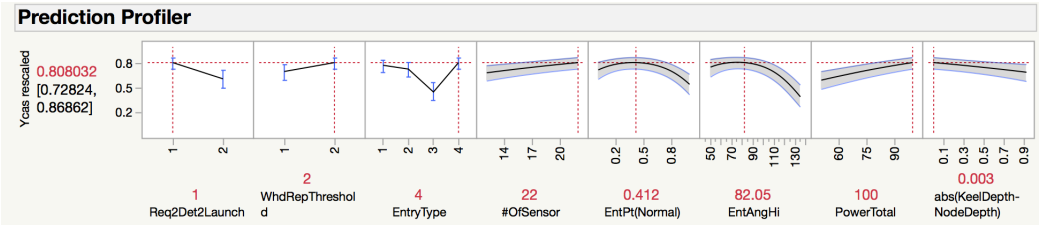
Source	DF	Sum of Squares	Mean Square	F Ratio
Model	14	633.5952	45.2568	31.0004
Error	497	725.5584	1.4599	Prob > F
C. Total	511	1359.1536		<.0001*

Sorted Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
EntryType[4-3]	1.641882	0.160229	10.25	<.0001*
Req2Det2Launch[2-1]	-0.991187	0.107542	-9.22	<.0001*
(EntPt(Normal)-0.5)*(EntAngHi-92.5)	0.0624269	0.007835	7.97	<.0001*
EntryType[3-2]	-1.194521	0.153949	-7.76	<.0001*
PowerTotal	0.0211416	0.00372	5.68	<.0001*
WhdRepThreshold[2-1]	0.5936521	0.107319	5.53	<.0001*
(EntAngHi-92.5)*(EntAngHi-92.5)	-0.000536	0.0001	-5.34	<.0001*
(EntPt(Normal)-0.5)*(EntPt(Normal)-0.5)	-3.366607	0.723703	-4.65	<.0001*
(EntAngHi-92.5)*(abs(KeelDepth-NodeDepth)-0.33185)	0.0385018	0.009682	3.98	<.0001*
#OfSensor	0.0666902	0.018379	3.63	0.0003*
EntryType[2-1]	-0.247911	0.147397	-1.68	0.0932
abs(KeelDepth-NodeDepth)	-0.285235	0.232812	-1.23	0.2211
EntPt(Normal)	-0.115917	0.186625	-0.62	0.5348
EntAngHi	-0.000184	0.002186	-0.08	0.9329

The prediction profiler, Figure 4.8, shows the relationship of the main effects to the response. The following factor value settings maximize the response (i.e., increase Y_{cas}), `Req2DetectToLaunch` = FALSE, `WarheadRepThreshold` = TRUE, `EntryType` = RANDOM, `NumberOfSensors` = 22, `EntryPoint(Normal)` = 412, `EntryAngle(High)` = 82, `PowerTotal` = 100, abs [`KeelDepth` - `NodeDepth`] = 3. Most of the settings match intuition but a few are worth noting.

Figure 4.8. Prediction profiler for Y_{cas} in Case II.



The AUWS model calls for a requirement to have two independent detections prior to launch, and varying `Req2DetectToLaunch` shows the factor is significant. A two-detection requirement results in a decrease in the casualty rate. In the real world, two independent detections should increase targeting accuracy and reduce false detections thereby increasing the casualty rate, however the simulation is not fusing tracking data from the first detection with the second detection. Instead the simulation is "launching" and detonating on the second detection of a transitor. Since some transitors will only be detected once and therefore not cause a detonation, the casualty rate is lower when two detections are required.

The simulation allows for the replenishment of warheads and `WarheadRepThreshold` is shown to be statistically significant. In Case II, 10 transitors attempt to cross the minefield, so it is expected that in DPs where all six torpedoes are fired, the ability to "reload" the pod would be statistically significant.

Maximizing sensors' total power while minimizing power used for receiving communications results in higher casualty rates. Because the range of the torpedo and the communication ranges are both longer than the depth of the minefield, by themselves, `NodeDepth`, `KeelDepth`, `WarheadRange`, and communication factors are not significant.

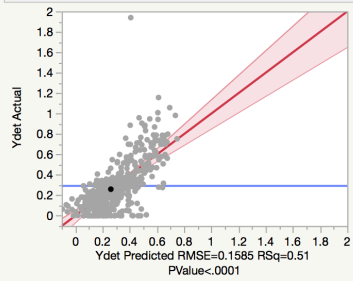
Y_{det} Analysis

The regression output in Figure 4.9 is for the detonation per transitor response. The summary of fit indicates that the model's R^2 = 0.51. Nine main effects and 3 two-way interactions explain this variability.

Figure 4.9. Stepwise regression of Y_{det} in Case II.

Response Ydet

Actual by Predicted Plot



Summary of Fit

RSquare	0.510972
RSquare Adj	0.499211
Root Mean Square Error	0.158544
Mean of Response	0.29232
Observations (or Sum Wgts)	512

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Ratio
Model	12	13.105813	1.09215	43.4492
Error	499	12.542996	0.02514	Prob > F
C. Total	511	25.648809		<.0001*

Sorted Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
WhdRepThreshold[2-1]	0.2353699	0.020325	11.58	<.0001*
PowerTotal	0.0034719	0.000486	7.14	<.0001*
#OfSensor	0.0136869	0.002405	5.69	<.0001*
Req2Det2Launch[2-1]*WhdRepThreshold[2-1]	-0.154591	0.028356	-5.45	<.0001*
PwrReceive	-0.012949	0.002422	-5.35	<.0001*
EntryType{2&3-4}	-0.041259	0.008711	-4.74	<.0001*
(EntAngHi-92.5)*(EntAngHi-92.5)	-6.155e-5	0.000013	-4.73	<.0001*
Req2Det2Launch[2-1]	-0.089589	0.019571	-4.58	<.0001*
Sens&ModReliability	0.7128719	0.160856	4.43	<.0001*
WhdRepThreshold[2-1]*(RepCriteria{1&2&3-4}-0.45313)	0.0700873	0.015879	4.41	<.0001*
EntAngHi	-0.000486	0.000287	-1.70	0.0904
RepCriteria{1&2&3-4}	-0.00479	0.010835	-0.44	0.6586

The following factor value settings increase Y_{det} `Require2DetectionsToLaunch` = FALSE, `WarheadRepThreshold` = YES, `ReplenishCriteria` = BOTH, `EntryType` = RANDOM, `Sensor&ModemReliability` = 1.0, `NumberOfSensors` = 22, `EntryAngleHi` = 90, `PowerTotal` = 100, `PowerReceive` = 5. These settings match intuition.

Figure 4.10. Prediction profiler of Y_{det} in Case II.

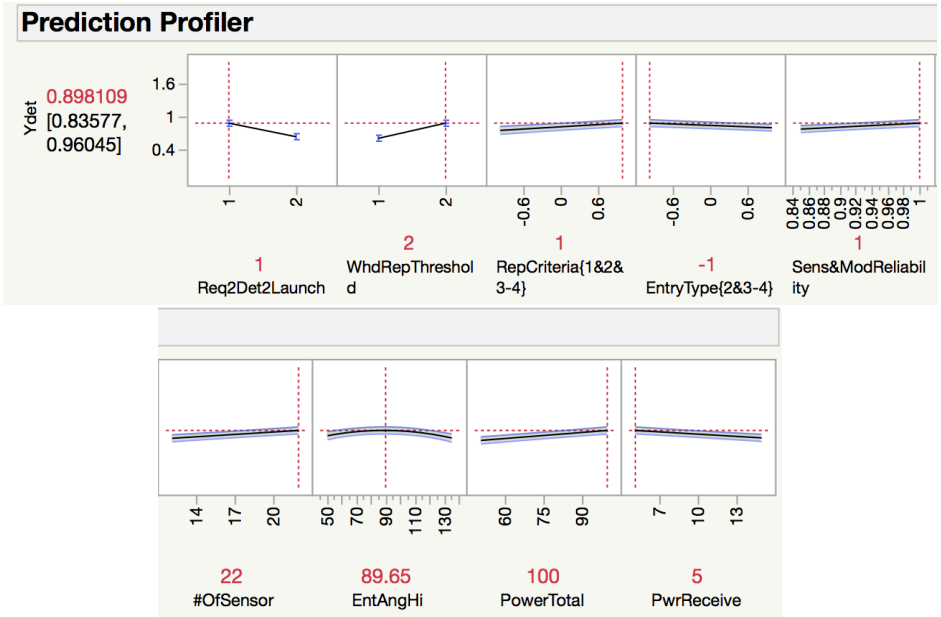
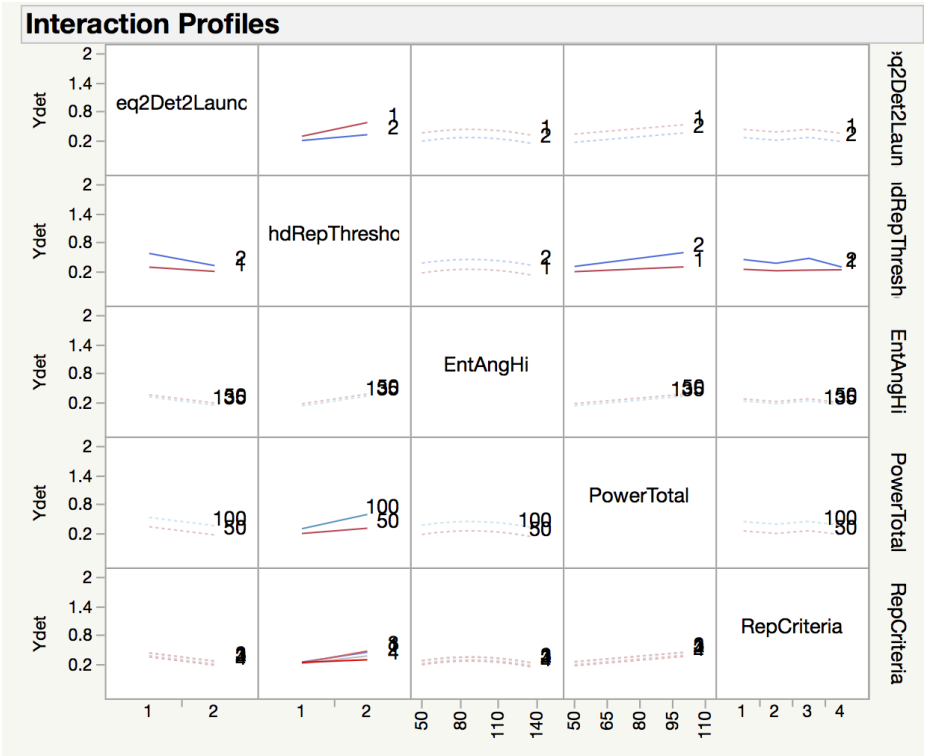


Figure 4.11 is the interaction profiler for Y_{det}. When warheads are being replenished, it results in higher Y_{det} values, but the effect is further amplified when there is higher amount of power for receiving and transmitting detections. This is unlike the effect of requiring two detections prior to launching which lowers Y_{det} values. `RepCritieria` may vary between four options (ADJACENT, LOWPOWER, BOTH, and NONE) but only influences Y_{det} value when warheads can be replenished.

Figure 4.11. Interaction profiler of Y_{det} in Case II.



4.4.3 Case II Results

For both Y_{cas} and Y_{det} predicted responses, the only statistically significant factor that is under the transitor's control is `EntryType`. `EntryType` may vary between four options (BOUNDED, CHANNEL, NORMAL, and RANDOM). It appears that when `EntryType` = RANDOM Y_{cas} and Y_{det} values are slightly increased. This matches intuition as the bounded and channel entries were offset based on the minefield's dimensions, while both random and normal distribution allow transitors to enter along the entire minefield's width.

Significant factors in Case II that impact the minefield's effectiveness as measured by Y_{cas} are: `Require2DetectionsToLaunch`, `WarheadRepThreshold`, `PowerTotal`, `EntryType`, `NumberOfSensors`, and `ReplenishCriteria`. Significant factors that explain Y_{det} are: `Require2DetectionsToLaunch`, `WarheadRepThreshold`, `PowerTotal`, `PowerReceive`, `NumberOfSensors`, `Sensor&ModemReliability`, and `ReplenishCriteria`.

In summary, the analyses of both cases revealed statistically significant factors, the interaction of which and contribution to the response variables match intuition. Insights for Case I, such as tuning `SensorRange` and `WarheadRange` to avoid premature detonations (i.e., detonations outside of the warhead damage range) that result in a decreasing Y_{cas} predicted are informative. Likewise, although in Case II the number of detections before attacking a target was expected to be significant, it was confirmed that `Require2DetectionsToLaunch` was the factor that had the largest impact on the casualty rate. However, it appears that the simulation is not fusing targeting data resulting in a decrease in casualty rates when two detections are required. Additionally, the interactions between replenishing warheads and total power, two detections to launch, receive power, and replenishment criteria show that the simulation appeared to be operating as per its user guide. Regression metamodels in both cases could have value based on their relatively high R^2 values, however Case I metamodels should have better predictive power than the ones for Case II, given the better R^2. Partition analysis was consistent with regression results, but had lower R^2 with more terms, making the regression results more attractive.

CHAPTER 5: Conclusion

GAMET can be used for traditional minefield avoidance and offensive minefield planning; however, the data-entry complexities of manipulating multiple files in order to get GAMET to run are onerous. This study confirms the user guide's recommendation that GAMET is best used by a trained analyst. Even someone who is familiar with mine warfare would struggle to properly use GAMET.

GAMET shows potential, primarily as a tool to evaluate more complex, still theoretical minefields and mine systems where relationships between factors may be ascertained without the requirement of specific probabilities. Although only one AUWS model has been studied, GAMET and a DOE approach can be jointly used to compare and contrast multiple models.

This study's exploration of GAMET yields insights into its operation and input factors with significant impact on the outcomes. It confirms that the simulation behaves as expected per the user guide and intuition. Additionally, GAMET was used to model an AUWS concept proposal and identifies factors significant to its development. A significant benefit of this study is that the VBA tool created here provides a foundation to conduct further DOEs of more complicated mines and sensor technologies.

5.1 Results

5.1.1 State Space Analysis

Programming allows for GAMET to be run millions of times, and using an appropriate DOE approach ensures a better exploration of its state space. Evaluation did not reveal dependencies and potential program-specific problems; all inputs either worked as expected or held no statistical significance as part of the regression analysis. GAMET provides responses for both non-cooperative and cooperative minefields consistent with the user guide. `DepthSigma`, discussed in Chapter 3, is not statistically significant in either case; a non-zero value for `DepthSigma` will ensure that the simulation is able to read `Sensor` and

`Warhead` files.

GAMET significant factors were identified and discussed in Chapter 4. Their identification helped confirm that GAMET is producing data consistent with experience. Many other parameters are not available in GAMET that could be relevant to the successful crossing of a vessel across a minefield. The success of a ship crossing a traditional or AUWS minefield depends primarily on not being detected by the minefield. Therefore, ship characteristics such as induced magnetic field, noise, and counter measures are important considerations to avoid mine damage. Additionally, environmental conditions impact detection too.

Although these attributes are not directly modeled, they may be accounted for by calculating probabilities for sensor detection, warhead damage, and communication ranges based on a specific transitor, mine, and environment. Thus, the construction of the `Sensor, Warhead`, and `Comms Table` probability matrices is critical to a reasonable evaluation of a minefield, and constrains the application of the metamodels.

The Case I metamodels developed through regression are lengthy but explain enough of the variability to make them useful as an operational/tactical planning tool while the Case II metamodels explain less of GAMET's response. However, given the data-entry complexities of manipulating multiple files in order to enable GAMET to run, Case II regression metamodels may allow a strategic planner to draw rough order-of-magnitude comparisons between various factor levels.

Lastly, GAMET was run over 100,000,000 times throughout the course of this study. The programming of a simple yet powerful input file tool makes the application of many DPs and designs possible. Although the VBA user tool was not the subject of the thesis, the tool developed is a valuable contribution to one interested in implementing multiple design points and conducting future research. The source code can be easily modified to accommodate new DOEs.

5.1.2 Applicability to AUWS

An AUWS concept (i.e., Case Study II) was modeled in GAMET. The model produced results that were consistent with expectations, resulting in a high MOE when compared to a minefield with a comparable number of mines. The concept's attributes were modeled

in GAMET with the following exception: the large diameter UUV and the torpedoes were modeled as a single static weaponized node. The ability to model mobile nodes does not exist in GAMET. As a result, the movement of an AUWS weapon pod was not programmed into the VBA user form. Although the weapon pod's movement was not modeled, the range of the mobile mines was modeled by the `Warhead Table`. The table provides warhead pod detonation probabilities based on distance to the transitor, allowing a user to model the increased range associated with a non-static mine.

5.2 Recommendations

5.2.1 Follow-on Research

Many areas for further research exist that may expand the current GAMET body of knowledge. First, utilizing the same NOB design blueprint, alternate factors and levels could be explored, producing new DPs for GAMET, and thereby providing new output data for analyses. Because the study varied factors across limited levels, it is possible that factor significance could be affected if a different range of values is used. Research could explore additional factors such as `EntryType = WAYPT` and `TriggerType = SC` (ship count), both of which support usage of other parameters. Also, other simulation output may be utilized as response variables (i.e., MOEs).

Second, GAMET has two modes of operation available to the user. The planning mode allows the user to select a target MOE value and GAMET provides the number of mines (or nodes) to reach that value. An evaluation of GAMET's planning mode output with the analysis mode could provide more information on the simulation's operation.

Third, GAMET's modeling of the shortest path algorithm used for cooperative node communications could be evaluated. Multiple communication-related output parameters exist that would aid in determining if communication between nodes is modeled accurately.

Fourth, additional AUWS concepts can be modeled and compared. AUWS is a concept that includes the use of sensors in multiple domains (i.e., sea, air, land, space, etc.) and although GAMET cannot be used outside of the sea domain, design parameters (e.g., weapon and sensor quantities and characteristics) and deployment parameters (e.g., depth of pod and sensors) can be evaluated. Additionally, successful operation of AUWS is based on the

ability of the sensors to communicate with one another and the weapon pod(s) within the AUWS. Therefore, examining the power requirements for the sensors and the pod would lead to insights into the functionality and persistence of an AUWS model.

5.2.2 GAMET

A GAMET user guide and scenario-specific formatted sample data files exist. However, they do not detail interactions between various options and resulting mandatory entry fields nor explain simulation output in detail. Most importantly, the lack of a GUI for input parameter entry increases the time required to specify a DP. Also, program-specific terminology and a large number of input variables further hinder casual use. GAMET may require manual modification of up to 13 input file types that capture 65 inputs to create a scenario. Upon completion of a single run, GAMET creates up to 16 output files recording 47 output values. Thus, obtaining output requires the user to manually open select files and then interpret data using limited documentation. Additionally, the user guide stresses that text file creation is key to success, and this was found to be true as potential mistakes were revealed and corrected by the review of these files.

The following is a list of recommendations for improving the GAMET user guide:

1. Incorporate descriptions of each individual data field found within each output file.
2. Provide an alphabetized list of all inputs and outputs.
3. Clarify conditions for use within the simulation for the following parameters in the user guide: `DepthSigma`, `InterShipDeadPeriod`, `PowerTotal`, `PowerReplenish- Threshold`, `ProbOfSensorSurvival`, `ReplenishCriteria`, `Require2DetectionsToLaunch`, `Sensor&Modem Reliability`, `WarheadRep-Threshold`, and `Weaponized`.
4. Provide a hierarchical parameter flowchart similar to the input file flowchart to show the relationship between factor selections. For instance, a useful flowchart might indicate that when `EntryType = WAYPT`, then `Repeat = TRUE / FALSE`. And then if `Repeat = TRUE`, then `Random = TRUE / FALSE`; however, if `Repeat = FALSE`, then `Random = FALSE`. Another example in which factors are "enabled" based on another factor selection, is `NodesCooperative = TRUE / FALSE`.

The following is a list of recommendations for improving the GAMET simulation:

1. Develop a GUI for DP (or scenario) construction and input parameter entry.
2. Allow for seed to be inputted via data file to allow for automated run control via VBA. Although reproducibility of runs is important by utilizing the same start seed, the ability to ensure independence by manipulating start seed outside of the GUI would aid in ensuring non-correlation between runs.
3. Have animation graphics provide consistent representation of minefield size.
4. Allow for modeling of mine (node) movement. Dynamic modeling of this in GAMET would help in modeling prospective or innovative AUWS configurations.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A: Listing of Input Parameters

Batch file <*.scb> (0 factors)

1. `NumberOfFiles`. Not an input factor. Number of scenario files for batch processing.

Scenario file <*.sce> (0 factors)

1. `NumberOfRuns`. Not an input factor. Number of scenario replications/runs.

Planning file <*.pln> (0 factors)

1. Planning mode was not reviewed as part of this thesis. It is a separate execution mode of GAMET.

Boundary file <*.fld> (2 Continuous)

1. `Minefield:xLow`. This value will remain fixed as simulation defines the lower left minefield boundary as 0, 0.
2. `Minefield:yLow`. This value will remain fixed as simulation defines the lower left minefield boundary as 0, 0.
3. `Minefield:xHi`. Continuous. Minefield high x and y coordinate values (yards).
4. `Minefield:yHi`. Continuous. Minefield high x and y coordinate values (yards).

Mine file <*.min> (17 Continuous, 6 Discrete, 7 Categorical)

1. `InterShipDeadPeriod`. Continuous. Defines the minimum time (minutes) allowed for a mine between actuations (i.e., decrementing the ship count). This factor enabled for `TriggerType = SC` (ship count) for both `NodeCooperative = TRUE / FALSE`.
2. `NodeCooperative`. Categorical. Set to either `TRUE / FALSE`. If `TRUE`, then allows modeling of sensor and weapon nodes to communicate with one another. In a cooperative field, non-weaponized nodes (sensors) can pass detection

messages to the nearest weaponized node through the communication network allowing target prosecution by a remote weapon node.

3. `NodeDepth`. Continuous. Mine or Node depth (feet).
4. `NodeGroupId`. Not an input factor. A parameter whose value changes based on the number of node groups and which one is being referenced in the node file.
5. `NodeLine-XY`. Continuous. Node start position x,y and node end position x,y (yards). VBA internal transformation of `Minefield:xHi` and `Minefield:yHi` coordinates. Applicable when `NodePlacement = LINE`.
6. `NodeList-XY`. Continuous (variable number) for each node position (yards). Applicable when `NodePlacement` set to `LIST`. List of horizontal (x-axis) and vertical (y-axis) positions within the minefield.
7. `NodePlacement`. Categorical. Set to one of three options: `LINES, RANDOM`, or `LIST`. `LINE` allows for mines or nodes to be placed in lines as defined in a later input; `RANDOM` allows for mines or nodes to be distributed randomly within minefield boundaries; and `LIST` allows for mines or nodes to be placed by position as defined in a later input.
8. `NodeTypeName`. Not an input factor. A descriptor for proper reference file creation.
9. `NumberOfNodesInLine/Group`. Discrete. The number of mines or nodes in the group. For `NodesCooperative = FALSE`, this number is the total number of mines per line or group. For `NodesCooperative = TRUE` this is the number of sensors per line or group.
10. `NumberOfWarheads`. Discrete. The number of warheads per minefield or mine line if `NodesCooperative = FALSE`. If `NodesCooperative = TRUE`, then for the sensor nodes this value = 0, since they do not have co-located warheads and for the weapon node, this value equals the number of warheads per weapon.
11. `NumberOfNodeGroups`. Discrete. A number between one and unknown maximum. Nodes may be grouped to allow for creation of different groups of mines with unique characteristics within the larger minefield.
12. `NumberofPOFsettings`. Discrete. Sets the number of POF settings for the purpose of creating groups of mines with differing probabilities of firing.
13. `NumberofShipCountSettings`. Discrete. Sets the number of different ship count settings within the mine or node field.

14. `POFsettings`. Continuous. A listing of probability values(s) based on the `NumberofPOFsettings` (i.e., mine groups).
15. `PositionSigma (LIST / LINES)`. Continuous. Varies positions of mines in the x,y (yards) as they are evenly spaced along the `LINE` or as they are positioned at specified points in `LIST`. Applicable when `NodePlacement` set to `LIST` or `LINE`.
16. `PowerDetect`. Continuous. Power used by node to detect a message.
17. `PowerReceive`. Continuous. Power used by node to receive a message.
18. `PowerReplenishThreshold`. Continuous. Low power replenish threshold. Enabled when `ReplenishCriteria = LOWPOWER`.
19. `PowerTotal`. Continuous. Total stored power of a node for the transmission and detection of messages.
20. `PowerTransmitLow`. Continuous. Power used by node to transmit a message at low power.
21. `PowerTransmitHi`. Continuous. Power used by node to transmit a message at high power.
22. `ProbOfSensorSurvival`. Continuous. A value reflecting the probability that the node will survive a warhead detonation. This factor enabled for `NodesCooperative = TRUE` and applicable for non-weaponized sensor nodes.
23. `ReplenishCriteria`. Categorical. Set to one of four options: `ADJACENT, LOWPOWER, BOTH`, or `NONE`. Describes the criteria that non-weaponized nodes will be replenished (with power) when a weapon node is replenished. `ADJACENT` allows for non-weaponized nodes to be replenished when 1-hop from replenished weapon node; `LOWPOWER` allows non-weaponized nodes to be replenished which have reached their `PowerReplenishThreshold`; `BOTH` allows non-weaponized nodes to be replenished based on `ADJACENT` and `LOWPOWER` criteria; and `NONE` means non-weaponized nodes will not be replenished. This factor is enabled for `NodesCooperative = TRUE`.
24. `ReplenishTimeDelay`. Continuous. The time (minutes) between warhead replenishments. This factor enabled for `NodesCooperative = TRUE`.
25. `Require2DetectionsToLaunch`. Categorical. Set to either `TRUE / FALSE`. If `TRUE` then two remote detections of a transitor are required at a weapon node in order for a mobile weapon to be fired at the target. This factor enabled for

NodesCooperative = TRUE.

26. Sensor&ModemReliability. Continuous. A value reflecting the probability that the sensor/modem is functioning at time of usage. In a cooperative field this value is applied to both node and modem.
27. ShipCountSettings. Discrete. A list of ship count settings based on the number of NumberofShipCountSettings.
28. TriggerType. Categorical. Sets the trigger type to POF or SC. When the trigger is POF then the mine or node will fire or launch a weapon with that probability upon target detection. If the trigger is SC then the mine or node will fire or launch a weapon upon detection of that number of targets.
29. WarheadReliability. Continuous. A value reflecting the probability that the warhead will detonate upon activation.
30. WarheadRepThreshold. Categorical. Set to -1 for none or 1 for yes. None means no replenishment after weaponized node is exhausted, while yes means a weaponized node may be "reloaded" after the ReplenishTimeDelay has passed. This factor enabled for NodesCooperative = TRUE.
31. WaterDepth. Continuous. Minefield water depth (feet).
32. Weaponized. Categorical. Set to either TRUE / FALSE. If TRUE, then node is a sensor with a collocated weapon, (i.e., mine). If FALSE, then node is a sensor only.

Sequence file <*.seq> (2 Continuous, 1 Discrete)

1. DamageDelay. Continuous. The amount of time (minutes) transitor will wait before entering the minefield given the previous transitor was damaged.
2. Delay. Continuous. The amount of time (minutes) between transitors given the previous transitor was not damaged.
3. NumberOfTransitors. Discrete. The number of transitors that will attempt to cross minefield during each run of the scenario.
4. TransitorType. Not a factor. Just a descriptor for the transitor, (e.g., ship5, sub3, etc).

Comms file <*.cms> (2 Continuous, 2 Discrete)

1. CommIndexHiPower. Not an input factor. This value references a row in the

probability matrix for successful communications located in the `Comms Table` file.

2. `CommIndexLowPower`. Not an input factor. This value references a row in probability matrix for successful communications located in the `Comms Table` file.
3. `MaxTransmissionAttemptsHiPower`. Discrete. The maximum number of transmission attempts by a single node at high power; must be zero or greater.
4. `MaxTransmissionAttemptsLowPower`. Discrete. The maximum number of transmission attempts by a single node at low power; must be one or greater.
5. `NodePairProcessTime`. Continuous. This is the node-to-node (hop) message processing time (seconds).
6. `TransmissionRetryDelay`. Continuous. This is the amount of time between node transmission attempts (seconds).

Comms Table file <*.cmt> (2 Continuous)

Contains a matrix of probabilities for successful node communications as a function of range (yards) between nodes and power index (low / high). There must be a *.cmt file for each node type in a cooperative field (see Table A.1 for annotated example).

1. `CommRangeHi`. Continuous. The range in yards that a node may communicate successfully with another node using high power.
2. `CommRangeLow`. Continuous. The range in yards that a node may communicate successfully with another node using low power.

Table A.1. Probability of successful communications matrix found in example `Comm` file.

dimension (row, col)	2 5					
range (yards)		0.0	2999.9	3000.0	4999.9	5000.0
high power index	5.0	1.0	0.9	0.9	0.7	0.0
low power index	3.0	1.0	0.8	0.0	0.0	0.0

Sensor file <*.sen> (1 Continuous)

Contains a matrix of probabilities for target detection at specific water depth (feet) as a function of depth difference between sensor and transitor (feet) and horizontal range between sensor and transitor (yards) (see Table A.2 for annotated example). If transitor is detected then detonation occurs.

1. `SensorRange`. Continuous. The range in yards that a node may detect a transitor.

Table A.2. Probability of detection matrix found in example `Sensor` file.

dimension (row, col)	3 6						
horizontal range (yards)		0.0	57.2	114.5	114.5	229.0	229.1
max depth diff (trans above)	260	1.0	1.0	1.0	1.0	0.0	0.0
sensor & trans at same depth	0.0	1.0	1.0	1.0	1.0	1.0	0.0
max depth diff (trans below)	-260	1.0	1.0	1.0	1.0	0.0	0.0

Warhead file <*.whd> (1 Continuous)

Contains a matrix of probabilities for warhead resulting in a casualty as a function of depth difference between warhead and transitor (feet) and horizontal CPA range between warhead and transitor (yards). If the mine is stationary then the max range is the blast radius and if the mine is mobile then this range is the travel range (see Table A.3 for annotated example).

1. `WarheadRange`. Continuous. The range in yards that a mine may inflict damage on a transitor.

Table A.3. Probability of transitor casualty matrix found in example `Warhead` file.

dimension (row, col)	3 6						
horizontal range (yards)		0.0	55.2	110.5	165.8	221.0	221.1
max depth diff (trans above)	260	1.0	0.8	0.7	0.6	0.0	0.0
warhead & trans at same depth	0.0	1.0	1.0	0.8	0.6	0.5	0.0
max depth diff (trans below)	-260	1.0	0.8	0.7	0.6	0.0	0.0

Transitor file <*.trn> (3 Continuous)

1. `DepthSigma`. Continuous. The standard deviation of the transitor's keel depth (feet).
2. `KeelDepth`. Continuous. Transitor's keel depth (feet).
3. `Speed`. Continuous. Transitor's speed (knots).

Track file <*.trk> (9 Continuous, 1 Discrete, 2 Categorical)

1. `EntryAngle (low)`. Continuous. Single/ smaller angle between 0 - 180° that bounds the transitor angle of entry. A single angle may be inputted.
2. `EntryAngle (high)`. Continuous. Larger of two angles between 0 - 180° that bound the transitor angle of entry.
3. `EntryFaceNumber`. Categorical. Entry face number is an integer between 1 and 4 identifying the side of the minefield transitor entry will occur and allows referencing of parameter values unique to entry face.
4. `EntryPoint:x (normal centerpoint)`. Continuous. The x normal centerpoint value (yards). Used when `EntryType = NORMAL`.
5. `EntryPoint:x (bounded low)`. Continuous. The x low coordinate value (yards). Used when `EntryType = BOUNDED`.
6. `EntryPoint:y (channel low)`. Continuous. The y low coordinate value (yards). Used when `EntryType = CHANNEL`.
7. `EntryPoint:x (bounded high)`. Continuous. The x high coordinate value (yards). Used when `EntryType = BOUNDED`.
8. `EntryPoint:y (channel high)`. Continuous. The y low coordinate value (yards). Used when `EntryType = CHANNEL`.
9. `EntrySigma (NORMAL)`. Continuous. This is the standard deviation of the x and y coordinate values (yards) when `EntryType=NORMAL`.
10. `EntrySigma (CHANNEL)`. Continuous. This is the standard deviation of the x and y coordinate values (yards) when `EntryType = CHANNEL`.
11. `EntryType`. Categorical. Entry type may be one of five selections: `BOUNDED, CHANNEL, NORMAL, RANDOM`, and `WAYPT`.
12. `NumberOfEntryOptions`. Discrete. Number of transitor entry options into the minefield. Value is integer value between 1 and 4 dependent on `EntryType`.

Waypoint file <*.wyp> (4 Continuous, 1 Discrete, 2 Categorical)

1. `PositionOfWaypoints (LIST)`. Continuous (variable number). Waypoints are listed by x,y (yards) position.
2. `MaxLegLength`. Continuous. Maximum leg length (yards) that transitor proceeding to waypoint may traverse. Only applicable (needs specified) for `Random = TRUE`.
3. `NumberOfWaypoints`. Discrete. A value representing the number of waypoints located within the minefield boundary.
4. `PositionSigma (WAYPT)`. Continuous. The standard deviation of navigation error in x and y (yards) waypoint coordinates.
5. `Random`. Categorical. Set to either `TRUE / FALSE`. If `TRUE` then transitor will randomly cycle through waypoints (not available if repeat is not selected (i.e., `FALSE`).
6. `Repeat`. Categorical. Set to either `TRUE / FALSE`. If `TRUE` then transitor will continue to cycle through waypoints in order or randomly.
7. `Timeout`. Continuous. Total time (hours) that transitor will cycle between waypoints if `REPEAT` option is selected.

APPENDIX B:
Listing of Output Parameters

Casualty file

1. Number of casualties in a run.
2. Probability of number of casualties over all the runs.

Casualty Density Type file

1. Casualty density distribution.

Comm Stats file

1. Minimum time for successful end-to-end message communication.
2. Maximum time for successful end-to-end message communication.
3. Average time for successful end-to-end message communication.
4. Number of successful communications.
5. Total number of created messages.
6. Percentage of successful communications (number of successful communications per number of created messages).

Cumulative Casualty Density Type file

1. Cumulative casualty density distribution.

Depletion file

1. Number of mines removed (detonated) prior to the start of a particular transitor.
2. Threat at particular depletion level.
3. Number of transitors at particular depletion level.

Detonation file

1. Number of lethal detonations.
2. Total number of detonations.
3. Number of runs and mines of each type in the case.
4. Maximum number of detonations per run.
5. Minimum number of detonations per run.
6. Expected number of detonations per run.
7. Standard deviation of detonations per run.

Distance to Casualty file

1. Maximum distance to casualty.

2. Minimum distance to casualty.
3. Expected distance to casualty.
4. Standard deviation of distance to casualty.
5. Number of casualties per transitor type.

Node Stats Avg file

1. Node location given x and y position.
2. Average power outages per run.
3. Average power remaining at end of each run.
4. Average warhead outages per run.
5. Average number of warheads remaining at end of each run.
6. Average number of warhead replenishments per run.

Scenario file

1. Number of runs.

Stopping Criteria by Type file

1. Number of penetrators by type.
2. Stopping criteria probability.

Threat file

1. Threat profile.
2. Expected casualties by transitor type.

Threat Type file

1. Threat probability (profile) by type.
2. Expected casualties by transitor type.

Time to Casualty file

1. Maximum time to casualty.
2. Minimum time to casualty.
3. Expected time to casualty.
4. Standard deviation of time to casualty.
5. Number of casualties per transitor type.

Transitor Frequency

1. Max time between transitors.
2. Min time between transitors.
3. Mean time between transitors.
4. Standard deviation of time between transitors.

APPENDIX C: Parameters Varied in Case I

1. `CommRangeLow`. Varied between 1000 - 2000 yards. This factor is believed to only be significant for cooperative fields only.
2. `CommRangeHi`. Varied between 3000 - 5000 yards. This factor is believed to only be significant for cooperative fields only.
3. `DamageDelay`. Varied between 0 - 10 minutes. Case study is for single transitor and therefore the amount of time a transitor must wait before entering the minefield given the previous transitor was damaged is expected to not be a significant factor.
4. `Delay`. Varied between 0 - 10 minutes. Case study is for single transitor and therefore amount of time between transitors given previous transitor was not damaged is expected to not be a significant factor.
5. `DepthSigma (Keel)`. Varied between 1 - 10 feet. Although, setting `DepthSigma = 1` foot resulted in unexplained 4 - 18% reduction in Y_{cas}, setting to zero resulted in large number of unexplained zero detonation DPs. It is believed more can be learned about how `DepthSigma` is being utilized if a setting other than zero is utilized.
6. `EntryAngle (low)`. Varied between 45 - 135° using a transformation of 0.9 - 1.0 of `EntryAngle (high)`.
7. `EntryAngle (high)`. Varied between 50 - 135°.
8. `EntryType`. Varied between 1 - 4 to reflect entry types: `BOUNDED, CHANNEL, NORMAL`, and `RANDOM`. Simulation permits a fifth entry option: `WAYPT` however, `WAYPT` is not within the constraints of the case study.
9. `EntryPoint:x (NORMAL centerpoint)`. Varied between 0 - 1 using VBA internal transformation of `Minefield:yHi` for `EntryType = NORMAL`.
10. `EntryPoint:x (BOUNDED low)`. Varied between 0 - 0.5 using internal VBA transformation of `Minefield:yHi` for `EntryType = BOUNDED`.
11. `EntryPoint:y (CHANNEL low)`. Varied between 0 - 0.5 using VBA internal transformation of `Minefield:yHi` for `EntryType = CHANNEL`.
12. `EntryPoint:x (BOUNDED high)`. Varied between 0.51 - 1.0 using VBA internal transformation of `Minefield:yHi` for `EntryType = BOUNDED`.
13. `EntryPoint:y (CHANNEL high)`. Varied between 0.51 - 1.0 using VBA internal

transformation of `Minefield:yHi` for `EntryType = CHANNEL`.

14. `EntrySigma (CHANNEL)`. Varied between 0 - 500 yards for `EntryType = CHANNEL`.
15. `EntrySigma (NORMAL)`. Varied between 0 - 500 yards for `EntryType = NORMAL`.
16. `InterShipDeadPeriod`. Varied between 1 - 10 minutes. This factor is not expected to be significant (used when `TriggerType = SC`), however is expected to only effect cases with more than one transitor.
17. `KeelDepth`. Varied between 1 - 990 feet using a transformation of 0.001 - 0.990 of `WaterDepth` value. This allows the transitor to model a surface as well as a submarine vessel.
18. `MaxTransmissionAttemptsHiPower`. Varied between 1 - 4.
19. `MaxTransmissionAttemptsLowPower`. Varied between 1 - 4.
20. `NodeDepth`. Varied between 0 - 990 feet using a transformation of 0.00 - 0.99 of `WaterDepth` value. This transformation ensures that mine depth is always less than water depth and allows the mine to be placed within realistic value range for close-tethered, moored, and surface mines.
21. `NodeLine-XY`. Varied as a function of `NodePlacement` and `NumberOfWarheads` and internal VBA construction. Individual mine placement is based on uniform distribution with associated `PositionSigma` based on varying `NumberOfWarheads`. Line placement is constant with constant minefield dimensions. Applicable when `NodePlacement = LINES`. Four equally spaced lines are created running parallel to the y axis of the minefield.
22. `NodePairProcessTime`. Varied between 0 - 30 seconds.
23. `NodePlacement`. Varied between `RANDOM` and `LINES`. Case study is for both random and mine line laydown of mines.
24. `NumberOfNodesInLine`. Varied between 20 - 75 mines as a function of `NodePlacement` and `NumberOfWarheads` and internal VBA construction. Applicable when `NodePlacement = LINES`.
25. `NumberOfNodeGroups`. Varied between 1 or 4 as part of internal VBA construction. `NodePlacement = RANDOM / LINES` then `NumberofNodeGroup = 1 or 4`.
26. `NumberOfWarheads`. Varied between 80 - 300. These values are based on historical and presumed minefield densities given greater lethal damage ranges of current mines and a fixed minefield area. The resulting mine density will vary between 1.6 and 6 mines per square mile.

27. `POFsettings`. Varied between 0.85 - 1.0. It is not believed that there is a practical case where mines within a minefield would be set below probability of firing (POF) < 0.85.
28. `PositionSigma (LIST / LINES)`. Varied between 0 - 500 yards. This should only be significant when `NodePlacement = LIST / LINES`.
29. `PositionSigma (WAYPT)`. Parameter varied between 0 - 500 yards. This factor is not expected to be significant since `WAYPT` is not selected as an entry type.
30. `PowerDetect`. Varied between 5 - 15 units of power.
31. `PowerTotal`. Varied between 50 - 100 units of power. Since power relationships are not defined for the simulation, an attempt to vary the parameters over a broad range that captures the assumed relationships / ratios between the various related factors has been made. For instance, `PowerTotal` is greater than `PowerReplenishment Threshold` and `PowerTransmit` will on average be greater than `PowerReceive`.
32. `PowerTransmitLow`. Varied between 5 - 20 units of power.
33. `PowerTransmitHi`. Varied between 10 - 40 units of power.
34. `PowerReceive`. Varied between 5 - 15 units of power.
35. `PowerReplenishThreshold`. Varied between 1 - 25 units of power. This factor is significant for a cooperative minefield when `ReplenishCriteria` does not equal NONE.
36. `ProbofSensorSurvival`. Varied between 0.85 - 1.0. The probable range for a sensor surviving a nearby warhead detonation is unknown, but it is believed it would be designed to be robust. Believed to only be applicable for cooperative minefields, where the sensor is not collocated with the weapon.
37. `ReplenishCriteria`. Categorically varied between `ADJACENT, LOWPOWER, BOTH`, and `NONE`. However, factor is utilized only for a cooperative minefield.
38. `ReplenishTimeDelay`. Varied between 10 - 60 minutes. However, factor is utilized only for a cooperative minefield (i.e., weaponized node).
39. `Require2DetectionsToLaunch`. Varied between `TRUE` and `FALSE`. However, factor is utilized only for a cooperative minefield.
40. `Sensor&ModemReliability`. Varied between 0.85 - 1.0. Realistic values expected of military hardware were used.
41. `SensorRange`. Varied between 100 - 400 yards.
42. `Speed`. Varied between 3 - 60 knots. Covers reasonable range of most maritime

vessels.

43. `Timeout (WAYPT)`. Parameter varied between 0.5 - 5 hours based on example in documentation. Since it is the total time that the transitor will cycle between waypoints if repeat is selected, it is not expected that a significant change in casualty status as a result of repeated transits of the same path will occur (although there is a position error deviation applied to waypoints that will cause path to be altered for each cycle).
44. `TransmissionRetryDelay`. Varied between 0 - 30 seconds.
45. `WarheadRange`. Varied between 100 - 300 yards.
46. `WarheadReliability`. Varied between 0.85 - 1.0. Realistic values expected of military hardware were used.
47. `WarheadRepThreshold`. Alternates between `-1` for none or `1` for yes. However, factor is utilized only for a cooperative minefield (i.e., weaponized node).

APPENDIX D: Parameters Held Constant in Case I

1. `EntryFaceNumber`. 1. Case study is for a transitor crossing perpendicular to mine lines. The alternate sides of the minefield were independently tested.
2. `MaxLegLength`. 50000 yards. This parameter is only used when `EntryType = WAYPT` per experimentation.
3. `Minefield:xHi`. 14,142 yards. Initially, minefield parameters were varied, however balancing the ratio of number of mines to minefield size, i.e., mine density was important for the simulation to run. Too high a density and the simulation crashed, too low a density and no response recorded, i.e., no detonations. The minefield created is a square of 50 square nautical miles in area.
4. `Minefield:yHi`. 14,142 yards. See above.
5. `NodesCooperative`. `FALSE`. Case study is for a non-cooperative field.
6. `NodeList-XY`. N/A. Only applicable when `NodePlacement = LIST`.
7. `NumberofPOFsettings`. 1. The DOE varies this parameter therefore, varying it within the simulation is redundant.
8. `NumberofShipCountSettings`. 1. Only used if `TriggerType = SC`.
9. `NumberOfTransitors`. 1. Case study is for a single transitor. Once one ship is damaged by a minefield it is assumed that the minefield is known and it deters future transitors.
10. `NumberOfEntryOptions`. 1. Because this parameter is not equally applied across all five entry types, i.e., `CHANNEL` and `WAYPT` entry types only allow for one entry option, and `EntryType` is varied as part of the DOE, this factor was held constant at 1.
11. `NumberOfWaypoints`. 4. This parameter is only used when `EntryType = WAYPT` per experimentation.
12. `PositionOfWaypoints (LIST)`. This parameter is only used when `EntryType = WAYPT` per experimentation.
13. `Random`. Parameter fixed at `FALSE` as part of `EntryType`. This factor is not expected to be significant since `WAYPT` is not selected as an entry type.
14. `Repeat`. Parameter fixed at `FALSE` as part of `EntryType`. This parameter is not

expected to be significant since `WAYPT` is not selected as an entry type.

15. `ShipCountSettings`. 1. Only used if `TriggerType = SC`.
16. `TriggerType`. POF. Case study is for single transitor and therefore `SC` trigger is obviated per documentation.
17. `WaterDepth`. 1000 feet. This is a reasonable depth for a minefield in the case study based on submarine and mine operational characteristics.
18. `Weaponized`. TRUE. Case study is for weaponized nodes, i.e., mines. This parameter behaved as expected with zero response for both cooperative and non-cooperative nodes when `Weaponized = FALSE`.

APPENDIX E: Parameters Varied in Case II

1. `CommRangeHi`. Varied between 2000 - 2400 yards. Based on case study.
2. `CommRangeLow`. Varied between 1600 - 2000 yards. Based on case study.
3. `DamageDelay`. Varied between 0 - 10 minutes. Case study is for ten transitors and therefore some transitors will wait before entering the minefield given the previous transitor was damaged.
4. `Delay`. Varied between 0 - 10 minutes. Case study is for ten transitors and should result in transitors waiting between damaged transitors.
5. `DepthSigma (keel)`. Varied between 1 - 10 feet.
6. `EntryAngle (low)`. Varied between 45 - 135° using a transformation of 0.9 - 1.0 of `EntryAngle (high)`.
7. `EntryAngle (high)`. Varied between 50 - 135°.
8. `EntryType`. Varied between 1 - 4 to reflect entry types: `BOUNDED, CHANNEL, NORMAL`, and `RANDOM`. Simulation permits a fifth entry option: `WAYPT` however, `WAYPT` is not within the constraints of the case study.
9. `EntryPoint:x (NORMAL centerpoint)`. Varied between 0 - 1 using VBA internal transformation of `Minefield:yHi` for `EntryType = NORMAL`.
10. `EntryPoint:x (BOUNDED low)`. Varied between 0 - 0.5 using internal VBA transformation of `Minefield:yHi` for `EntryType = BOUNDED`.
11. `EntryPoint:y (CHANNEL low)`. Varied between 0 - 0.5 using VBA internal transformation of `Minefield:yHi` for `EntryType = CHANNEL`.
12. `EntryPoint:x (BOUNDED high)`. Varied between 0.51 - 1.0 using VBA internal transformation of `Minefield:yHi` for `EntryType = BOUNDED`.
13. `EntryPoint:y (CHANNEL high)`. Varied between 0.51 - 1.0 using VBA internal transformation of `Minefield:yHi` for `EntryType = CHANNEL`.
14. `EntrySigma (CHANNEL)`. Varied between 0 - 500 yards for `EntryType = CHANNEL`.
15. `EntrySigma (NORMAL)`. Varied between 0 - 500 yards for `EntryType = NORMAL`.
16. `InterShipDeadPeriod`. Varied between 1 - 10 minutes. It is not expected to be significant.
17. `KeelDepth`. Varied between 1 - 990 feet using a transformation of 0.001 - 0.990

of `WaterDepth` value. This allows the transitor to model a surface as well as a submarine vessel.

18. `MaxTransmissionAttemptsHiPower`. Varied between 1 - 4.
19. `MaxTransmissionAttemptsLowPower`. Varied between 1 - 4.
20. `NodeDepth`. Varied between 0 - 990 feet using a transformation of 0 - 0.99 of `WaterDepth` value. This transformation ensures that mine depth is always less than water depth and allows the mine to be placed within realistic value range for close-tethered, moored, and surface mines. Both sensor and weaponized "pod" are at equal depth.
21. `NodeLine-XY`. Varied as a function of `NodePlacement` and `NumberOfSensors` and internal VBA construction. Individual mine placement is based on uniform distribution with associated `PositionSigma` based on varying `NumberOfNodesInLine`. Line placement changes based on introduced factor AngleIntersect. Applicable when `NodePlacement = LINES`. Produces two lines intersecting at midpoint of minefield whose angle of intersection varies between 20° and 90°.
22. `NodePairProcessTime`. Varied between 0 - 30 seconds.
23. `NumberOfNodesInLine`. Varied between 7 - 11 sensors per line. Applicable when `NodePlacement = LINES`.
24. `POFsettings`. Varied between 0.85 - 1.0. It is not believed that there is a practical case where mines within a minefield would be set below POF < 0.85.
25. `PositionSigma(LIST/LINES)`. Varied between 0 - 500 yards. However, only be significant when `NodePlacement = LIST or LINES`.
26. `PositionSigma (WAYPT)`. Parameter varied between 0 - 500 yards. This factor is not expected to be significant since `WAYPT` is not selected as an entry type.
27. `PowerDetect`. Varied between 5 - 15 units of power.
28. `PowerTotal`. Varied between 50 - 100 units of power. Since power relationships are not defined for the simulation, an attempt to vary the parameters over a broad range that captures the assumed relationships / ratios between the various related factors has been made. For instance, `PowerTotal` is greater than `PowerReplenishmentThreshold` and `TransmitPower` will most of the time be greater than `PowerReceive`.
29. `PowerTransmitLow`. Varied between 5 - 20 units of power.
30. `PowerTransmitHi`. Varied between 10 - 40 units of power.
31. `PowerReceive`. Varied between 5 - 15 units of power.

32. `PowerReplenishThreshold`. Varied between 1 - 25 units of power. This factor is only significant when `ReplenishCriteria` does not equal `NONE`.
33. `ProbofSensorSurvival`. Varied between 0.85 - 1.0. The probable range for a sensor surviving a nearby warhead detonation is unknown, but it is believed it would be designed to be robust. This should not be significant since it is believed to only be applicable for cooperative minefields, where the sensor is not collocated with the weapon.
34. `ReplenishCriteria`. Categorically varied between `ADJACENT, LOWPOWER, BOTH`, and `NONE`.
35. `ReplenishTimeDelay`. Varied between 10 - 60 minutes. These are not realistic values, however, realistic times would have increased processing time hindering the study.
36. `Require2DetectionsToLaunch`. Varied between `TRUE` and `FALSE`. Since the AUWS model depicts this as a defining characteristic, although it sets the factor at `TRUE`, it was varied in order to learn more about the factor.
37. `Sensor&ModemReliability`. Varied between 0.85 - 1.0. Realistic values expected of military hardware were used.
38. `SensorRange`. Varied between 3600 - 4000 yards. Based on case study.
39. `Speed`. Varied between 3 - 60 knots. Covers reasonable range of most maritime vessels.
40. `Timeout (WAYPT)`. Parameter varied between 0.5 - 5 hours based on example in documentation. Since it is the total time that the transitor will cycle between waypoints if repeat is selected, it is not expected that a significant change in casualty status as a result of repeated transits of the same path will occur (although there is a position error deviation applied to waypoints that will cause path to be altered for each cycle).
41. `TransmissionRetryDelay`. Varied between 0 - 30 seconds.
42. `WarheadRange`. Varied between 16000 - 20000 yards. Based on case study for a modern torpedo.
43. `WarheadReliability`. Varied between 0.85 - 1.0. Realistic values expected of military hardware were used.
44. `WarheadRepThreshold`. Alternated between `-1` for none and `1` for yes.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX F: Parameters Held Constant in Case II

1. `EntryFaceNumber`. 1. Case study is for a transitor crossing perpendicular to mine lines. The alternate sides of the minefield were independently tested.
2. `MaxLegLength`. 50000 yards. This parameter is only used when `EntryType = WAYPT` per experimentation.
3. `Minefield:xHi`. 14,142 yards. Initially, minefield parameters were varied, however balancing the ratio of number of mines to minefield size, i.e., mine density was significant. Too high a density and the simulation crashed, too low a density and no response recorded, i.e., no detonations. The minefield created is a square of 50 square nautical miles in area.
4. `Minefield:yHi`. 14,142 yards. See above.
5. `NodeList-XY`. N/A. Only applicable when `NodePlacement = LIST`.
6. `NodePlacement`. LINES. Case study is for a specific pattern of mine deployment.
7. `NodesCooperative`. TRUE. Case study is for a cooperative field.
8. `NumberOfPOFsettings`. 1. The DOE varies this parameter therefore, varying it within the simulation is redundant.
9. `NumberOfNodeGroups`. 3. Case study is for one weapon pod (group) and two lines (groups) of sensors.
10. `NumberOfWarheads`. 6. Case study is for an AUWS weapon pod with 6 torpedoes.
11. `NumberofShipCountSettings`. 1. Only used if `TriggerType = SC`.
12. `NumberOfEntryOptions`. 1. Because this parameter is not equally applied across all five entry types, (i.e., "channel" and "waypoint" entry types only allow one entry option), and the entry options are all varied as part of the DOE, this factor was held constant at 1.
13. `NumberOfTransitors`. 1. Case study is for a single transitor. Once one ship is damaged by a minefield it is assumed that the minefield is known and it deters future transitors.
14. `NumberOfWaypoints`. 4. This parameter is only used when `EntryType = WAYPT` per experimentation.
15. `PositionOfWaypoints (LIST)`. This parameter is only used when `EntryType =`

WAYPT per experimentation.

16. Random. Parameter fixed at FALSE as part of EntryType. This factor is not expected to be significant since WAYPT is not selected as an entry type.
17. Repeat. Parameter fixed at FALSE as part of EntryType. This parameter is not expected to be significant since WAYPT is not selected as an entry type.
18. ShipCountSettings. 1. Only used if TriggerType = SC.
19. TriggerType. POF. Case study is for single transitor and therefore SC trigger is obviated per documentation.
20. WaterDepth. 1000 feet. This is a reasonable depth for a minefield in the case study based on submarine and mine operational characteristics.
21. Weaponized. FALSE for the sensor groups and TRUE for the pod group.

APPENDIX G: Supplementals

The following list of supplemental materials may be obtained by contacting the NPS Dudley Knox Library.

1. `VBA and JMP files.` Includes VBA user forms for executing Case I and Case II DOE. Both user forms include the VBA code for reading the Case I & II NOB design, creating the text files necessary to run GAMET, and the extraction of MOEs output, respectively. Also, the files include the respective NOB spreadsheet with factor level values and respective Y_{cas}, Y_{det}, and Y_{sd} outputs. Lastly, JMP collated output file is included for both Case I & II (with identical output data as NOB Excel spreadsheet).
2. `GAMET V 3.3.0 software.` This is the GAMET simulation software with user guide and default input files.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

Bard W (2013) *Naval minefield modeling and simulation: An examination of General Analytical Minefield Evaluation Tool (GAMET) and other minefield models*. Master's thesis, Naval Postgraduate School, Monterey, CA.

Belton B (2013) *General Analytical Minefield Evaluation Tool (GAMET) User's Guide Version 3.3* (Panama City, FL).

Belton B (2015a) Personal email regarding GAMET operation.

Belton B (2015b) General analytic minefield evaluation tool [PowerPoint slides]. Not presented.

Cioppa TM (2002) *Efficient nearly orthogonal and space-filling designs for high-dimensional complex models*. Ph.D. thesis, Naval Postgraduate School, Monterey, CA.

Cioppa TM, Lucas TW (2007) Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics* 49(1).

DeHollan AN (2015) *Investigating Navy officer retention using data farming*. Master's thesis, Naval Postgraduate School, Monterey, CA.

Department of Defense (2011) *Joint Publication 3-15: Barriers, Obstacles, and Mine Warfare for Joint Operations* (Washington, DC: Government Printing Office).

Everhart D (2013) Office of naval research advanced undersea weapon system update [PowerPoint slides]. Presented at Concepts to Technology - '13, Arlington, VA.

Faraway JJ (2005) *Extending the linear model with R: Generalized linear, mixed effects and nonparametric regression models* (Boca Raton, FL: CRC Press).

Garrold T (1998) Mine warfare introduction: The threat [PowerPoint slides]. Presented at Surface Warfare Officers School Command, U.S. Navy.

Glisson J (2015) *A decison support system for evaluating systems of undersea sensors and weapons*. Master's thesis, Naval Postgraduate School, Monterey, CA.

Hartmann GK (1979) *Weapons that wait: Mine warfare in the U. S. Navy* (Annapolis, MD: Naval Institute Press).

JMP (2015) JMP pro version 12.1.0. URL http://www.jmp.com/en_us/home.html.

Sánchez PJ (2018) Personal email regarding de-synchronization with reference to seeding of GAMET.

Sanchez SM, Lucas TW, Sanchez PJ, Nannini CJ, Wan H (2012) Designs for large-scale simulation experiments, with application to defense and homeland security. Hinkelmann K, ed., *The Design and Analysis of Experiments, Volume 3: Special Designs and Applications*, chapter 12, 413–437 (Hoboken, NJ: John Wiley & Sons, Inc.).

Sanchez SM, Sánchez PJ, Wan H (2014) Simulation experiments: Better insights by design. *Proceedings of the 2014 Summer Simulation Multiconference*, 53:1–53:8, SummerSim '14 (San Diego, CA, USA: Society for Computer Simulation International), http://dl.acm.org/citation.cfm?id=2685617.2685670.

Sanchez SM, Wan H (2009) Better than a petaflop: The power of efficient experimental design. Rossetti MD, Hill RR, Johansson B, Dunkin A, Ingalls RG, eds., *Winter Simulation Conference*, 60–74, WSC '09 (Winter Simulation Conference), ISBN 978-1-4244-5771-7, http://dl.acm.org/citation.cfm?id=1995456.1995470.

Sanchez SM, Wan H (2012) Work smarter, not harder: A tutorial on designing and conducting simulation experiments. Laroque C, Himmelspach J, Pasupathy R, Rose O, Uhrmacher AM, eds., *Proceedings of the Winter Simulation Conference*, 170:1–170:15, WSC '12 (Winter Simulation Conference), http://dl.acm.org/citation.cfm?id=2429759.2429986.

SEED Center (2015) Seed center for data farming. Accessed May 14, 2015, https://my.nps.edu/web/seed/software-downloads.

Vieira H Jr (2012) Nearly orthogonal and balanced mixed 512dp template v1.xls design spreadsheet. Accessed Jul. 21, 2015, http://harvest.nps.edu.

Williams R (2015) Personal conversation regarding AUWS concept of operation.

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California