NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

AN INDEPENDENT ASSESSMENT OF THE ENERGY ENHANCEMENTS TO THE SYNTHETIC THEATER OPERATIONS RESEARCH MODEL (STORM)

by

John J. Renquist

September 2018

Thesis Advisor:	Thomas W. Lucas
Second Reader:	Alejandro Hernandez

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** September 2018	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE AN INDEPENDENT ASSESSMENT OF THE ENERGY ENHANCEMENTS TO THE SYNTHETIC THEATER OPERATIONS RESEARCH MODEL (STORM)	**5. FUNDING NUMBERS**
6. AUTHOR(S) John J. Renquist	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The Synthetic Theater Operations Research Model (STORM) is a stochastic campaign-level simulation model used by the Department of Defense to provide insight into military strategy, capabilities, and campaign analysis. Recent enhancements to STORM incorporate several energy-related features and logistical considerations. The primary focus of this research was to assess and explore the energy-related enhancements to STORM (known as STORM-E) using data farming. That is, embedding STORM-E in an environment using sophisticated designs of experiments and a computing cluster allows researchers to get faster and more comprehensive results. This research assists in model verification and validation efforts by examining the impact of a large number of STORM-E energy-related inputs on multiple model responses. A total of 1,700 campaigns were simulated using an efficient design of experiments. By using an efficient design and a computing cluster, the time to complete the simulations was reduced from an estimated 12,000 years to 121 hours. The new energy features have been shown to impact both logistic and mission-oriented measures in intuitive ways, thus enhancing our confidence in STORM-E's new energy features.

14. SUBJECT TERMS Synthetic Theater Operations Research Model , STORM, data farming, design of experiments, DOE, simulation, campaign analysis, Energy Operations	**15. NUMBER OF PAGES** 65
	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.

AN INDEPENDENT ASSESSMENT OF THE ENERGY ENHANCEMENTS TO THE SYNTHETIC THEATER OPERATIONS RESEARCH MODEL (STORM)

John J. Renquist
Lieutenant Commander, United States Navy
BS, The Citadel, 2006

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 2018

Approved by: Thomas W. Lucas
Advisor

Alejandro Hernandez
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The Synthetic Theater Operations Research Model (STORM) is a stochastic campaign-level simulation model used by the Department of Defense to provide insight into military strategy, capabilities, and campaign analysis. Recent enhancements to STORM incorporate several energy-related features and logistical considerations. The primary focus of this research was to assess and explore the energy-related enhancements to STORM (known as STORM-E) using data farming. That is, embedding STORM-E in an environment using sophisticated designs of experiments and a computing cluster allows researchers to get faster and more comprehensive results. This research assists in model verification and validation efforts by examining the impact of a large number of STORM-E energy-related inputs on multiple model responses. A total of 1,700 campaigns were simulated using an efficient design of experiments. By using an efficient design and a computing cluster, the time to complete the simulations was reduced from an estimated 12,000 years to 121 hours. The new energy features have been shown to impact both logistic and mission-oriented measures in intuitive ways, thus enhancing our confidence in STORM-E's new energy features.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

ARC	Anglo Republic of Carthage
CONOPS	concept of operations
DOD	Department of Defense
DOE	design of experiments
E2O	Expeditionary Energy Office
GUI	graphical user interface
HTML	hyper-text markup language
NPS	Naval Postgraduate School
NOLH	nearly-orthogonal Latin hypercube
OPNAV N81	Office of the Chief of Naval Operations, Assessments Division
POL	petroleum, oil, lubricants
SEED	Simulation Experiments and Efficient Design
STORM	Synthetic Theater Operations Research Model
SWEMP	Swiss Empire

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

Today, military decision makers within the Department of Defense (DOD) must make investment and strategy decisions regarding force structure and weapon system capabilities for the forces of tomorrow. The problem, however, is that these decisions must be made with a limited budget and limited information regarding the future strategic environment. Furthermore, the development of new technologies takes years, sometimes decades, before it reaches initial operational capability. Given these long lead times, a strict budget, and the uncertainty of the future global environment, analysts utilize tools such as campaign analysis to quantitatively explore scenarios to determine what factors and complex interactions have the most significant effect on future outcomes.

Synthetic Theater Operations Research Model (STORM) is a multi-sided, stochastic campaign-level computer simulation that covers military operations across all domains (Group W, 2017a). Recently, the Marine Corps Expeditionary Energy Office (E2O) sponsored several enhancements to STORM that incorporate multiple new energy-related features and logistical considerations. These enhancements include (1) the conditional arrival of fuel storage containers; (2) the representation of dynamically-laid fuel pipelines; (3) naval bases to act as supply issuers in the logistics model; (4) the ability to model the fuel consumption of surface transportation vehicles; and (5) the degradation of ground unit movement due to fuel shortages (Group W, 2017d).

The focus of this research is to assess and explore the energy-related enhancements of STORM (known as STORM-E or version 2.7), specifically the aforementioned items (4) and (5). This research also assists in model verification and validation efforts by examining the impact of a large number of STORM-E energy-related inputs on multiple model responses. This thesis explores the recent energy-related enhancements made to STORM using data farming. That is, STORM-E is imbedded in an environment that allows running sophisticated designs of experiments on a computing cluster. This allows us to run a large number of STORM-E experiments while simultaneously efficiently varying many input variables. We use this capability to test and verify various new features E2O added to STORM. As part of the data farming assessment, we explore modifications to the existing STORM-E scenario PUNIC 21 to seek new insights.

PUNIC 21 is a naval and ground battle between two opponents. These are the allied nations (Blue forces) of the Anglo Republic and Carthage (ARC), and the Red forces of the Swiss Empire (SWEMP) (Bickel, 2014). The scenario is broken into four battle phases: Battle of the Atlantic, Battle of the Mediterranean, Fight for Spain, and Fight for Italy (Group W, 2012). The duration of the engagement is 20 days, incorporating surface, air, and ground engagements in each phase of the battle. In the PUNIC 21 scenario, both sides possess roughly equal military strengths. Both military forces contain air, land, sea, and logistical elements with roughly the same force structures. While smaller than some of the scenarios OPNAV N81 uses, PUNIC 21 contains hundreds of objects that require thousands of inputs to specify their capabilities and behaviors.

Experimental design has been used to conduct research studies in a variety of fields throughout history. With the technological breakthroughs in computing power, the development of high-dimensional computational models, and the amount of data available, decision makers are turning to experimental design in order to conduct experiments within these complicated and complex simulation models in a timely matter. Data farming enables researchers to conduct many more experiments, such as simulated battles, in much less time (Kleijnen et al., 2005). It also allows them to vary a large number of inputs, such as force sizes, weapon capabilities, environmental conditions, concepts of operations (CONOPS), etc., in efficient ways. This enables researchers to determine the relationships among multiple outputs (probability of mission success, length of battle, etc.) in relation to the inputs that were varied (Lucas, 2017). A good design needs to be space-filling, that is, the design points should be scattered throughout the experimental region (Cioppa & Lucas, 2007). On the SEED Center website (see http://harvest.nps.edu), there exists a catalog of good space-filling and nearly orthogonal Latin-hypercube (NOLH) designs that allow experimenters to alter up to 29 factors and create a NOLH design in the units of the problem. For this research, a NOLH design was used to allow for the experimental change of seven factors.

The selection of the variables for this study was relatively easy. Given the recent nature of the upgrades, the STORM developers have only included a few of the energy upgrades into the PUNIC 21 scenario. Prior to STORM version 2.7, when the ground units requested fuel, it was provided to them immediately, regardless of which type of supply vehicle was needed to provide it. In version 2.7, the supply vehicles can now be constrained by their fuel consumption (miles per gallon), their amount of fuel capacity (in tons), and by the number of supply vehicles in each supply convoy. As one can imagine, if a supply truck or train has a very high fuel consumption rate, it may not have enough fuel to reach the ground units, depending on the distance. In addition to these constraints, the analyst can now choose to allow supply vehicles the ability to resupply at alternate fuel sources or whether they must return to the logistic source node. To clarify, these variables were chosen because they are the only ones that are linked within the PUNIC 21 scenario.

The following is a summary of our findings:

- We used an advanced DOE to efficiently explore multiple energy factors in 1,700 simulated campaigns with the STORMFarmer software.
- This research proves the ability to efficiently explore new STORM energy features. By using an efficient design and a computing cluster, it took 121 hours to simulate the 1700 campaigns. Using a brute force full factorial design and a single processor would have required nearly 12,000 years. STORM analysts can use this new capability to provide faster and more comprehensive studies.
- New STORM energy features have been shown to impact both logistic and mission-oriented measures in intuitive ways. This provides enhanced confidence in STORM 2.7's verification.
- In Punic 21, changes in logistic factors lead to an increase in variability across all of our metrics of interest.

Next steps: The new STORM energy features are ready for more extensive testing on an existing real-world classified scenario.

References

Bickel, W.G. (2014). Improving the analysis capabilities of the Synthetic Theater Operations Research Model (STORM). Master's thesis, Naval Postgraduate School, Monterey, CA.

Cioppa, T.M., & Lucas, T.W. (2007). Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics, 49*(1), 45–55.

Group W. (2012). *Punic 21: A New History Overview*. Fairfax,VA: Author.

Group W. (2017a). *STORM: User's manual version 2.7*. Fairfax, VA: Author.

Group W. (2017d). *What's new in STORM version 2.7*. Fairfax, VA: Author.

Kleijnen, J.P., Sanchez, S.M., Lucas, T.W., & Cioppa, T.M. (2005). A user's guide to the brave new world of designing simulation experiments. *INFORMS Journal on Computing*, *17*(3), 263–289.

Lucas, T. W. (2017). *Getting the most from your model through efficient design of experiments.* Unpublished slides, Department of Operations Research, Naval Postgraduate School, Monterey, CA.

ACKNOWLEDGMENTS

First, I would like to thank my thesis advisor, Professor Thomas Lucas, for the assistance and support he provided. Any time I felt overwhelmed or anxious he would sit me down and make sure that I had all the tools I needed to succeed. I can't thank him enough. I would also like to thank my second reader, Associate Professor Andy Hernandez, for all of the guidance and council he provided during this process.

Second, and just as important, I would like to thank SEED Center research associates Stephen Upton and Mary McDonald. What they were able to do in mere days would have taken me many months to accomplish. I seriously could not have completed my thesis without their help. Thank you!

Third, I would like to thank my fellow OR graduates for all the help they provided on numerous agonizing and time-consuming assignments. Rey Cabana and Charlie Deibler, I couldn't have gotten through many of the courses without our study sessions. Brendan Bunn, Joy Layug, and Pat Saluke, your support and knowledge were invaluable to my success. Thank you to all of my classmates!

Fourth, I would like to thank the Navy for sending me to NPS to get a master's degree at one of the best OR schools in the nation. Not many people have an opportunity to get paid while getting an excellent education in a beautiful location.

Lastly, and certainly not least, I would like to thank my wife, Taylor, and our wonderful children: Owen, Munro, Mills, and Sullivan. Through all of the long nights and weekend work, you stayed by my side and gave me the support I needed to succeed. I love and cherish all of you more than words can describe.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

Today, military decision makers within the Department of Defense (DOD) must make investment and strategy decisions regarding force structure and weapon system capabilities for the forces of tomorrow. The problem, however, is that these decisions must be made with a limited budget and limited information regarding the future strategic environment. Furthermore, the development of new technologies takes years, sometimes decades, before it reaches initial operational capability. Given these long lead times, a strict budget, and the uncertainty of the future global environment, analysts utilize tools such as campaign analysis to quantitatively explore scenarios to determine what factors and complex interactions have the most significant effect on future outcomes. Campaign analysis provides a baseline for comparative analysis of future weapon systems and platforms, where campaign outcomes can be credited to differences in model parameters or the concept of operations (Kline, Hughes, & Otte, 2011).

An organization responsible for providing senior Navy and DOD decision makers with analytic support is the Office of the Chief of Naval Operations, Assessments Division (OPNAV N81). OPNAV N81 utilizes the Synthetic Theater Operations Research Model (STORM) to analyze and explore concepts of operations, new technologies and capabilities, and readiness levels using campaign analysis. STORM is a multi-sided, stochastic campaign-level computer simulation that covers military operations across all domains (Group W, 2017a). Recently, the Marine Corps Expeditionary Energy Office (E2O) sponsored several enhancements to STORM that incorporate multiple new energy-related features and logistical considerations. These enhancements include (1) the conditional arrival of fuel storage containers; (2) the representation of dynamically-laid fuel pipelines; (3) naval bases to act as supply issuers in the logistics model; (4) the ability to model the fuel consumption of surface transportation vehicles; and (5) the degradation of ground unit movement due to fuel shortages (Group W, 2017d).

The primary focus of this research is to assess and explore the energy-related enhancements of STORM (known as STORM-E or STORM 2.7), specifically the aforementioned items (4) and (5). This research also assists in model verification and validation efforts by examining the impact of a large number of STORM-E energy-related inputs on multiple model responses. In addition, exploring these energy-related modifications in an existing STORM-E scenario (named PUNIC 21) may help us gain new insights on the value of including energy features in a campaign model.

B. RESEARCH QUESTIONS

The primary goal of this thesis is to determine whether the recently added STORM-E energy-related factors affect the campaign outcome of the unclassified, pre-installed PUNIC 21 scenario. As a result, this research is guided by the following questions:

1. Do the energy-related enhancements incorporated into the STORM-E model work as intended?

2. Are there certain conditions or logistical constraints that cause the simulation to crash?

3. Do the model runs give invalid results under certain conditions?

4. Is the model particularly sensitive to certain inputs or parameters, or parameter combinations?

5. Another goal is to explore the questions on a real-world, classified scenario.

C. METHODOLOGY

This thesis explores the recent energy-related enhancements made to STORM using data farming. That is, STORM-E is imbedded in an environment that allows running sophisticated designs of experiments on a computing cluster. This allows us to run a large number of STORM-E experiments while simultaneously efficiently varying many input variables. We use this capability to test and verify various new features E2O added to STORM. As part of the data farming assessment, we explore modifications to the existing

STORM-E scenario PUNIC 21 to seek new insights. Following the assessment, we may be interested in a search for solutions (campaign strategies and tactics) that are robust to sources of uncertainty related to energy resources and logistics.

D. BENEFITS OF STUDY

This research assists in model verification and validation efforts by examining the impacts of a large number of STORM-E energy-related inputs on multiple model responses. This effort also identifies aspects of STORM-E that need improvement and/or provides expanded confidence in STORM-E's ability to produce valuable campaign-level insights. Additionally, a comprehensive model assessment of the energy-related features in STORM-E is conducted using the findings from our data farming experiments. This provides valuable energy-related insight to the campaign analysis process conducted by the DoD.

E. LITERATURE REVIEW

The primary sources of research for this thesis are scholarly articles and STORM user manuals, which serve as a reference for conducting a design of experiments within the STORM simulation model.

The *User's Manual* is a practical guide for running the model and was created for the STORM end user (Group W, 2017a). Along with providing operating instructions the user can use while interacting with STORM, it presents various input tools and output tools that the user can implement while running different designs. The *Analyst's Manual* was created for individuals who employ the simulation as a campaign-level tool in order to produce reasonable and credible results for top-level decision makers (Group W, 2017b). It is organized to provide analytical modeling and simulation discussion across a range of experience levels depending on the knowledge the user requires. The *Programmer's Manual* is by far the most technical of the STORM manuals and offers guidance to those individuals who are concerned about developing or designing source code (Group W, 2017c). The newest version of STORM provides many new features, specifically energy-related enhancements. These and many other upgrades are described in detail within the "*What's New in Version 2.7*" document which serves as an add-on to the previously mentioned User's Manual (Group W, 2017d).

To gain an understanding of the usage and implementation of experimental design, this researcher consulted numerous documents created by members of the Simulation Experiments and Efficient Design (SEED) Center at the Naval Postgraduate School (NPS). In multiple journal articles, NPS Professors Thomas Lucas, Suzan Sanchez, and Thomas Cioppa have addressed the benefits of using a design of experiments (DOE) and their utilization to improve efficiency when learning from a simulation (Kleijnen et al., 2005). The utilization of a DOE and implementation within the STORM framework is based on the work of Lucas and Sanchez (Sanchez et al., 2012). Their research has been used on proof-of-concept theses and papers and provides the foundation for this thesis. (See http://harvest.nps.edu for more information).

Additionally, several NPS master's theses have utilized a design of experiments within the STORM framework. Bickel (2014) employed a small factorial experiment using STORM's framework and built-in training scenario Punic 21 to provide N81 with a quick way to produce robust estimates for future force structure tradeoffs. Importantly, the process of running the design was automated and enabled parallel execution. Seymour (2014) sought "to evaluate the tradeoffs between the number of replications and the probability of coverage of associated confidence intervals." His research allowed N81 to capitalize on STORM's full potential depending on the decision-making time-line. Although most research used the training scenarios with the STORM framework, Morgan et al. (2017) mention a small proof-of-concept study, which utilized a classified scenario. Cobbs' thesis (2018) built upon the work of Bickel (2014). Cobbs conducted a design of experiments with a more sophisticated and fully mature classified scenario. That scenario was previously used to inform analysts from N81's staff about future resourcing decisions, whereas, Cobb's research focused on command and control thresholds within the Campaign 2020 scenario.

II. STORM OVERVIEW

This chapter provides the reader with a high-level description of STORM and describes some of its key features. This chapter only addresses critical features in STORM that are essential to this research. For a more in-depth review of STORM, the reader should review the reference materials mentioned in the literature review section. Lastly, this chapter provides a description of the Punic 21 scenario, including the force structures, objectives, and mission parameters of the two sides involved.

A. OVERVIEW OF STORM

STORM is a multi-dimensional stochastic computer simulation of joint military operations across all domains; land, maritime, air, and space (Group W, 2017a). Originally created for the Air Force by the HQ USAF Studies, Analysis and Assessments Division (HQ USAF/A9), significant advancements in recent years, including upgrades to naval combat and expeditionary warfare capabilities, has led to the adoption of STORM by multiple services, specifically the United States Navy and Marine Corps. A Navy organization that utilizes STORM and provides analytic support to these services' decision makers is the Office of the Chief of Naval Operations, Assessments Division (OPNAV N81). OPNAV N81 primarily utilizes STORM as a campaign analysis tool. It provides senior decision makers with reasonably quick answers to campaign-level questions regarding force structures, weapon system capabilities, and concepts of operations (CONOPS) in a joint warfighting context (Group W, 2017c). With this in mind, STORM is an extremely valuable asset to DoD policy makers—as well as to those in the acquisition and operational communities.

B. KEY FEATURES OF STORM

1. STORM's Design

Initially, STORM was designed as a simulation model that would allow a user to quickly and inexpensively change key features in a campaign-level, joint military operation to determine the effect those changes would have on simulated campaign outcomes.

Keeping that in mind, the developers designed STORM as a data-driven construct, meaning the user specifies the representations and entities involved in the simulation (Group W, 2017b). The importance of the entities in STORM being data driven is that they can be easily changed by the user. This allows the user to easily respond to changes in requirements, technology, or funding that often arise when running complex simulations.

Each STORM scenario is comprised of hundreds of input files. Each of these files contains information that when combined deal with all aspects of the scenario, such as the probability of a weapon system hit, resupply limits, environmental conditions, and concepts of operations (CONOPS), etc. (Group W, 2017b). This analyst's research is conducted on a built-in STORM scenario, known as Punic 21. A detailed description of Punic 21 is provided later in this chapter.

2. STORM-The User

STORM contains many tools that are critical for conducting analysis. All of these tools can be found in STORM's Studytool, which employs a Graphical User Interface (GUI) (see Figure 1). This gives the user a central application to access all the components needed to setup, execute, and visualize the results (Group W, 2017a). Each of these tools can be broken into three separate functional areas: input, execution, and output. As one can surmise, input refers to the input data files that are built in the user-generated scenario, known within STORM as a study. Each study contains the pertinent data files that are then pushed into STORM when a simulated run is executed. These can be easily retrieved under the study manager tab in STORM's GUI (see Figure 1).

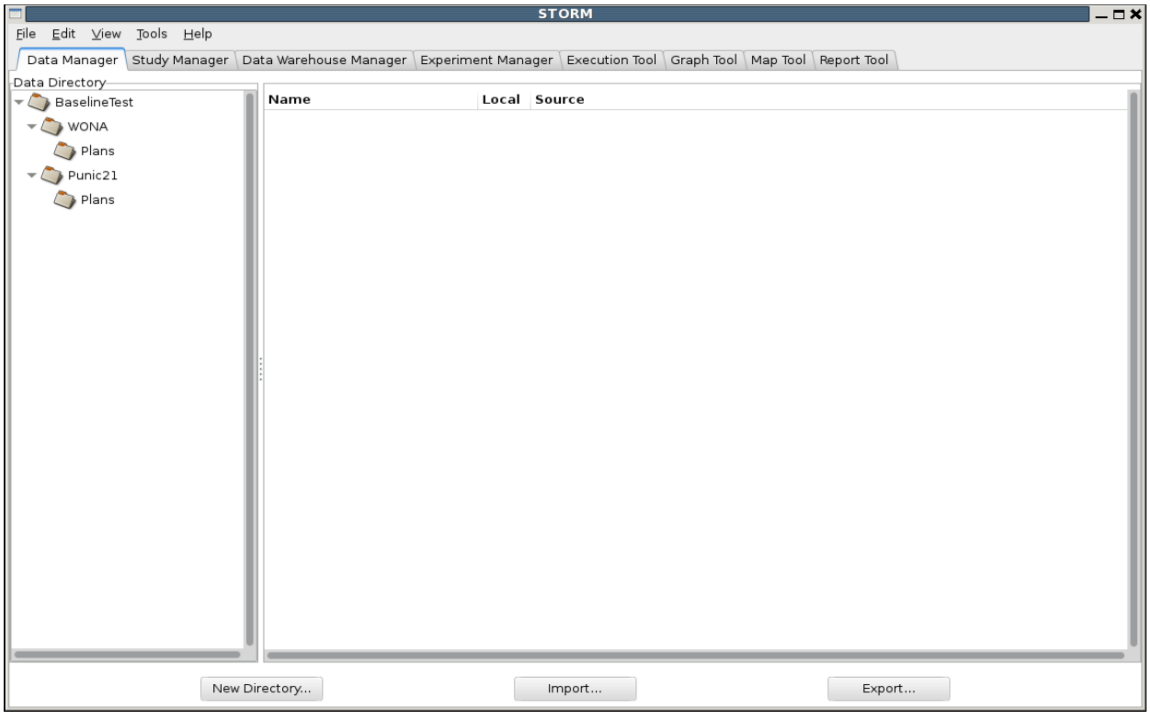


Figure 1. STORM's Studytool GUI. Source: Group W (2017a).

In addition to the input functional area, the execution and output interfaces can also be quickly accessed through STORM's GUI to complete simulation runs or to perform post-run analysis. Within the execution functional area, there are numerous customizable options, such as the amount of simulation runs the user wants to conduct or whether the user wants to compress the output files. The output functional area contains three user-friendly tools that allow the user to view and analyze the results of a study (Group W, 2017a). These are the Graph Tool, Map Tool, and the Report Tool.

a. Graph Tool

The Graph Tool is an application the user can use to organize and display the results of a study (Group W, 2017a). These results can be displayed in simple graphic form, typically bar graphs or line graphs, or in the form of data tables. These data tables can also be transferred to a separate statistical application, such as R or JMP, which allows the user to conduct a more thorough analysis of the output (see Figure 2).

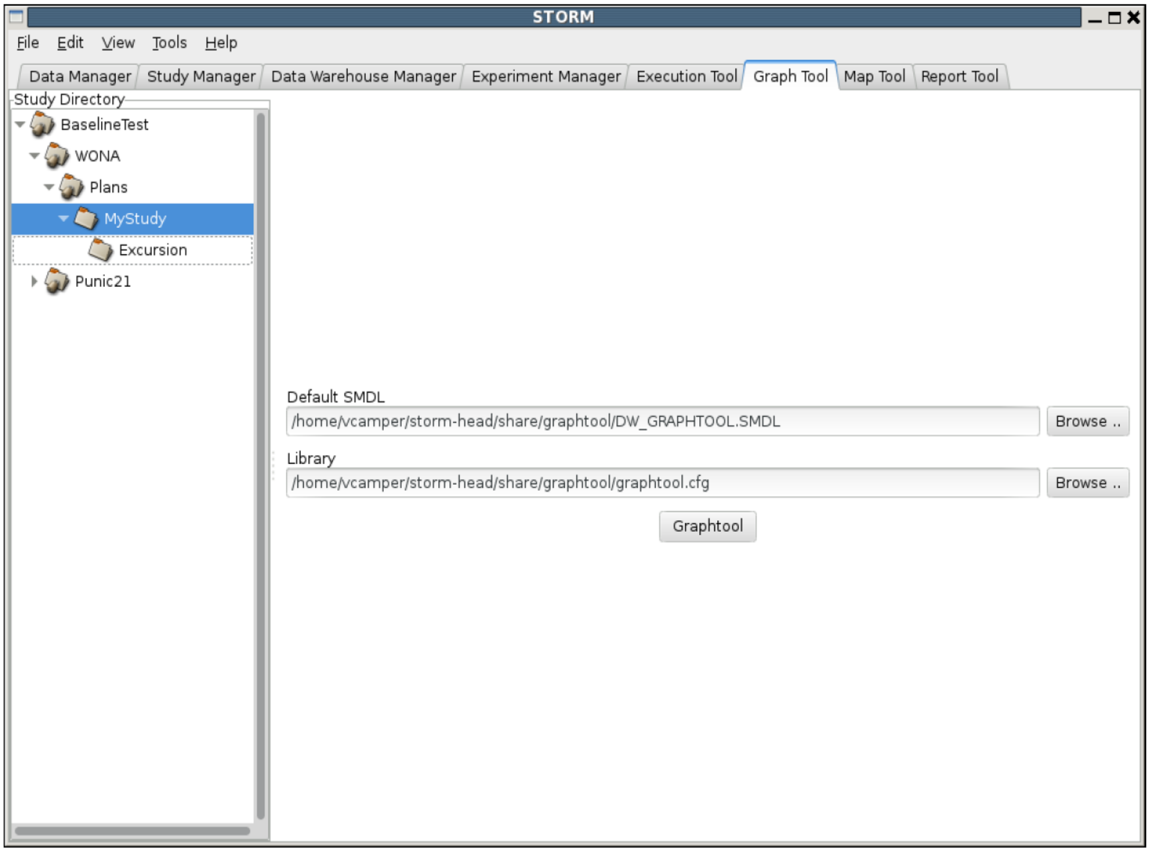


Figure 2. STORM's Graph Tool interface. Source: Group W (2017a).

b. Map Tool

The Map Tool is an application that visually displays an illustration of user-chosen assets within a geographical representation (Group W, 2017a). The user can select any number of assets through field filters, such as air, ground, sea, etc., and then watch these assets interact as the campaign unfolds (see Figure 3). The user can also choose to view support and mobility assets such as supply trains or fuel trucks. Each run, however, must be completed prior to utilizing the Map Tool, as it does not work until the simulation has completed running.

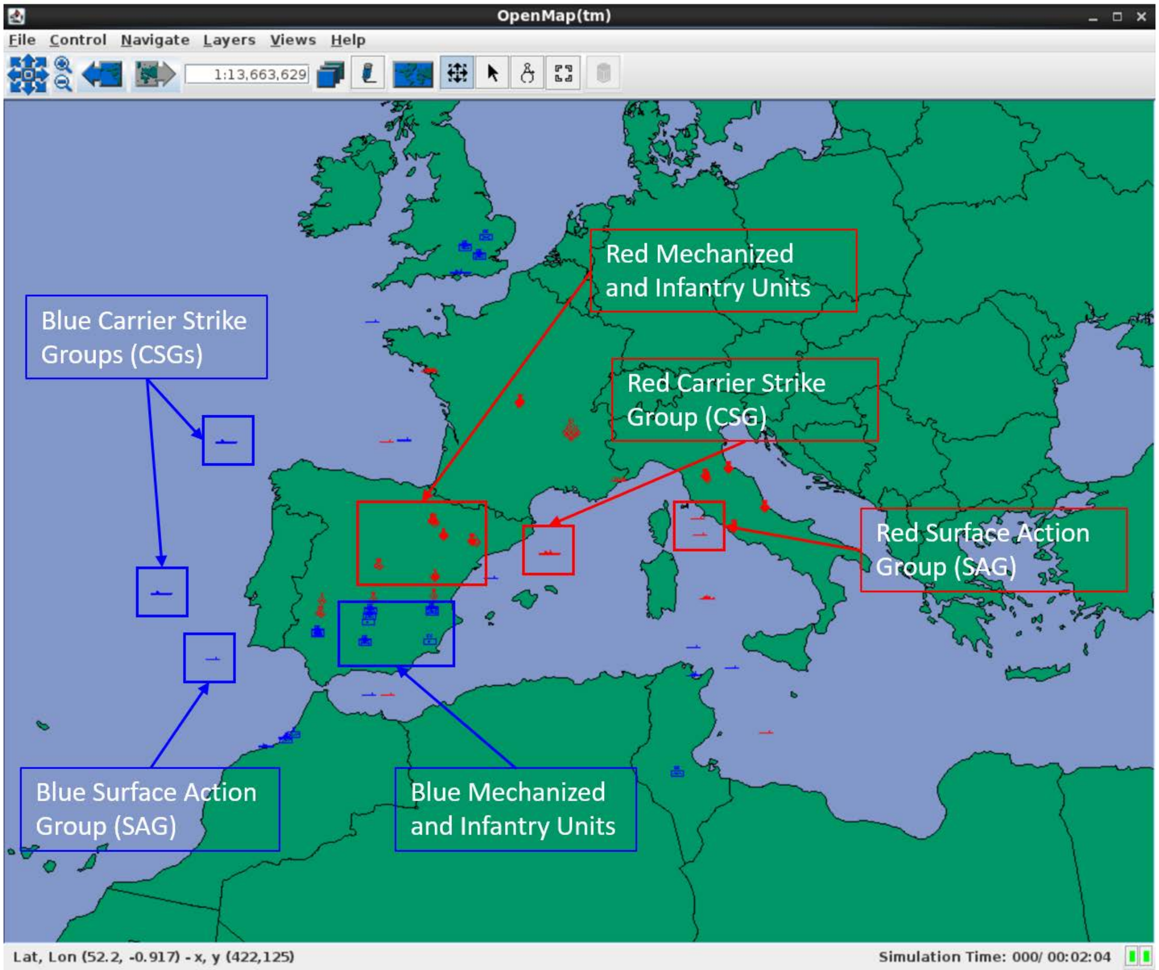


Figure 3. STORM's Map Tool interface. Source: Group W (2017a).

c. *Report Tool*

The Report Tool application allows users to extract certain results from the data output into tabular form, such as a comma separated file (.csv) or hyper-text markup language (HTML) (Group W, 2017a). This differs from the graph tool because the user can pick specific aspects of the simulation to include in the file. For example, the user can pull out information on the number of supply vehicles, the daily fuel consumed by these vehicles, and which type of supply vehicle (road, rail, or sea) (see Figure 4). Although this is incredibly useful, each STORM simulation run generates hundreds of gigabytes of output data, which makes it difficult to identify useful information even with an experienced user.

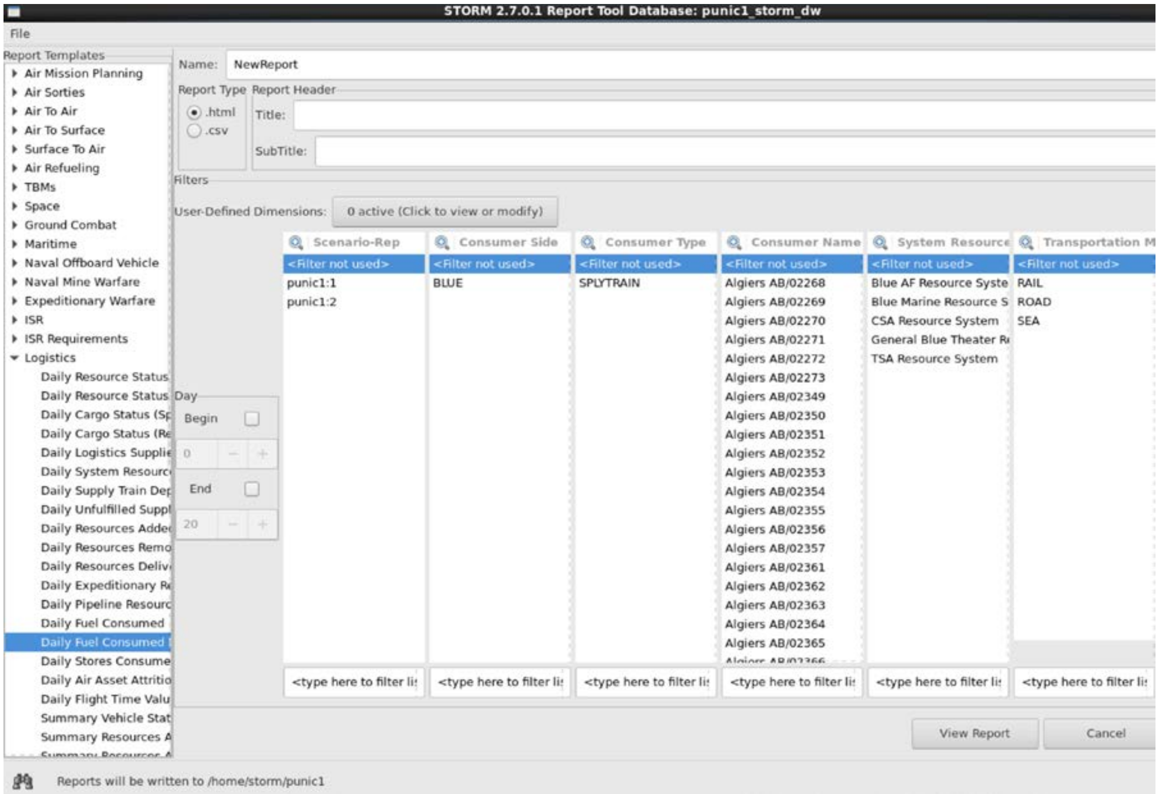


Figure 4. STORM's Report Tool interface. Source: Group W (2017a).

C. STORM–PUNIC 21

STORM contains two unclassified scenarios that were designed to allow users to test, experience, and assess the various features of STORM's functionality; WONA (War of Northern Aggression) and PUNIC 21. However, the recent energy enhancements in the current version of STORM have only been included within the PUNIC 21 scenario, and therefore, PUNIC 21 is the test scenario for this thesis. This section is designed to provide a brief overview of the PUNIC 21scenario.

1. Current Situation and Phases of Battle

PUNIC 21 is a naval and ground battle between two opponents. These are the allied nations (Blue forces) of the Anglo Republic and Carthage (ARC) and the Red forces of the Swiss Empire (SWEMP) (Bickel, 2014). The Swiss Empire is determined to seize control and make the entire Iberian Peninsula part of the empire, causing tensions to rise between these nations (Group W, 2012). The scenario is broken into four battle phases: Battle of the Atlantic, Battle of the Mediterranean, Fight for Spain, and Fight for Italy (Group W, 2012). The duration of the campaign is 20 days, incorporating numerous surface, air, and ground engagements in each phase of the battle. Figure 5 shows a geographic illustration of the occupied territories in the PUNIC 21 scenario.

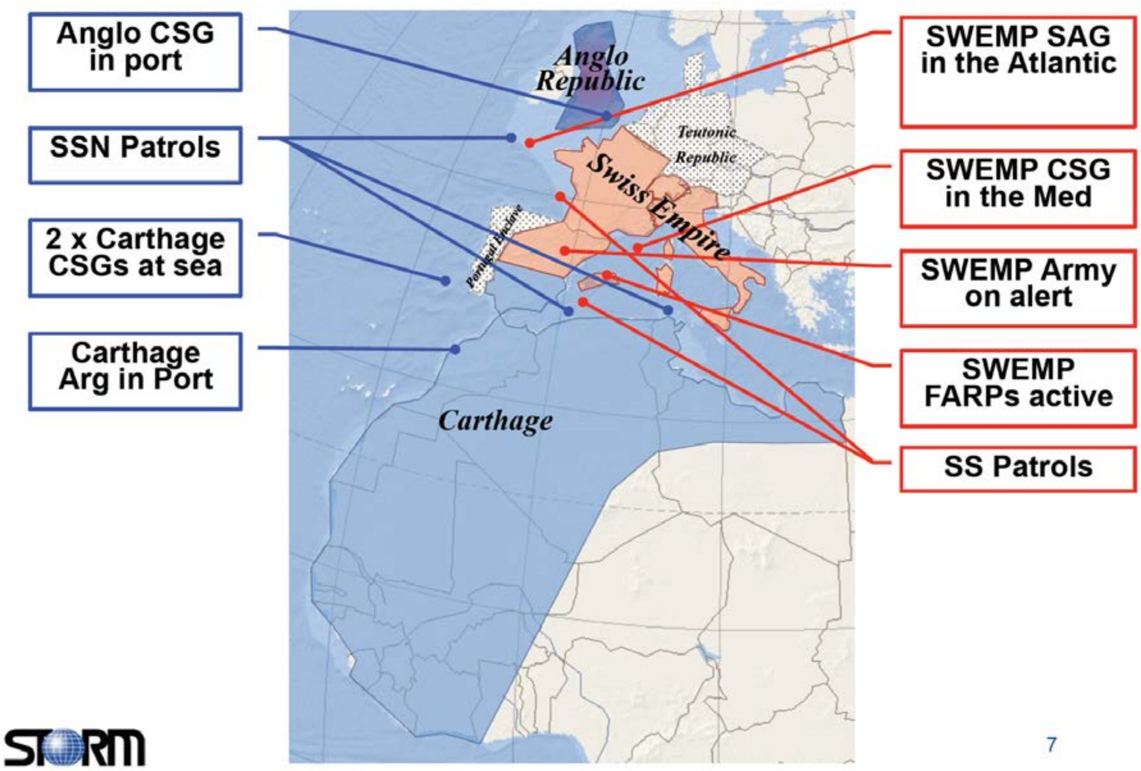


Figure 5. Occupied territories in STORM's PUNIC 21 scenario. Source: Group W (2012).

2. Order of Battle

In the PUNIC 21 scenario, both sides possess roughly equal military strengths. Both military forces contain air, land, sea, and logistical elements with roughly the same force structures. The maritime and air force structures for both the Blue and Red forces can be seen in Table 1 and Table 2, respectively. While smaller than some of the scenarios N81 uses, PUNIC 21 contains hundreds of objects that require thousands of inputs to specify their capabilities and behaviors.

Table 1. A list of the Blue and Red maritime forces for the PUNIC 21 scenario. Source: Bickel (2014).

Blue Navy	Quantity	Red Navy	Quantity
CV (Carrier)	3	CV	1
LHD (Amphibious Assualt)	3	LHD	0
CG (Guided Missile Cruiser)	8	CG	12
DDG (Guided Missile Destroyer)	24	DDG	27
MIW (Counter-Mine)	2	MCM	0
SSN (Attack Submarine)	10	SSN	11
SSGN (Guided Missile Submarine)	1	SSGN	0
Combat Vessels	51	Combat Vessels	50
CLF (Combat Logistic Force)	11	CLF	2
CLF Oiler	6	CLF Oiler	
MRF-N (Mobile Riverine Forces)	120	MRF-N	100
MRF-M	40	Total MRF	100
MRF-EW	15	AEW	3
MRF-Tanker	15	MPA	8
Total MRF	190	Vertical Assault	0
AEW (Airborne Early Warning)	9		
MPA (Maritime Patrol)	12		
Vertical Assault	40		

Table 2. A list of the Blue and Red Air Forces for the PUNIC 21 scenario. Source: Bickel (2014).

Blue Air	Quantity	Red Air	Quantity
MRF	138	MRF	144
MRF-EW	12	MRF-EW	10
FTR	70	FTR	64
BOMBER	32	BOMBER	32
Combat Aircraft	252	Combat Aircraft	250
Tanker	36	Tanker	0
AEW	12	AEW	10
HVA (ISR)	8	HVA (ISR)	8
UAV (ISR)	16	UAV (ISR)	16
AIRLIFT	24	AIRLIFT	24
Total Aircraft	348	Total Aircraft	308
		SSM	180
		SRBM	40
		IRBM	24
		Total Missiles	244

The current version of STORM, version 2.7, incorporates many of the energy-enhanced features in the PUNIC 21 scenario, however, not all of them. In the scenario, only ground unit movement is affected by the logistical support, or lack thereof, provided to them. Therefore, this thesis concentrates on the ground units and their respective logistical support within the Battle for Spain and the Battle for Italy. Figure 8 and Figure 9 illustrate the order of battle and the forces involved during each of those phases of the campaign, respectively.

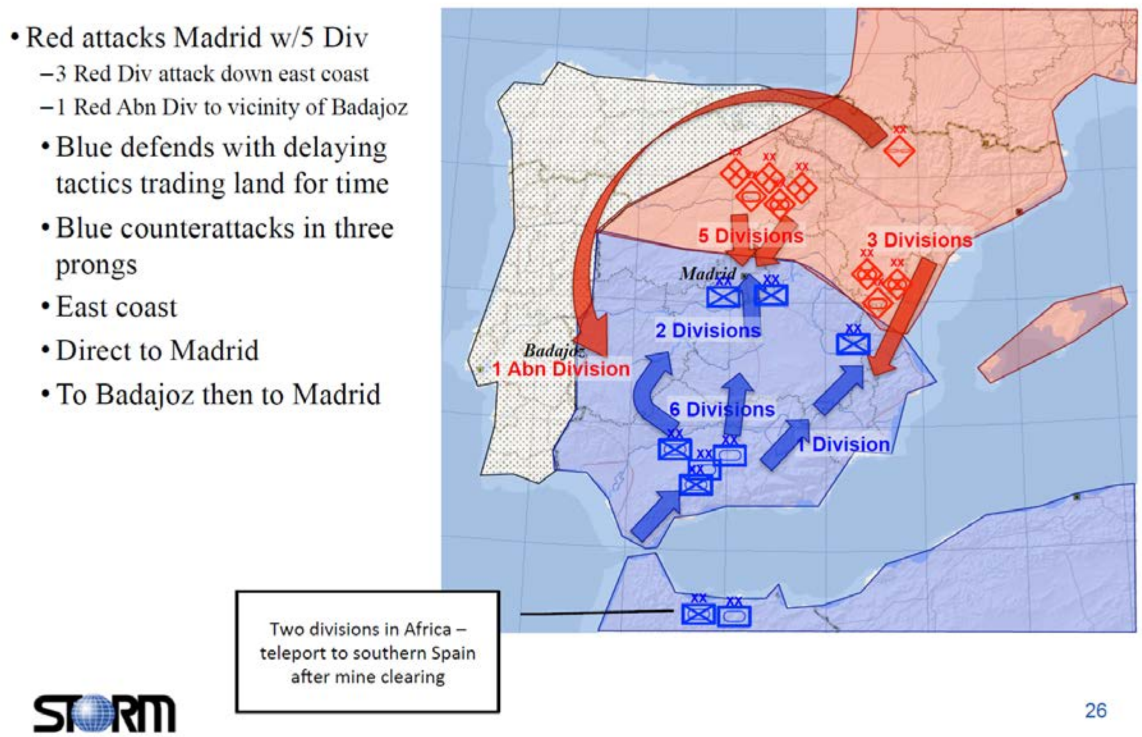


Figure 6. Battle for Spain in STORM's PUNIC 21 scenario. Source: Group W (2012).

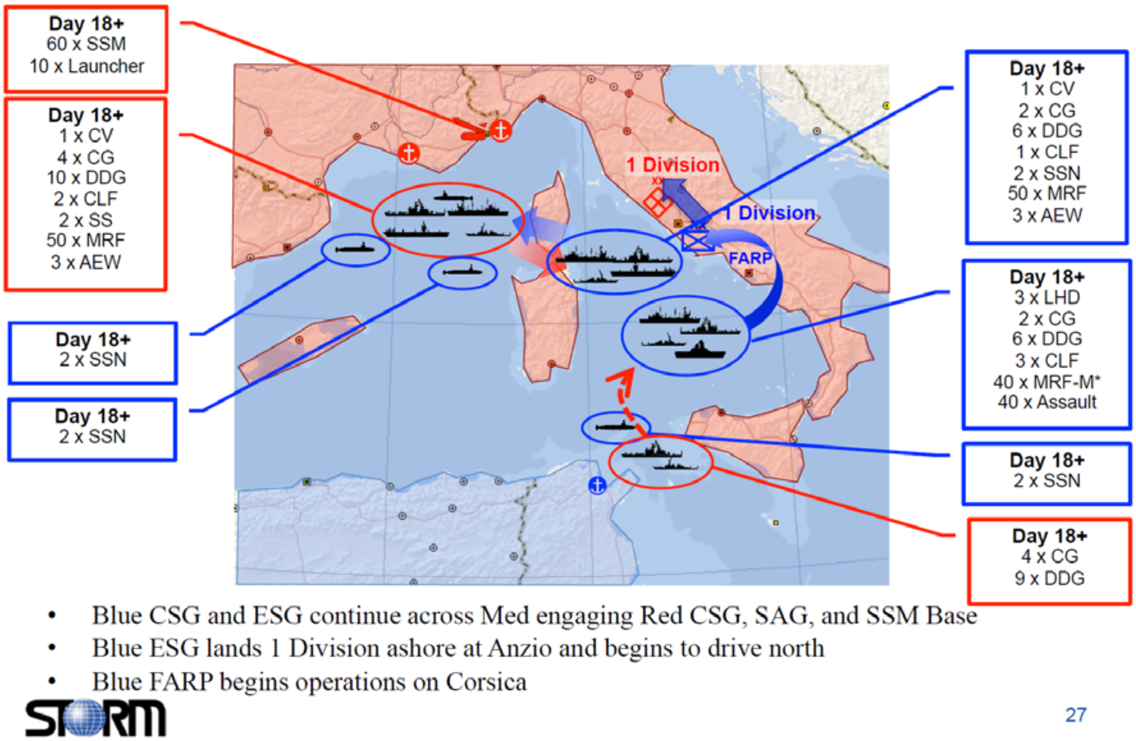


Figure 7. Battle for Italy in STORM’s PUNIC 21 scenario. Source: Group W (2012).

III. DESIGN AND EXECUTION

This chapter provides a general overview of the purpose and implementation of a design of experiments to study a simulated campaign. It also provides a detailed explanation of how to run data farming experiments within the STORM framework (Upton, 2018). Lastly, it identifies the carefully chosen factors of interest and their range, or set (if categorical), of values we utilized when running the simulation experiments.

A. DESIGN OF EXPERIMENTS OVERVIEW

Experimental design has been used to conduct research studies in a variety of fields throughout history. With the technological breakthroughs in computing power, the development of high-dimensional computational models, decision makers are turning to experimental design in order to conduct experiments within these complicated and complex simulation models in a timely matter. Data farming enables researchers to conduct many more experiments, such as simulated battles, in much less time (Kleijnen et al., 2005). It also allows them to vary a large number of inputs, such as force sizes, weapon capabilities, environmental conditions, etc., in efficient ways. This enables researchers to determine the relationships among multiple outputs (probability of mission success, length of battle, etc.) in relation to the inputs that were varied (Lucas, 2017). A good design needs to be space-filling, that is, the design points should be scattered throughout the experimental region (Cioppa & Lucas, 2007). On the SEED Center website (see http://harvest.nps.edu), there exists a catalogue of good space-filling and nearly orthogonal Latin-hypercube (NOLH) designs that allow you to alter up to 29 factors and output a NOLH design in the units of the problem. For researchers requiring more than 29 factors, they can use the mixed-integer programing approach of Hernandez et al. (2012) to generate their NOLH. Furthermore, MacCalman, Vieira, & Lucas (2017) details a method for generating Latin hypercubes that are nearly orthogonal for complete second order models. For this research, a NOLH design was used to allow for the experimental change of seven factors. If the reader would like more information regarding the efficient use of design of experiments, the SEED Center website (see http://harvest.nps.edu) contains many scholarly articles, papers, and projects with many practical applications in a variety of fields.

B. VARIABLE SELECTION

The selection of the variables for this study was relatively easy. Given the recent nature of the upgrades, the STORM developers have only included a few of the energy upgrades into the PUNIC 21 scenario. The energy-enhancements within the PUNIC 21 scenario are illustrated in the ground portion of the scenario, meaning, that only the ground units are affected by the available fuel, or lack thereof. These energy upgrades are located in the logsystem.dat file within the STORM-E framework.

The logsystem.dat file contains thousands of lines of code and hundreds of variables to specify for the simulation runs. Figure 8 shows a small portion of the logsystem.dat input. Since the ground units for both sides are the only attributes of the PUNIC 21 scenario that are affected by the fuel they receive, we decided to see how the amount of fuel they receive affects their ability to reach and/or accomplish their mission objectives. To do this, we decided to experiment with the supply vehicles responsible for providing the ground units with the fuel they demand.

```
    Define the attributes and initial inventory for each
    logistics system in the theater.

**********************************************************************

Begin Logsystem_File

Icon_default: "Supply Truck"

Begin System_Resource_Manager_List

    ID: "General Blue Theater Resource System" {
        Side:                   "Allied Coalition"
        Log_Planning_Cycle_Execution_Duration:  12   //hours
        First_Log_Planning_Cycle_Time:      1.0  //simtime
        Is_Expeditionary?               NO

        Begin Transportation_Mode_List
        Transportation_Mode:        ROAD
        Init_Number_Of_Vehicles:    15000
        Convoy_Speed:           64  //km per hr
        Max_Vehicles_Per_Convoy:    #GeneralBlue_VehiclesRoad_SUB#
        Vehicle_Capacity:       #CapacityRoad_SUB#  //Stons
        Target_Type:            "TRUCK"
        Rear_Area_Attack_Delay_Type:    "Road Supply Convoy Attack Delay"
        Icon:               "Supply Truck"
        Fuel_Type:          GROUND_WAR_POL
        Fuel_Consumption_Rate:      #ConsumpRoad_SUB#    //gallons per hour
        Allow_Alternate_Fuel_Source?    YES

        Transportation_Mode:        RAIL
        Init_Number_Of_Vehicles:    2500
        Convoy_Speed:           50  //km per hr
        Max_Vehicles_Per_Convoy:    #GeneralBlue_VehiclesRail_SUB#
        Vehicle_Capacity:       #CapacityRail_SUB#  //Stons
        Target_Type:            "RAILCAR"
        Rear_Area_Attack_Delay_Type:    "Railroad Attack Delay"
        Icon:               "Supply Train"
        Fuel_Type:          GROUND_WAR_POL
        Fuel_Consumption_Rate:      #ConsumpRail_SUB#     //gallons per hour
        Allow_Alternate_Fuel_Source?    YES
```

Figure 8. Logsystem.dat input file from STORM

Prior to STORM version 2.7, when the ground units requested fuel, it was provided to them immediately, regardless of which type of supply vehicle was needed to provide it. In version 2.7, the supply vehicles can now be constrained by their fuel consumption (gallons per hour), their amount of fuel capacity (in tons), and by the number of supply vehicles in each supply convoy. As you can imagine, if a supply truck or train has a very high fuel consumption rate, it may not have enough fuel to reach the ground units, depending on the distance. In addition to these constraints, the analyst can now choose to allow supply vehicles the ability to resupply at alternate fuel sources or whether they must return to the logistic source node. To clarify, these variables were chosen because they are the only ones that are linked within the PUNIC 21 scenario. Table 3 provides a description of the factors we explore and their baseline values. For the purpose of this research, only the Blue force supply vehicle constraints were experimented with in the design matrix.

Table 3. Logistics factors and their baseline values.

Factor	Description	Baseline Value
Vehicles_Road	The number of supply trucks in a convoy.	Vehicles_Road = 200 vehicles
Capacity_Road	The amount of fuel each supply truck can carry and deliver to demanding ground units.	Capacity_Road = 5 tons
Consumption_Road	The fuel consumption rate of each supply truck in a convoy.	Consumption_Road = 8.0 gallons per hour
Vehicles_Rail	The number of supply trains in a train convoy.	Vehicles_Rail = 10 vehicles
Capacity_Rail	The amount of fuel each supply train can carry and deliver to demanding ground units.	Capacity_Rail = 30 tons
Consumption_Rail	The fuel consumption rate of each supply train in a convoy.	Consumption_Rail = 3.2 gallons per hour
Allow_Alternate_Fuel_Source	Categorical variable (yes or no) that allows supply vehicles to resupply at logistic nodes other than where they started.	Allow_Alternate_Fuel_Source = Yes

The factor ranges were chosen broadly for this research to increase the chances of observing an effect on the responses. For the number of supply trucks and trains in a convoy, the DOE uses a multiplier of 0.25 to 1.5, which is then multiplied to the baseline value. For example, 0.25 is multiplied by 200 supply trucks (the baseline value) to come to 50 supply trucks allowed in a convoy for that specific experiment. The other factor ranges are as follows: ConsumptionRoad: 1–8 gallons per hour; CapacityRoad: 2–10 tons; ConsumptionRail: 1–8 gallons per hour; CapacityRail: 10–30 tons. The Allow_Alternate_Fuel_Source categorical variable is set to 'Yes' for half of the experiments and 'No' for the other half of the experiments.

1. Design Matrix

The Excel spreadsheet that was used to generate the design matrix is from the SEED Center website (see http://harvest.nps.edu). This spreadsheet allows the research to vary up to 29 factors while still achieving orthogonality or near-orthogonality. A design matrix is considered nearly orthogonal if all of the absolute values of the pairwise correlations between the columns are less than or equal to 0.05. This minimizes the effects of multi-collinearity. In general, the larger the number of factors selected for experimentation, the greater the number of design points needed to achieve near-orthogonality. This study uses the smallest design (up to seven factors) as it is a balance between the computational resources needed to run the design and the space-filling attributes desired. With seven factors, we use 34 design points, a 17 design point NOLH crossed with a two-level factor, each one representing a separate experiment, to achieve near-orthogonality (See Figure 9 below).

Design Point	VehiclesRoad	CapacityRoad	ConsumpRoad	VehiclesRail	CapacityRail	ConsumpRail	AllowAltSource
1	0.64	10	6.7	0.72	15	7.6	YES
2	0.33	4	7.1	0.95	10	3.2	YES
3	0.41	5.5	1.4	0.56	22.5	6.7	YES
4	0.48	7	3.2	1.5	21.3	1.9	YES
5	1.19	9.5	4.1	0.41	16.3	1	YES
6	1.5	4.5	3.6	1.27	11.3	6.3	YES
7	1.03	3.5	8	0.64	27.5	4.1	YES
8	0.95	9	6.3	1.42	26.3	5.4	YES
9	0.88	6	4.5	0.88	20	4.5	YES
10	1.11	2	2.3	1.03	25	1.4	YES
11	1.42	8	1.9	0.8	30	5.8	YES
12	1.34	6.5	7.6	1.19	17.5	2.3	YES
13	1.27	5	5.8	0.25	18.8	7.1	YES
14	0.56	2.5	4.9	1.34	23.8	8	YES
15	0.25	7.5	5.4	0.48	28.8	2.8	YES
16	0.72	8.5	1	1.11	12.5	4.9	YES
17	0.8	3	2.8	0.33	13.8	3.6	YES
18	0.64	10	6.7	0.72	15	7.6	NO
19	0.33	4	7.1	0.95	10	3.2	NO
20	0.41	5.5	1.4	0.56	22.5	6.7	NO
21	0.48	7	3.2	1.5	21.3	1.9	NO
22	1.19	9.5	4.1	0.41	16.3	1	NO
23	1.5	4.5	3.6	1.27	11.3	6.3	NO
24	1.03	3.5	8	0.64	27.5	4.1	NO
25	0.95	9	6.3	1.42	26.3	5.4	NO
26	0.88	6	4.5	0.88	20	4.5	NO
27	1.11	2	2.3	1.03	25	1.4	NO
28	1.42	8	1.9	0.8	30	5.8	NO
29	1.34	6.5	7.6	1.19	17.5	2.3	NO
30	1.27	5	5.8	0.25	18.8	7.1	NO
31	0.56	2.5	4.9	1.34	23.8	8	NO
32	0.25	7.5	5.4	0.48	28.8	2.8	NO
33	0.72	8.5	1	1.11	12.5	4.9	NO
34	0.8	3	2.8	0.33	13.8	3.6	NO

Figure 9. A matrix of the simulation input values generated by the NOLH design.

Prior to conducting the experiment, the analyst needs to verify that the design's characteristics are suitable. Figure 10 below is an output from JMP's multivariate function. This function displays the correlation and scatterplot matrices, which are used to verify that the NOLH excel spreadsheet worked as desired. The top half of the figure shows the correlation matrix. A correlation of zero indicates that the factors are independent and orthogonal (when centered) to each other, which is the best-case scenario. A low correlation between the factors indicates that there is a very low amount of confounding (i.e., co-linearity) occurring with the analysis results. As one can see in the correlation matrix, the strongest correlation value is 0.0067, which indicates a very small amount of co-linearity exists. Additionally, low correlation reduces the error in coefficient estimates in regression models, and isolates the impact of factors associated with the coefficient. The lower half of Figure 10 is the scatterplot matrix. The scatterplot matrix illustrates how well the design points sample the design space. It does this by plotting each design point onto two-dimensional subplots for each pair of factors. The analyst can then see in the illustration how space-filling this design appears to be. As one can see in the figure, the design points seem to cover the design space as intended. Therefore, the NOLH design functions as desired.

Multivariate

Correlations

	VehiclesRoad	CapacityRoad	ConsumpRoad	VehiclesRail	CapacityRail	ConsumpRail
VehiclesRoad	1.0000	-0.0013	0.0009	0.0000	-0.0025	-0.0028
CapacityRoad	-0.0013	1.0000	0.0067	0.0019	0.0000	0.0028
ConsumpRoad	0.0009	0.0067	1.0000	-0.0029	0.0017	-0.0000
VehiclesRail	0.0000	0.0019	-0.0029	1.0000	-0.0025	0.0010
CapacityRail	-0.0025	0.0000	0.0017	-0.0025	1.0000	0.0017
ConsumpRail	-0.0028	0.0028	-0.0000	0.0010	0.0017	1.0000

Scatterplot Matrix

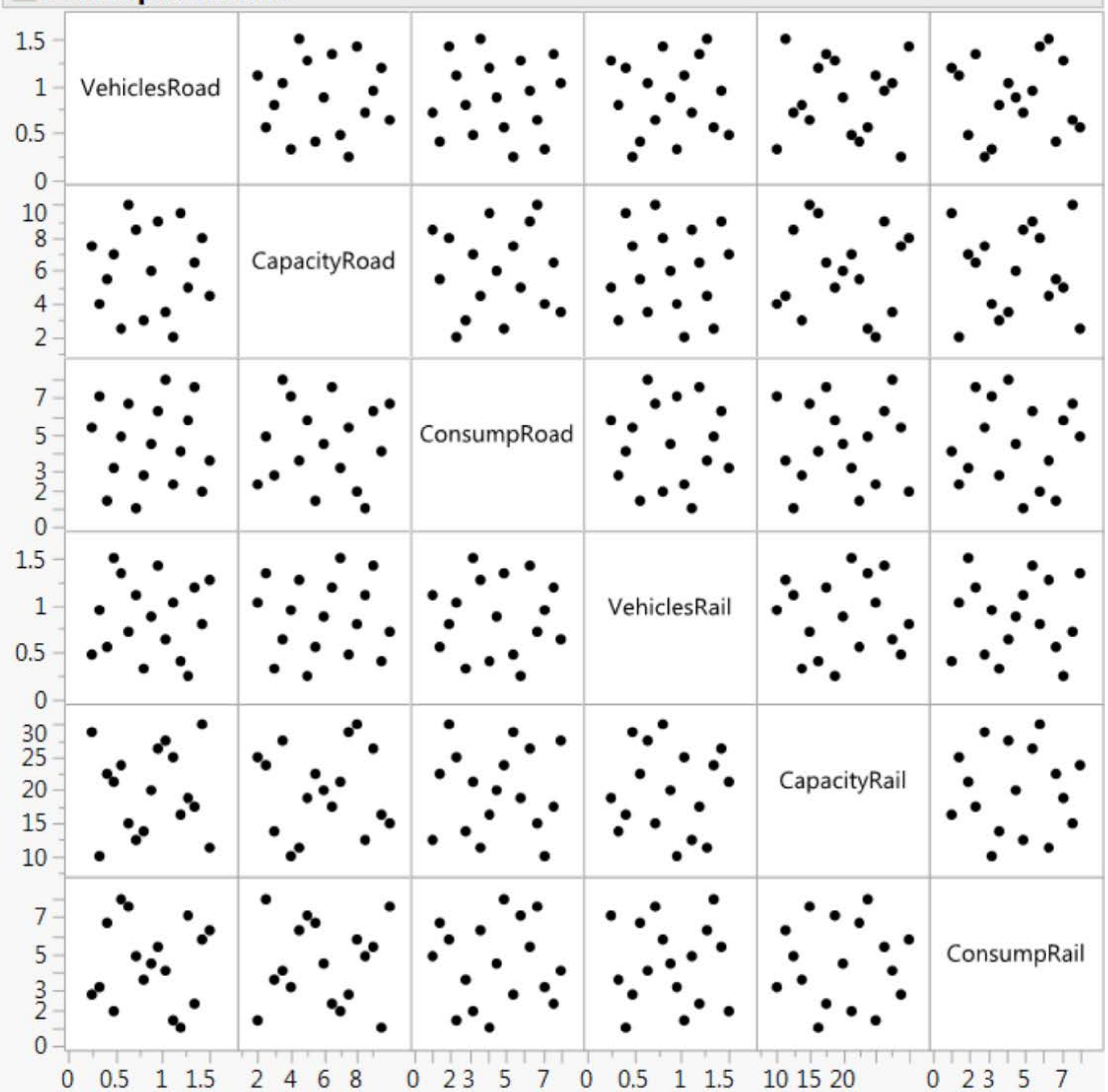


Figure 10. Correlation and scatterplot matrices of the design matrix.

2. Implementing Data Farming in STORM

This section provides the reader with a general overview of implementing data-farming in the STORM framework. The overview below, i.e. the rest of this section, was provided by Stephen C. Upton, SEED Center for Data Farming (Upton, 2018). See the appendix for step-by-step instructions, R code and shell scripts we used to facilitate the data-farming process with STORM (Upton, 2018).

In brief, the data farming process using STORM is as follows:

1. Develop a base case scenario in STORM

2. Create your design of experiments. This includes identifying the factors of interest, and their range, or set (if categorical), of values

3. Create/edit one or more STORM templates. These are the STORM input files with inserted text to mark which input variables (factors) to change when STORM builds the Excursions

4. Build the Excursions using STORM's Experiment Manager (STORM calls these "Cases")

5. Run the individual Cases as separate STORM "studies"

6. Post-process and analyze the output

7. Repeat as desired

Figure 11 gives a graphical depiction of the process. With a base case, a Run Matrix (your design of experiments), and the STORM templates, the STORM Experiment Manager is then used to build the Excursions. We then run those on one or more machines using the R doParallel package to execute one or more runs concurrently and manage the tasks. We then post-processed the output using the STORMMiner tool and other R scripts to put the output in a form that can be analyzed. The analyst then used JMP Pro 13 to perform the analyses, using regressions and partition trees among other methods, and to create a variety of plots and visualizations. Note that the use of JMP is not required; any statistical analysis software package could be used.

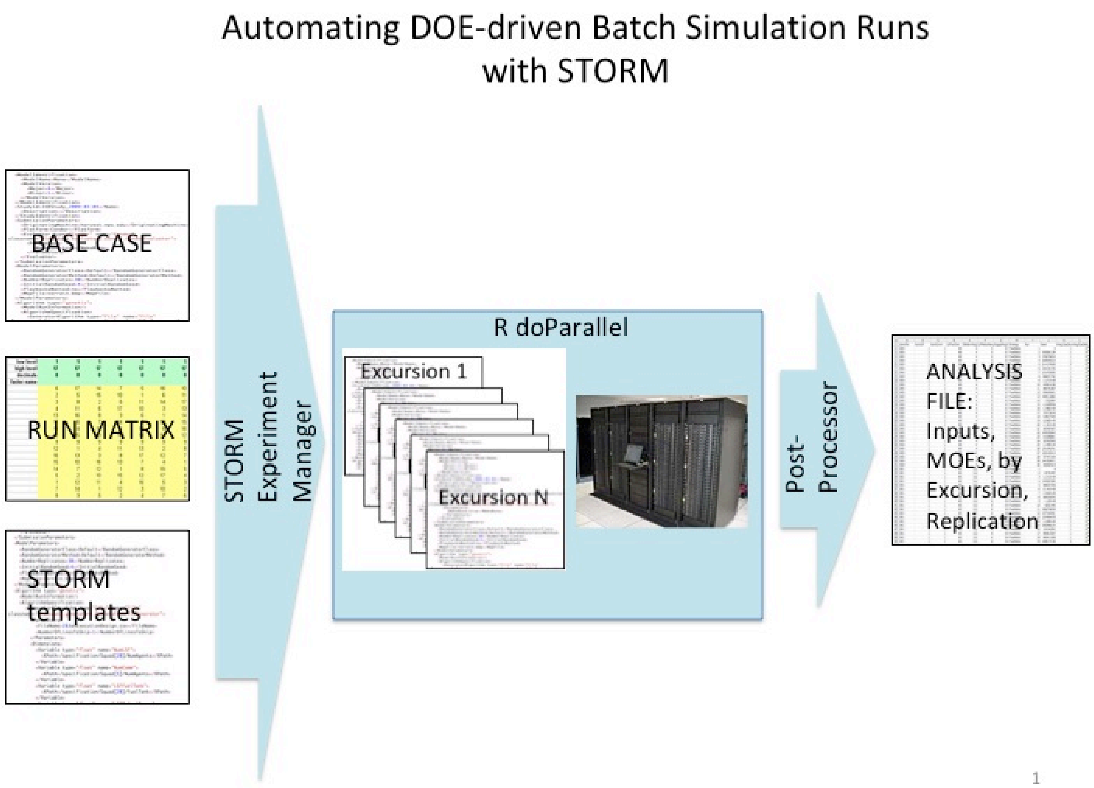


Figure 11. Graphical depiction of the data farming process in STORM. Source: Upton (2018).

IV. ANALYSIS

This section provides the reader with post-processing analysis of the results based on three chosen metrics of interest. The chosen metrics are not readily available from one of STORM's output tools. Therefore, custom SQL query statements were required in order to extract the data we needed from the database. If the reader is interested in the SQL scripts used to extract the data, contact the NPS SEED Center (see http://harvest.nps.edu). For this research, the analyst used the statistical software program JMP Pro13 to conduct the analysis, however, any statistical software program would have sufficed.

The primary statistical methods used to analyze our metrics are stepwise regression and partition trees. These allow the analyst to see the variability in the output and to determine which factors had the most impact on the various metrics chosen. Understanding the casual relationships between the factors and their interactions can lead the analyst to robust solutions, if they exist (Kleijnen et al., 2005). As noted by Kleijnen et al. (2005), robust solutions are often more desirable than optimal solutions when there are multiple sources of uncertainty. In addition to regression and partition trees, carefully chosen graphs and plots were also used to support the various observations.

A. AVERAGE INVENTORY LEVELS

When observing the average inventory levels across all resources, we wanted to see how allowing the supply vehicles to resupply at alternate sources, rather than their source nodes, would affect the various resource inventory levels during the campaign. When the factor Allow_Alt_Fuel_Source is no, supply vehicles must return to their originating node to resupply before venturing back out to meet demand. However, when Allow_Alt_Fuel_Source is yes, as the Blue ground forces move forward they can set up forward operating logistic nodes so that the supply vehicles can resupply at those forward nodes rather than returning to their source nodes to resupply. Figure 12 is a representation of the average inventory level for each resource and for each day during the campaign. The resource levels we are exploring are Ammo, Army POL (petroleum, oil, and lubricants), Dry Bulk, and Water. This data was collected by taking the average inventory level of all

50 replications for each design point and then separating out each inventory level by day of the campaign. We also wanted to explore how allowing alternate sources of fuel would affect the resource levels. We then plotted those average resource levels for each day of the campaign and split on allow alternate sources of fuel data, yes or no, see Figure 12. The first item that jumps out is the amount of variability in the Army POL (green line) average resource level. The other resources have a small amount of variability, but not close to the extent that Army POL does. This indicates that as we move further into the campaign, the amount of fuel the ground units are using (Army POL) has much more variation than the other resources. This makes sense because the amount of fuel required by a ground unit will change depending on the amount of combat they see. As we move further into the campaign, the Blue forces are venturing deep into Red occupied territory. Depending on the route and the unit, some units will see more combat and require more fuel than other units, leading to variability in the Army POL resource levels. Knowing that Army POL is more variable could be useful when planning safety stocks.

We also see in Figure 12 that there does not appear to be much of a difference in the inventory levels when we allow alternate sources of fuel. This was surprising because one would think that increasing the complexity of the logistics network would have an effect on the inventory resource levels. However, we do not see that in Figure 12.

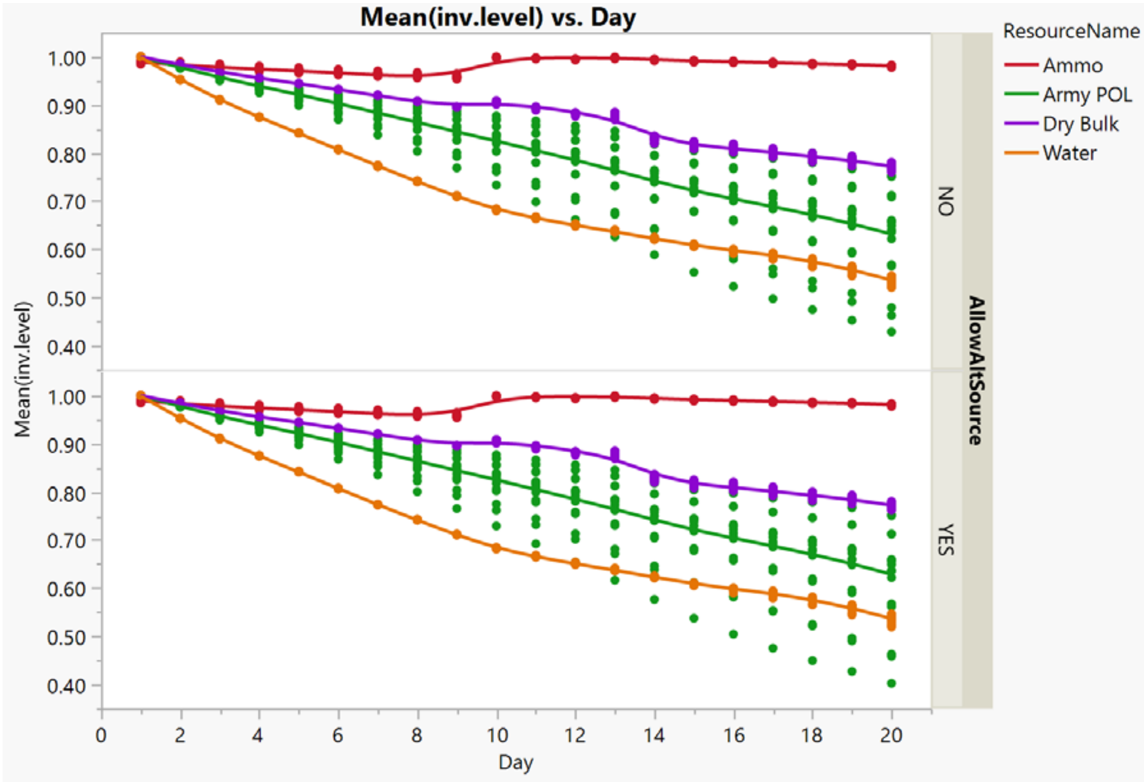


Figure 12. Average inventory level of each resource for each day of the campaign.

The histogram in Figure 13 helps us see the variability in the average Army POL inventory level throughout the campaign. We can see that the average Army POL inventory level varies from 70% to 90% across the 34 design points. Furthermore, by selecting the left and right most columns we can see the factors that cause these extreme values. It appears that when the supply road vehicles (supply trucks) have a high consumption rate and a low capacity the inventory level of Army POL is roughly 70%. On the other hand, if the supply trucks have a low consumption rate and a high capacity, the Army POL inventory level reaches 90%. This makes sense because the supply vehicles would be able to meet more demand if they can stay on the road longer (low consumption rate) and carry more supplies (high capacity).

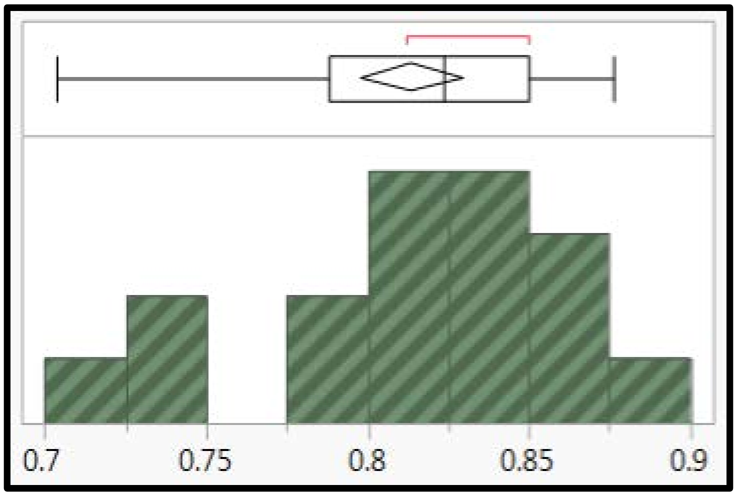


Figure 13. Histogram of average Army POL inventory level.

Next, we did a stepwise regression to see how well the factors we varied predicted the average inventory level of the Army POL, see Figure 14. As one can see, three of the terms (the factors ConsumpRoad and CapacityRoad, as well as their interaction) have a significant effect on the predicted inventory level of Army POL. A high RSquare value (0.96) indicates that our predictor variables fit the model extremely well. That is, 96% of the variance in the response is explained by the predictor variables. Additionally, the p-values of three factors (ConsumpRoad, CapacityRoad, and their interaction) is less than 0.001, which means they are statistically significant to the model. In the top part of Figure 14, we see the factors that have the most impact on the Army POL inventory level are the fuel consumption rate and capacity of the supply trucks, as well as their interaction. This is based on the significance of their p-values (less than 0.05)

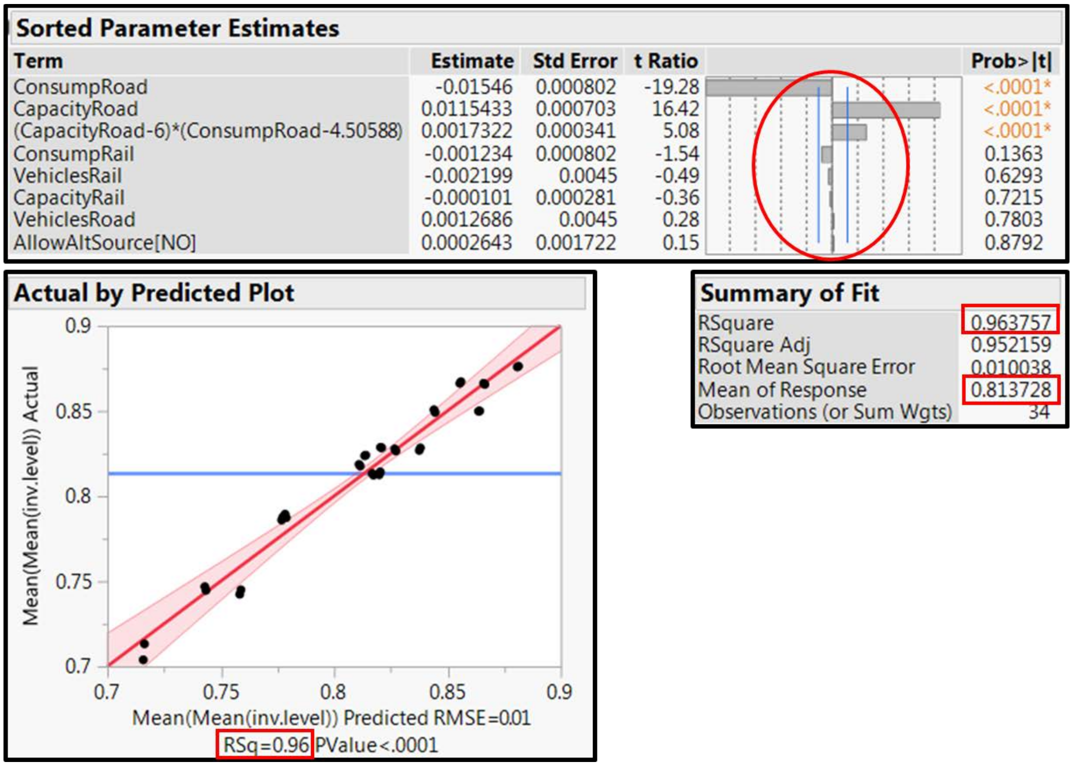


Figure 14. Regression with average Army POL inventory level as response variable.

Next, we created a partition tree to see which factors have the most impact on the Army POL inventory level, see Figure 15. To get this data we took the average of the Army POL inventory levels across the 50 replications for each of the 34 design points. As one can see, the average Army POL inventory level during the campaign is roughly 81%, (see leaf 1 in Figure 15). Next, we split the tree to determine which factors most affect the Army POL level. One can see that when the capacity of the supply trucks (CapacityRoad) is greater than 5.5 tons (leaf 2), the average inventory level improves to 84%, however, when CapacityRoad is less than 5.5 tons (leaf 3), the average inventory level decreases to 77%. This indicates that the more fuel the supply trucks carry leads to higher inventory levels, as one would expect. Furthermore, we see that the consumption rate of the supply road vehicles (ConsumpRoad) factor is the next most significant to the Army POL inventory level. With a higher capacity (greater than 5.5 tons) and a lower consumption rate (less than 4.5 gallons per hour), the Army POL inventory numbers improve to 86% (leaf 4). However, with a consumption rate greater than 4.5 gallons per hour, the Army POL inventory level decreases to 81%. We see the same effects on the other side of the tree when CapacityRoad is less than 5.5 tons (leaf 5). As one can see, the highest average inventory level for Army POL occurs when the supply trucks have a higher capacity and lower consumption rate.

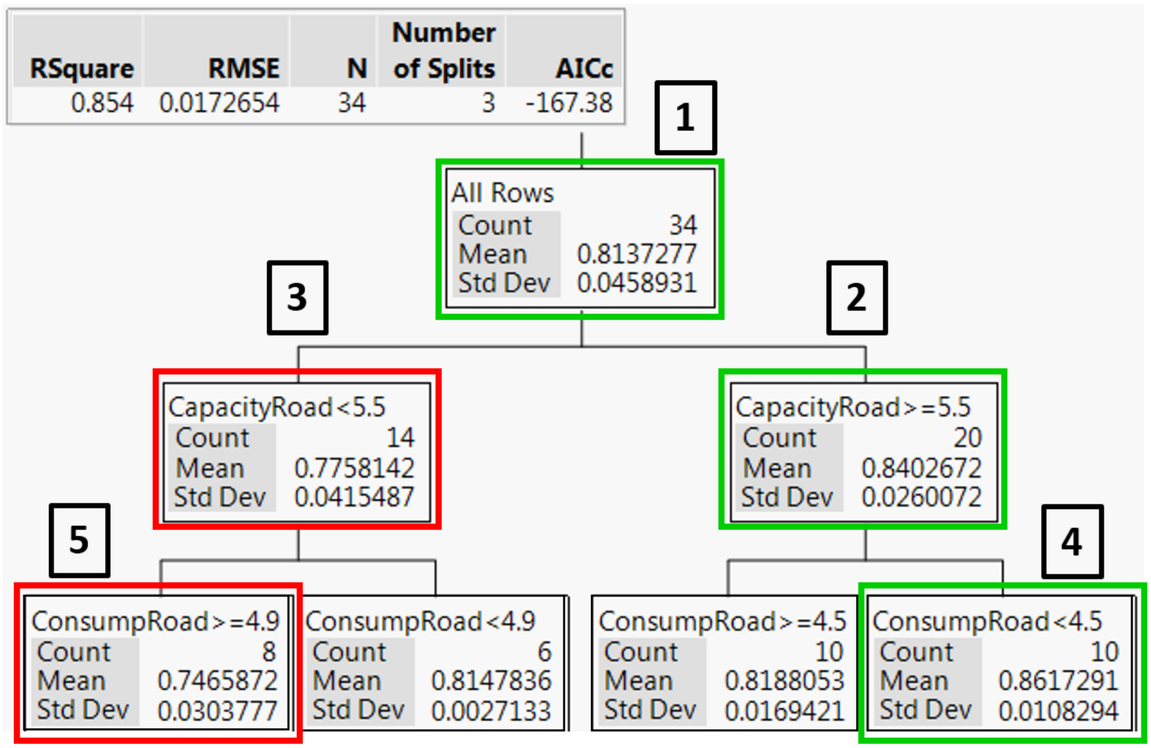


Figure 15. Average Army POL inventory level during the campaign.

B. FUEL CONSUMED

In analyzing the fuel consumed during the ground campaign, we first wanted to see how the supply vehicles, both road and rail, were able to meet the fuel demands imposed on them. The fuel-consumed data comes from the amount of fuel consumed by the supply vehicles in order to meet the supply demanded of them. As you can imagine, logistics networks are typically long and complex. If supply vehicles have smaller capacities, for example, they will need to make more trips in order to meet the demand requested from them. This analysis is designed to show how the capacity and consumption rates of the supply vehicles affects how much fuel they consume in order to meet the demand. In order to get this data, we took the average of the 50 replications for each of the 34 design points to get to the average amount of fuel consumed by each type of supply vehicles, road and rail, per design point.

1. Supply Road Vehicles (Supply Trucks)

Once we extracted the data, we created a histogram to see the variability in our results, see Figure 16. One can see that the average amount of fuel consumed varies by a large amount, roughly 5,000,000 to 30,000,000 gallons. By selecting the left and right most columns we were able to determine that the supply trucks used the most fuel (almost 30,000,000 gallons) when they had a low capacity and a high fuel consumption rate. This makes sense because if they carry less they have to resupply more in order to meet demand, which uses more fuel.

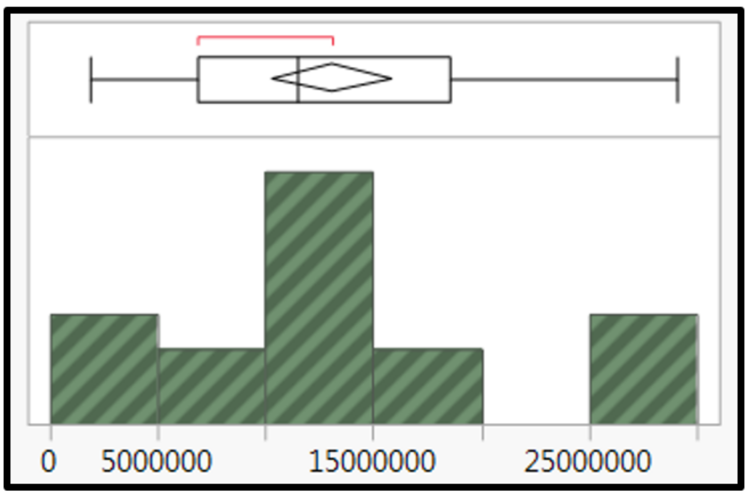


Figure 16. Histogram of the average fuel consumed by supply trucks.

Next, we analyzed the fuel consumed by the supply trucks, or road vehicles, see the partition tree in Figure 17. As one can see, the average fuel consumed for the supply trucks is 13,113,193 gallons for the entirety of the campaign (leaf 1). However, there are two factors that significantly affect the fuel consumed; the capacity the supply trucks can carry and their fuel consumption rates. As we can see in Figure 17, when the supply truck capacity (CapacityRoad) is less than 5.5 tons (leaf 2), the supply trucks consume

19,608,660 gallons of fuel on average; whereas, when the capacity is greater than 5.5 tons (leaf 3), they only consume 8,566,365 gallons of fuel on average. This result makes sense because when the supply truck capacity is less than 5.5 tons they need to make more trips in order to meet the demand, thus using more fuel. However, when they can carry more capacity, they do not need the additional trips, and therefore, use much less fuel.

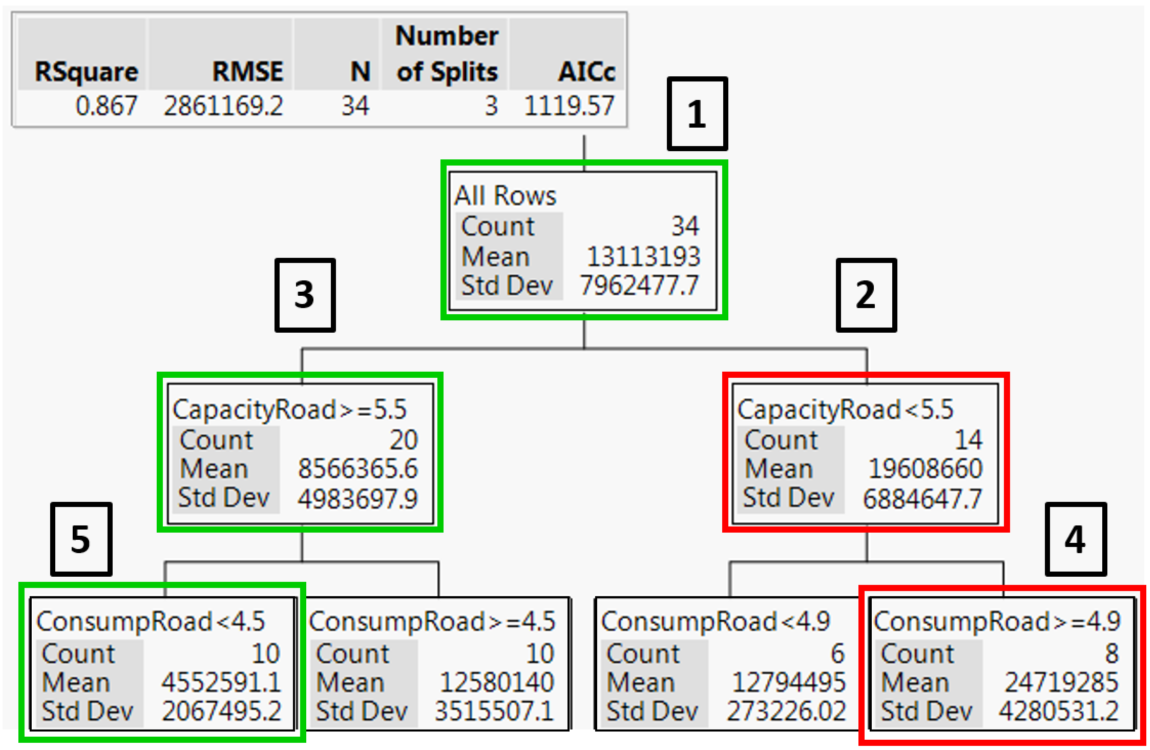


Figure 17. Fuel consumed by supply road vehicles

We can also see in Figure 17 that after CapacityRoad, the most significant factor is the consumption rate (ConsumpRoad) of the supply trucks. We can see on both sides of the partition tree that a higher consumption rate causes the supply trucks to use more fuel, which makes sense. Specifically, when the supply truck capacity is less than 5.5 tons, and the consumption rate is greater than 4.9 gallons per hour, the supply trucks use almost 25,000,000 gallons of fuel (leaf 4). However, when the consumption rate is less than 4.9 gallons per hour, the supply trucks use roughly half that number (12.8 million gallons). We see the same effect on the other side of the tree when the capacity is greater than 5.5 tons (leaf 5). This shows that by having a larger fuel consumption rate the supply trucks require more fuel to meet demand than if their consumption rate is lower.

2. Supply Rail Vehicles (Supply Trains)

After extracting the fuel consumption data of the supply trains, we created a histogram to see the variability in the results, see Figure 18. As with the supply trucks, we see lots of variability in our results. The fuel consumed ranges from less than 10,000 gallons to almost 70,000 gallons. In selecting the left and right most columns, we see that when the supply trains have a high capacity (greater than 16.3 tons) and low consumption rate (1 gallon per hour), they burn less than 10,000 gallons of fuel. This makes sense because they do not have to resupply as often and therefore burn less fuel. Naturally, we see the opposite on the other side of the graph. When the supply trains have a low capacity and high consumption rate, they burn more fuel.

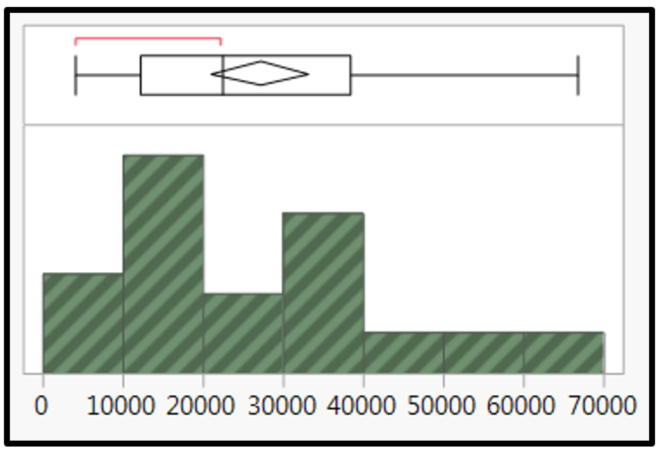


Figure 18. Histogram of the average fuel consumed by supply trains.

Next, we created a partition tree to look at the fuel consumed by the supply trains, or rail vehicles, to determine which factors are the most significant to the amount of fuel they consumed, see Figure 19. First, we noticed that the supply trains use much less fuel than the supply trucks. The average fuel consumed throughout the campaign for the supply trains is only 27,241 gallons (leaf 1), vice 13 million with the supply trucks. We believe this is because the supply trains are tied to the train tracks, whereas the supply trucks have more freedom of movement and are better able to follow and respond to the Blue forces' demand. Similar to the road network, the most significant factors are the consumption rate and the capacity of the supply trains. Unlike the supply trucks, the most significant factor impacting the fuel consumed by supply trains is their fuel consumption rate, not their capacity. We see that when their consumption rate is greater than 6.3 gallons per hour, they use just under 48,000 gallons of fuel (leaf 2), however, when their consumption rate is lower than 6.3 gallons, they only use 18,715 gallons of fuel on average (leaf 3). That is a difference of 30,000 gallons.

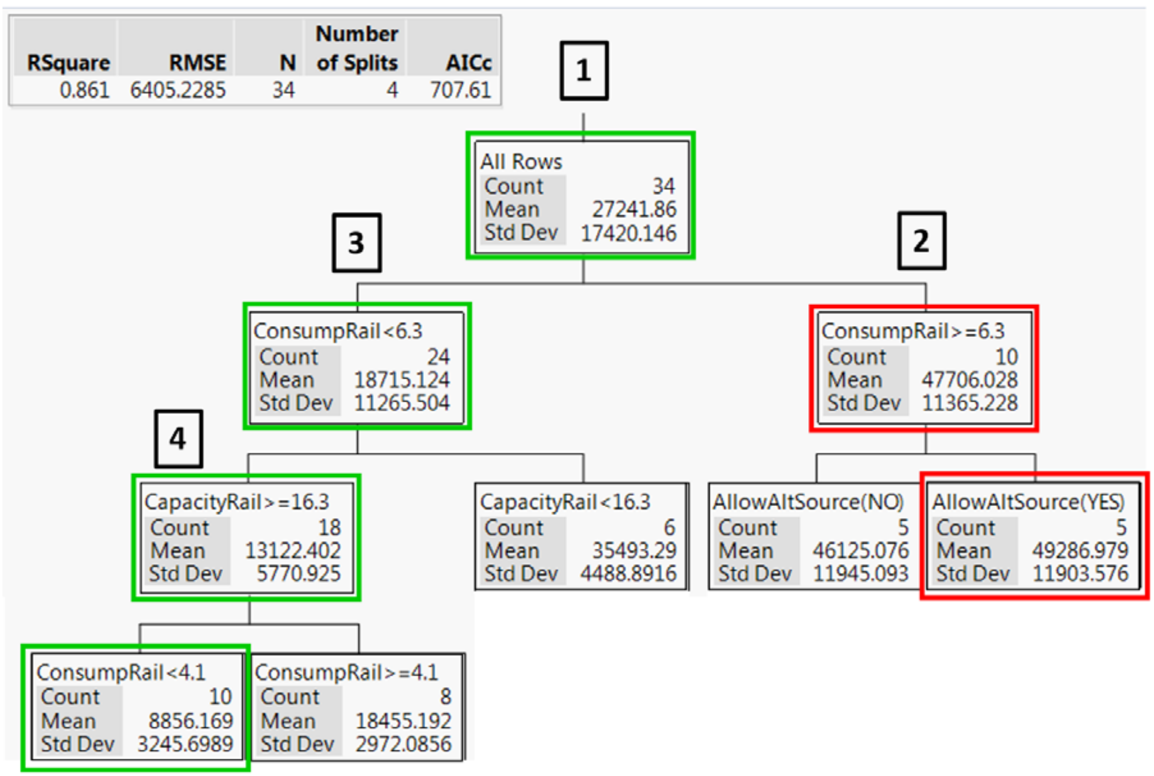


Figure 19. Fuel consumed by supply rail vehicles.

Exploring further in Figure 19, we can see that when the consumption rate is lower than 6.3 gallons per hour, the factor with the most significance is the capacity of the supply trains. We see that when the capacity is less than 16.3 tons, the average fuel consumed is roughly 35,000 gallons, whereas, when the capacity is greater than 16.3, the fuel consumed is 13,100 gallons (leaf 4). This is result is similar to that of the supply trucks. Since the trains are carrying less capacity, they have to return to resupply more frequently to meet the demand, thus using more fuel. However, when they can carry more, they do not need to resupply as often and therefore use much less fuel. As we have seen in these findings, the capacity and consumption rate of the supply vehicles has a significant impact on how much fuel they need to consume in order to meet the demand of the Blue forces.

C. PROBABILITY RED LOSES ZARAGOZA

Zaragoza is a city deep in Red territory and is the last military objective of the Blue ground units. In analyzing the probability that Zaragoza is lost by the Red forces, we first created a histogram to see the variability in the results, see Figure 20. As one can see, there is a lot of variability in our results, roughly 40% to 75%. First, we wanted to see what caused our lowest and highest results (left and right most columns). In selecting the left most column, we see that the probability that Zaragoza is lost is 44% when the number of supply road vehicles is less than 128 per convoy and while their capacity is less than four tons. On the other hand, Zaragoza is lost 74% of the time when there are greater than 128 supply road vehicles along with a low fuel consumption rate.

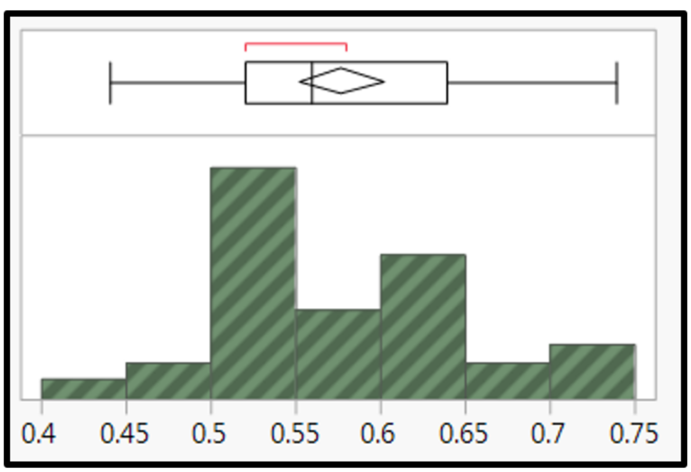
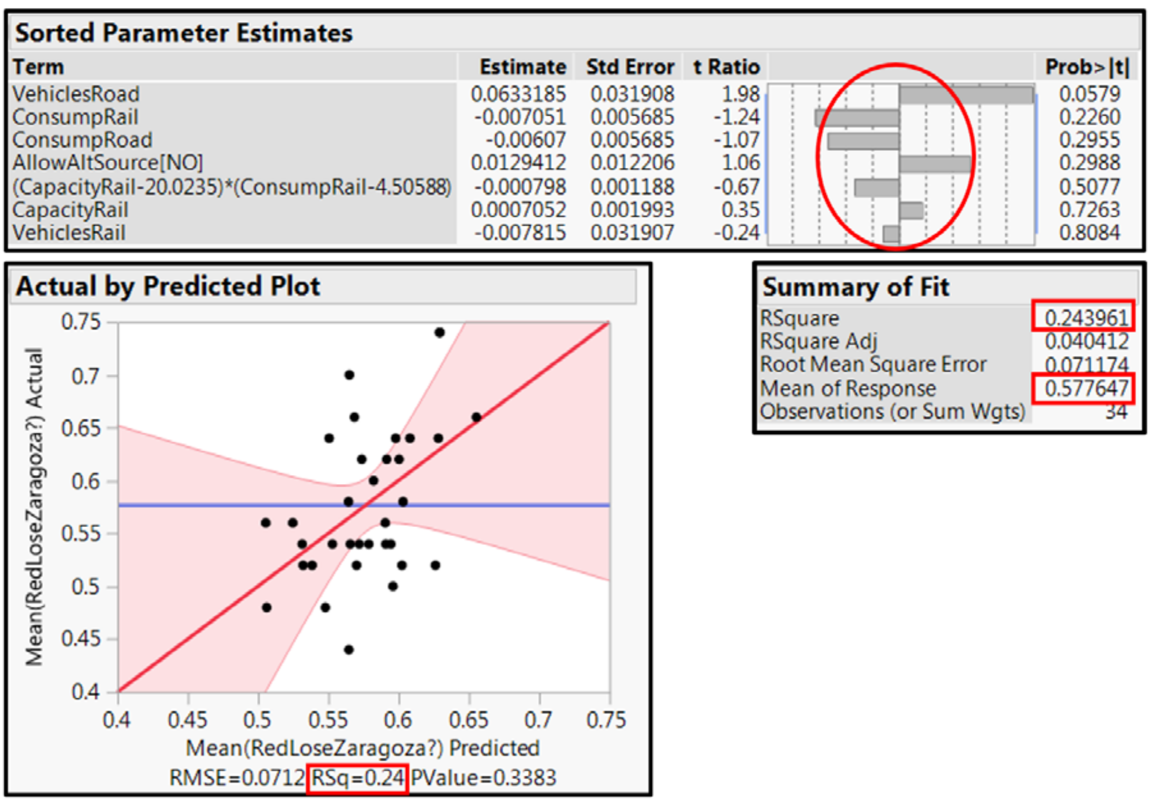


Figure 20. Histogram of the probability that Zaragoza is lost by Red forces.

Next, we created a regression to see how well our energy factors predict the probability that Red loses Zaragoza. In order to conduct this analysis, we took the average of the 50 replications over each of the 34 design points. We then fit a regression model with the probability that Zaragoza is lost as the response variable and the factors we varied as the predictor variables. This allows us to see how the probability Zaragoza is lost varies due to the level each of the predictor variables are set. In Figure 21, you can see that there is a lot of variability in our response variable. The probability that Zaragoza is lost goes from 44% to 74%. A very low RSquare value (0.24) indicates that our model does not explain a lot of the deviation we see. However, insights can still be gained. Even with a low RSquare value, it appears that the logistic factors we varied still have an effect on whether or not Blue is able to complete their final military objective of capturing Zaragoza.



Sorted Parameter Estimates

Term	Estimate	Std Error	t Ratio	Prob>|t|
VehiclesRoad	0.0633185	0.031908	1.98	0.0579
ConsumpRail	-0.007051	0.005685	-1.24	0.2260
ConsumpRoad	-0.00607	0.005685	-1.07	0.2955
AllowAltSource[NO]	0.0129412	0.012206	1.06	0.2988
(CapacityRail-20.0235)*(ConsumpRail-4.50588)	-0.000798	0.001188	-0.67	0.5077
CapacityRail	0.0007052	0.001993	0.35	0.7263
VehiclesRail	-0.007815	0.031907	-0.24	0.8084

Summary of Fit

RSquare	0.243961
RSquare Adj	0.040412
Root Mean Square Error	0.071174
Mean of Response	0.577647
Observations (or Sum Wgts)	34

Figure 21. Regression with response variable of probability Zaragoza is lost by Red forces.

We next conducted an analysis on the energy factor levels using a partition tree. As stated earlier, a partition tree allows us to see which factors affect the response variable the most. For this analysis, we took the average probability (over the 50 replications) that Zaragoza is lost by Red forces, for each of the 34 design points. In Figure 22, you can see that before the split the average probability that Zaragoza is lost is 0.578 (leaf 1). Keep in mind, splitting the partition tree allows us to see which factors are the most influential and also shows us which value (or threshold) those factors must be in order to affect the response variable (Probability Zaragoza is lost) the most.

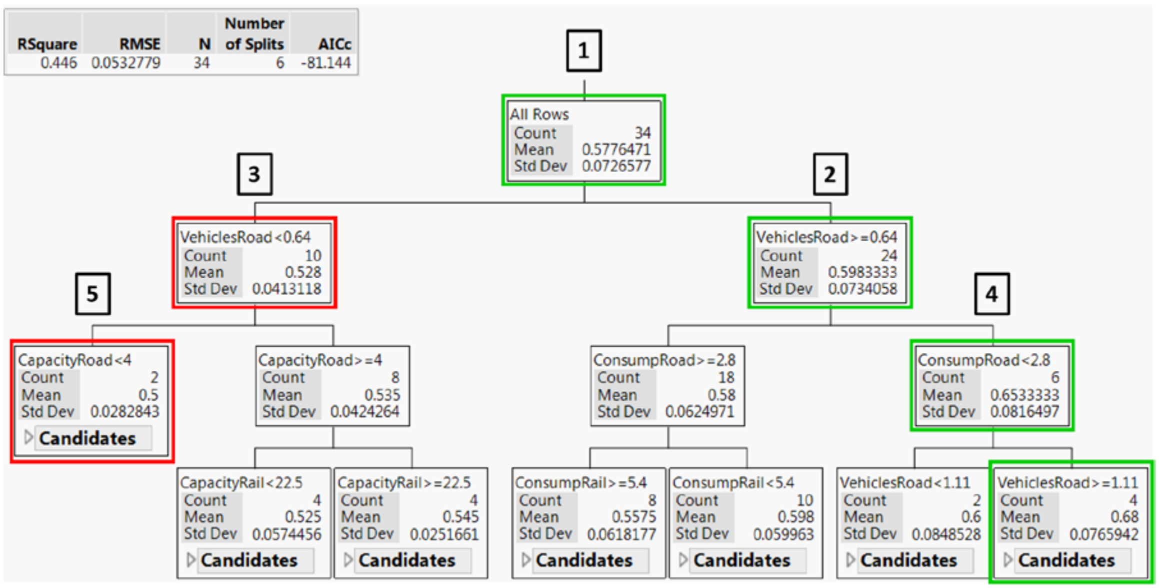


Figure 22. Partition tree illustrating influential factors that affect the probability Zaragoza is lost by Red forces.

As one can see in Figure 22, when the number of supply trucks (VehiclesRoad) is greater than 64% of 200, or 128 vehicles, the probability that Zaragoza is lost increases to roughly 0.60 (leaf 2). If you have less than 128 supply trucks, the probability decreases to 0.528 (leaf 3). Additionally, if the 128 supply trucks are more efficient, that is, if they have a consumption rate lower than 2.8 gallons per hour, the probability that Zaragoza is lost increases to 0.653 (leaf 4). This is significant because it shows decision makers that the number of supply vehicles in the supply convoy, and their consumption rates, have a significant impact on whether or not the ground units (in this scenario) can reach and achieve their objective.

THIS PAGE IS INTENTIONALLY LEFT BLANK

V. CONCLUSION

This thesis illustrates the advantages of using an efficient design of experiments to systematically change input parameters to explore the influence of energy-related variables on campaign outcomes. It is a proof-of-concept study that assists in enhancing the quality of analysis provided by STORM. Additionally, the research conducted in this thesis provides the STORM user community with a methodology that can be utilized in future studies.

A. SUMMARY OF FINDINGS

This data farming approach uses steps that are rational, observable, and offer a practical direction for analysis so that it can be easily incorporated into future studies. Utilizing an efficient designed experiment allows the analyst to vary a large number of input factors over a wider range. This allows the researcher to see whether response variables differ in means and variances over the design points and to determine which factors are the most influential on the outcome. Across all of the metrics of interest, we see that varying the logistical factors influence the amount of fuel consumed, the resource levels of the ground units, and also have a significant impact on the successful capture of Zaragoza, a campaign-level objective. This study illustrates to decision makers how employing a simulation model that incorporates energy and other logistical considerations can impact planning and executing future campaigns or designing future forces. Efforts to improve or increase STORM energy features are proving to be a worthwhile investment.

B. RECOMMENDATIONS FOR FUTURE RESEARCH

This research outlines a methodology to conduct sensitivity analysis on the energy enhancements in STORM. This methodology can be expanded to include other input factors, specifically the other energy-related enhancements, to explore outcome metrics of particular interest to the analyst. Further research can also be conducted using the methodology in this research on a real-world, classified scenario. Furthermore, the data farming approach may be useful both during and after the development of new STORM scenarios by focusing the analyst's attention on influential energy-related factors to reveal the campaign's sensitivity.

The following is a summary of our accomplishments:

- We used an advanced DOE to efficiently explore multiple energy factors with STORMFarmer in 1,700 simulated campaigns.

- This research proves the ability to efficiently explore new STORM energy features. By using an efficient design and a computing cluster, it took 121 hours to simulate the 1700 campaigns. Using a brute force full factorial design and a single processor would have required nearly 12,000 years. STORM analysts can use this new capability to provide faster and more comprehensive studies.

- New energy features have been shown to impact both logistic and mission-oriented measures in intuitive ways. This provides enhanced confidence in STORM 2.7's verification.

- In Punic 21, changes in logistic factors lead to an increase in variability across all of our metrics of interest.

Next steps: The new STORM energy features are ready for more extensive testing on an existing real-world classified scenario.

C. CLOSING THOUGHTS

Military decision makers within the Department of Defense (DOD) must make investment and strategy decisions regarding force structure and weapon system capabilities for the forces of tomorrow. The problem, however, is that these decisions must be made with a limited budget and limited information regarding the future strategic environment. Campaign models, such as STORM, enable decision makers to quickly and inexpensively change key features in a simulated campaign-level, joint military operation to determine the effect those changes would have on the campaign outcome. This thesis demonstrates how designed experiments may assist in the process of making investment and strategy decisions that ensure tomorrow's military is ready for future challenges.

LIST OF REFERENCES

Bickel, W.G. (2014). Improving the analysis capabilities of the Synthetic Theater Operations Research Model (STORM). Master's thesis, Naval Postgraduate School, Monterey, CA.

Cobbs, D.G. (2018). Implementing a design of experiments within the Synthetic Theater Operations Research Model (STORM). Master's thesis, Naval Postgraduate School, Monterey, CA. This document is classified SECRET//NORFORN.

Cioppa, T.M., & Lucas, T.W. (2007). Efficient nearly orthogonal and space-filling Latin hypercubes. *Technometrics, 49*(1), 45–55.

Group W. (2012). *Punic 21: A New History Overview*. Fairfax, VA: Author.

Group W. (2017a). *STORM: User's manual version 2.7*. Fairfax, VA: Author.

Group W. (2017b). *STORM: Analyst's manual version 2.7*. Fairfax, VA: Author.

Group W. (2017c). *STORM: Programmer's manual version 2.7*. Fairfax, VA: Author.

Group W. (2017d). *What's new in STORM version 2.7*. Fairfax, VA: Author.

Hernandez, A.S., Lucas, T.W., & Carlyle, M. (2012). Constructing nearly orthogonal Latin hypercubes for any nonsaturated run-variable combination. *ACM Transactions on Modeling and Computer Simulation*, *22*(4), 1-17.

Kleijnen, J.P., Sanchez, S.M., Lucas, T.W., & Cioppa, T.M. (2005). A user's guide to the brave new world of designing simulation experiments. *INFORMS Journal on Computing*, *17*(3), 263–289.

Kline, J., Hughes, W., & Otte, D. (2011). "Campaign analysis: An introductory review." In *Wiley Encyclopedia of Operations Research and Management Science*. New York: John Wiley & Sons.

Lucas, T. W. (2017). *Getting the most from your model through efficient design of experiments.* Unpublished slides, Department of Operations Research, Naval Postgraduate School, Monterey, CA.

MacCalman, A.D., Vieira, H., & Lucas, T.W. (2017). Second order nearly orthogonal Latin hypercubes for exploring complex stochastic simulations. *Journal of Simulation*, 11(2), 137-150.

McDonald, M.L., Upton, S.C., Seymour, C.N., Lucas, T.W., Sanchez, P.J., Sanchez, S.M., Schramm, H.C., & Smith, J.R. (2014). Enhancing the analytic utility of the Synthetic Theater Operations Research Model (STORM). *Proceedings of the 2014 Winter Simulation Conference,* 4136–4137.

Morgan, B.L., Schramm, H.C., Smith, J.R., Lucas, T.W., McDonald, M.L., Sanchez, P.J., Sanchez, S.M., & Upton, S.C. (2017). Improving U.S. Navy campaign analyses with big data. *Interfaces,* 48(2), 130–146.

Sanchez, S.M., Lucas, T.W., Sanchez, P.J., Nannini, C.J., & Wan, H. (2012). Designs for large-scale simulation experiments, with applications to defense and homeland security. In K. Hinkelmann (Ed.), *Design and Analysis of Experiments* (Vol. 1, p.600). Hoboken, NJ: John Wiley & Sons.

SEED Center. (2018). *STORMMiner User's Manual version 1.2.0.* Naval Postgraduate School, Monterey, CA.

Seymour, C.N. (2014). Capturing the full potential of the Synthetic Theater Operations Research Model (STORM). Master's thesis, Naval Postgraduate School, Monterey, CA.

Upton, S. C. (2018). *How to data farm with STORM.* Unpublished technical report, SEED Center for Data Farming, Naval Postgraduate School, Monterey, California.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California