

Dakota State University

Beadle Scholar

Masters Theses & Doctoral Dissertations

2-2023

DEFINING A CYBER OPERATIONS PERFORMANCE FRAMEWORK VIA COMPUTATIONAL MODELING

Briant Becote

Follow this and additional works at: <https://scholar.dsu.edu/theses>



DEFINING A CYBER OPERATIONS PERFORMANCE FRAMEWORK VIA COMPUTATIONAL MODELING

A dissertation submitted to Dakota State University in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

in

Cyber Operations

February 24th, 2023

By

Briant Becote

Dissertation Committee:

Dr. Bhaskar Rimal

Dr. Ronghua Shan

Dr. Yong Wang

Beacom College of Computer and Cyber Sciences



DISSERTATION APPROVAL FORM

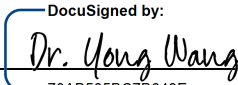
This dissertation is approved as a credible and independent investigation by a candidate for the Doctor of Philosophy degree and is acceptable for meeting the dissertation requirements for this degree. Acceptance of this dissertation does not imply that the conclusions reached by the candidate are necessarily the conclusions of the major department or university.

Student Name: Briant Becote

Dissertation Title: Defining a Cyber Operations Performance Framework Via Computational Modeling

Dissertation Chair/Co-Chair:  Date: April 20, 2023
Name: Bhaskar Rimal DocuSigned by: 05A737B4A4D5481...

Dissertation Chair/Co-Chair: _____ Date: _____
Name: _____

Committee member:  Date: April 21, 2023
Name: Dr. Yong Wang DocuSigned by: 70AB505BC7B649E...

Committee member:  Date: April 21, 2023
Name: Ronghua Shan DocuSigned by: 6E11517A148D4B4...

Committee member: _____ Date: _____
Name: _____

Committee member: _____ Date: _____
Name: _____

ACKNOWLEDGMENTS

My research herein represents a labor of love and learning that began long before I was accepted into the Ph.D. Cyber Operations program at Dakota State University. Its origins are in the organizational psychology courses that define me as a learner, researcher, professional leader, and college professor. This foundation directly characterizes my perspective as a scientist and is the driving factor in my approach to learning, self and organizational improvement, research, and teaching.

To reach this pivotal achievement, I'd like to thank the following:

- Past professors, whose seeds of education produced a curiosity, drive, and dedication to learning and growth demonstrated through countless hours of learning and teaching.
- Student colleagues and friends, who provided the necessary support and community to keep me moving forward when the goal line seemed beyond sight.
- My committee: Dr. Bhaskar Rimal, Dr. Rohgua Shan, and Dr. Yong Wang. Their time spent in feedback, guidance, support, and freedom to explore enabled growth, humility, and some tireless nights, but a better dissertation and researcher for their efforts.
- My Employer, the US Navy. Through a variety of programs, the Navy has enabled me to continuously learn, grow and advance as a student and a professional.
- My wife, Erin, and daughters Brooklyn in Cadence. Your patience and support over many years was everything in making this achievement possible and encouraging me to be the best me I can be.

ABSTRACT

Cyber operations are influenced by a wide range of environmental characteristics, strategic policies, organizational procedures, complex networks, and the individuals who attack and defend these cyber battlegrounds. While no two cyber operations are identical, leveraging the power of computational modeling will enable decision-makers to understand and evaluate the effect of these influences prior to their impact on mission success. Given the complexity of these influences, this research proposes an agent-based modeling framework that will result in an operational performance dashboard for user analysis. To account for cyber team behavioral characteristics, this research includes the development and validation of the Cyber Operations Self-Efficacy Scales (COSES). The underlying statistics, algorithms, research instruments, and equations to support the overall framework are provided. This research represents the most comprehensive cyber operations agent-based performance analysis tools published to date.

DECLARATION

I hereby certify that this dissertation constitutes my own product, that where the language of others is set forth, quotation marks so indicate, and that appropriate credit is given where I have used the language, ideas, expressions or writings of another. I declare that the dissertation describes original work that has not previously been presented for the award of any other degree of any institution.

Signed,

Briant Becote

Briant Becote

TABLE OF CONTENTS

Dissertation Approval Form	ii
Acknowledgments	iii
Abstract	iv
Declaration	v
Table of Contents	vi
List of Tables	x
List of Figures	xi
Chapter 1:	
Introduction	1
1.1 Problem Statement	2
1.2 Research Aims, Objectives, and Questions	2
1.3 Overview of the Framework	3
1.4 Motivation	4
1.5 Major Theories	5
1.5.1 Complexity Science & Computational Modeling	5
1.5.2 Self-Efficacy & Performance	6
1.5.3 Performance Management	7

1.6	Research Significance	7
Chapter 2:		
	Literature Review	9
2.1	Complexity Science Introduction	10
2.1.1	Early History	11
2.1.2	Terminology	12
2.1.3	Fields of Study	14
2.1.4	Complex Systems Tools	16
2.1.5	Cybersecurity Dynamics Framework	20
2.1.6	Agent-Based Modeling	21
2.1.7	Trends in Research	25
2.2	Self-Efficacy and Performance	27
Chapter 3:		
	Research Methodology	29
3.1	Introduction	29
3.2	Cyber Performance Metrics	30
3.2.1	COSES Methodology & Design	32
3.2.2	Friendly and Adversary Skills and Expertise	38
3.2.3	Network Characteristics	40
3.2.4	Alternative Cyber Performance Metrics	41
3.3	Computational Modeling Methodology & Design	41
3.3.1	Agent-Based Model Introduction	42
3.3.2	Agent Types	44
3.3.3	Environment Types	45
3.3.4	Simulation Start-up	46
3.3.5	Model Execution	48

3.3.6	Defender's Turn	48
3.3.7	Attacker's Turn	50
3.3.8	Simulation End	53
3.3.9	Model Sensitivity Analysis	53
3.3.10	Code Development	54
3.3.11	Dashboard Development	55

Chapter 4:

Research Results	56
4.1	Introduction 56
4.2	COSES Research Results 56
4.2.1	Phase I: COSES Initial Development 56
4.2.2	Phase II: COSES Scale Development 58
4.2.3	Phase III: COSES Scale Evaluation 61
4.3	Computational Model Research Results 64
4.3.1	Model Exploration 65
4.3.2	Model Sensitivity Analysis 67

Chapter 5:

Conclusions	78
5.1	Summary 78
5.2	Contributions 79
5.3	Limitations & Future Research 81
5.3.1	COSES Data Collection 81
5.3.2	Model Data Validation 81
5.3.3	Attack Type & Attacker Variations 82
5.4	Organizational Adaptation 82
5.5	Conclusions 84

References	86
Appendix A: Dakota State University IRB Approval Letter	99
Appendix B: Research Participant Consent Form	101
Appendix C: Expert Evaluation & Content Validity Tool	104
Appendix D: Cyber Operations Self-Efficacy Scales	121
Appendix E: Model Sensitivity Statistical Analysis Results	130

LIST OF TABLES

Table 3.1 Degree of Influence on Agent Performance Across Model Characteristics	44
Table 3.2 Network Node Status	45
Table 3.3 Network Node Status with Values	51
Table 4.1 Expert Analysis Feedback Krippendorff's α Results	57
Table 4.2 Polychoric Correlation of Cybersecurity items	59
Table 4.3 Polychoric Correlation of Cyber Offense items	60
Table 4.4 Cybersecurity PCA Total Variance	61
Table 4.5 Cybersecurity Multi-Variable Regression - Criterion Validity	64
Table 4.6 Cyber Offense Multi-Variable Regression - Criterion Validity	65
Table 4.7 Computational Model Independent Variables	66
Table 4.8 Baseline Parameters for Model Analysis	67

LIST OF FIGURES

Figure 1.1 Research Overview	4
Figure 2.1 Complex Systems Topics	13
Figure 2.2 Complex Systems Tools	17
Figure 2.3 Illustration of a complexity state diagram.	19
Figure 2.4 Triple Axis of Cybersecurity Dynamics Research	21
Figure 2.5 Cyber-fit Vulnerability matrix	24
Figure 2.6 Research trends across the literature	25
Figure 2.7 Research trends across the literature	26
Figure 2.8 Research trends across the literature	27
Figure 3.1 Two Research Methodologies Applied	29
Figure 3.2 Cyber Operations Performance Framework	31
Figure 3.3 Cyber Operations Self-Efficacy Assessment Construction Overview . .	34
Figure 3.4 COSES Conceptual Model	35
Figure 3.5 Theoretical Cyber Performance Latent Factor	37
Figure 3.6 Design Science Methodology	43
Figure 3.7 Cyber Performance Computational Model at Start-up	47
Figure 3.8 Computational Model Agent Phases	48
Figure 3.9 Cyber Performance Dashboard	55
Figure 4.1 Cybersecurity PCA Scree Plot	62

Figure 4.2 Baseline Parameters Runtime Length Analysis. Model length (steps)	
on both x and y axis	68
Figure 4.3 Node Count MANOVA against Availability	69
Figure 4.4 Node Count MANOVA against Sustainability	70
Figure 4.5 Friendly Count MANOVA against Sustainability	71
Figure 4.6 Friendly Skills MANOVA against Sustainability	73
Figure 4.7 Adversary Count MANOVA against Sustainability	74
Figure 4.8 Adversary Skills MANOVA against Sustainability	75
Figure 4.9 Node Count MANCOVA against Availability	76
Figure 4.10 Node Count MANCOVA against Sustainability	77
Figure 5.1 Cyber Program Management Cycle	83

Chapter 1

Introduction

The age of cyber warfare is not a near-distant future, it is today. The impact of overlooking cybersecurity requirements is quickly observed on the international, national, and local levels through massive data breaches, ransomware attacks, malware infections, and partial or complete service loss. National-level infrastructure implemented across cyber-physical systems are both commonplace and continuously exposed to new evolving threats. There is little doubt that maintaining an effective cybersecurity program is a critical cornerstone to safeguarding national interests, preventing cybercrime, and ensuring organizational continuity while protecting the privacy and interests of all internet users.

To address this growing need, individuals, organizations, and nations are dedicating increasingly greater resources to cyber operations every year, with total estimates in the trillions between 2017 - 2021 for the United States federal budget alone [1]. Of note, a significant portion of this budget is dedicated to offensive security as a means to ensure cybersecurity [2].

Cyber operations is the interdisciplinary study of cyber offense and defense and leverages tools and techniques across a wide variety of domains, including computer science, engineering, psychology, electrical engineering, and criminology. Given this breadth of scope and applicable expense, how does an organization determine what resources to dedicate to cyber operations and what impact strategic influences such as manpower, training, and network security play on tactical success? With unpredictable influences,

including new technology, software updates, malware infections, and a rapidly expanding technological footprint, the complexity of cyber operations has long stood as an enigmatic hurdle to developing a framework for predicting and evaluating cyber operations. The success or failure of a cyber operation is influenced by a wide range of factors, including strategic policies, organizational procedures, complex networks, and the individuals that attack and defend these cyber battlegrounds. Across each of these elements is a complex adaptive system upon which cyberspace operates. While a performance analysis of past cyber operations can provide a baseline assessment, it is a time-late reference that neglects to account for the emergent changes within cyber teams, networks, and evolving adversarial objectives. Organizational leaders responsible for cyber operations require actionable data for real-time effective decision-making.

1.1 Problem Statement

The lack of an effective framework to evaluate cyber operations performance based on near real-time factors forces organizational leaders to remain reactive and impairs decisions regarding high-level policy, logistical resource allocation, and network development. To date, the data collected on cyber team performance is a historical snapshot, and various complex influences such as behavioral characteristics and network status make predicting future performance difficult to ensure when personnel and computer systems constantly adapt and evolve.

1.2 Research Aims, Objectives, and Questions

This research aims to correct the disparity underlining the problem statement by establishing a cyber operations performance assessment framework via computational modeling defined by the following research objectives:

1. RO1 Develop a computational model that predicts cyber operational performance.
2. RO2 Develop a self-efficacy scale, the Self-Efficacy Cyber Operations Scale (COSES), as an input to the computational model for behavioral characteristic influence.
3. RO3 Develop a cyber performance dashboard that provides users with a simple-to-understand assessment of cyber performance based on user input.

These research objectives will aid in answering the research question: can we provide real-world assessments and predictions of performance for cyber operations?

1.3 Overview of the Framework

To accurately simulate cyber operations accounting for influential factors and produce actionable results, three fundamental elements of the framework must be developed:

1. The Self-Efficacy Cyber Operations Scales (COSES)
2. A computational, agent-based, cyber operations model
3. An operational performance dashboard

The COSES tool provides the computational model with behavioral characteristics that, along with environmental and adversarial factors, directly influence the performance of an individual or team within the cyber context. The agent-based model is a python-based software program developed to accept user-defined input regarding behavioral characteristics (provided by the COSES), environmental characteristics, including the number of agents, adversarial skills, tactics, and target network security strength. The agent-based model runs a predetermined number of simulations, resulting in performance analytics displayed in the form of an operational performance dashboard. Figure 1.1 provides an overview of the research requirements. Chapter Three, Methodology, provides

additional details regarding the development, validation, and overall construction and use of the Cyber Operations Performance Framework.

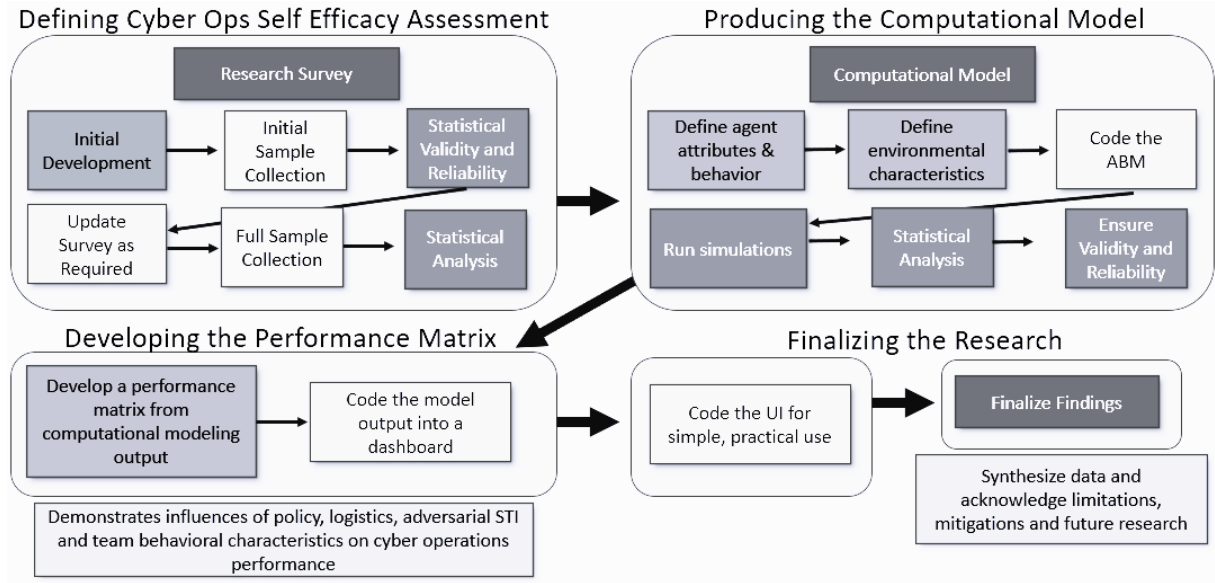


Figure 1.1: Research Overview

1.4 Motivation

Fundamental to this research is the application of complexity science to improve cyber operations. Cybersecurity remains a reactive exercise of constant responses to newly evolving threats. While every nuance cannot be accounted for, complexity science provides a basis for simulation and modeling that will encourage a lean-forward approach to cybersecurity. Additionally, strategic decisions can be better informed with near real-time tactical analysis, working to prevent a disconnect between organizational leadership and end-user performance. My pending publication, *Complex Systems Science and Cyber Operations: A Literature Survey* [3], provided an extensive discussion on the need for a scientific foundation for cyber operations and the broad benefits that complexity science affords cyber research and practice. My dissertation's research is the extension and application of this publication.

1.5 Major Theories

The foundation of this research is based on two major theoretical concepts: (1) complexity science applied through computational modeling and (2) self-efficacy as a behavioral characteristic. The following subsections are dedicated to providing a brief high-level overview of both concepts, with the literature review in Chapter Two intended to bring greater depth to each subject.

1.5.1 Complexity Science & Computational Modeling

Complexity science, or systems science, is a contemporary interdisciplinary field. Its roots are derived from the study of general systems theory and provides the basis for understanding, interpreting, and analyzing complex adaptive systems. Applying complexity science principles is critical to understanding the influence of system properties on elements within a system, directly expanding our understanding of the universe as once defined by classical science's reductionism and linear mathematics. Since its formal establishment in the mid-twentieth century, complex science has evolved into a number of influential fields, including game theory, nonlinear dynamics, systems theory, complex networks, and evolution and adaptation [4]. At the heart of complexity science are complex systems, distinguished by elements of a system that are dynamic, self-organizing, and evolving in a non-linear fashion. Recognizing that complex systems are inherently more than the sum of their parts, computational modeling provides the simulation-based tools necessary for observing emergent phenomena that would otherwise be impossible to analyze and understand [5]. Complexity science is particularly appropriate for analyzing cyber operations given the adaptive nature and nonlinear influences of cyber agents and cyberspace [6].

Agent-based modeling (ABM) is a computational modeling approach that has seen significant use in research and practical application across a wide range of fields, including

economics [7], biology [8], political science [9] and cyber research over the last twenty years [3]. While powerful in its utility, agent-based modeling is conceptually simple in design and execution, making it an ideal platform for organizational leaders to implement. ABMs analyze the actions and influence of individuals (agents) within the system with one another and their environment. Agents are autonomous, interactive, and self-organizing; they can vary from a few to millions and are programmed within the model based on simple rule sets. As often demonstrated in complex systems, these simple rules can produce unexpected results and demonstrate exponential influence, negative returns, or tipping points within a complex system [10]. As demonstrated by recent publications, agent-based modeling impacts research on elements of cybersecurity, including cyber-physical systems [11], cyberpsychology [12], and business risk assessment [13].

1.5.2 Self-Efficacy & Performance

Formally introduced as a personal behavioral construct by Albert Bandura as an element of social cognitive theory, self-efficacy is the confidence one holds in their ability to achieve success in a given pursuit [14]. Not to be confused with self-esteem, the overall self-assessment of one's worth [15]. As noted by recent research, there is a direct causal relationship between self-efficacy, performance prediction, and end-state outcome [16], resilience [17], and burnout [18]; all influential considerations in developing an effective cyber workforce. While there are perhaps numerous behavioral characteristics that define performance, a focus on self-efficacy for cyber operations is in line with current research [19] and ensures a practical scope for application in both research and practice. To capture the self-efficacy of cyber operators, the construction and validation of the Cyber Operations Self-Efficacy Scale (COSES) was completed as an element of this research. While previous research evaluated a layperson's self-efficacy with computers and cybersecurity [20] and competition participants [21], the COSES is the first of its kind: Two Likert-based situational scales defining a cyber professional's confidence given the tasks

and circumstances one is likely to encounter as a cyber operator. Statistical analysis from the scales provides direct input to the computational model as an agent characteristic.

1.5.3 Performance Management

While theoretical research is important, cyber operations is a practical field where performance can determine the trajectory for nations, organizations, and citizens applicable to nearly every facet of life. To interpret the findings of this research, the developed computational model produces a performance dashboard based on the user-defined input. The goal of this dashboard is to simplify user analysis, provide ROI input, and deliver a holistic framework that is immediately practical for organizational leaders executing cyber operations.

1.6 Research Significance

The importance and relevance of cyber operations research has never been greater or influenced more aspects of organizational or national operations and management. While cyber research takes a variety of forms and often focuses on tool development, security performance, and network or device engineering, this research aims to transform the field by reinforcing a foundation for cyber operations with complexity science, both theoretically and with practical application. A number of research efforts have already implemented agent-based modeling as a tool for cyber analysis [22], [23]. However, this research is the most comprehensive execution to date, providing a single framework that incorporates behavioral, environmental, and logistical properties into a model that illustrates the influential impact of these characteristics in an operational performance dashboard. With the results of this research in hand, organizational leaders can make improved decisions across a wide range of operational efforts, and cyber research can take another step forward in becoming a unified scientific field better equipped to predict and analyze the

complexities of cyberspace.

Chapter 2

Literature Review

At the heart of this research is an aim to produce a sophisticated computational model with simple mechanisms resulting in clear yet profound results. Organizational leaders with little to no technical background in programming or computational modeling can effortlessly run a variety of scenarios and then compare results to determine logistical influences on cyber operations. Underlying this simplicity are the research fields of two major areas of study: complexity science and social cognitive theory. The following chapter provides a formal introduction and review of the relevant research regarding complexity science and self-efficacy.

The literature review strategy across both areas of study focused on developing an understanding of each in broad terms and then focusing on their application to cyber operations. Each of these fields is relatively young, both having approximately forty years of development, though demonstrating incredible growth and breadth across a variety of domains. Many of the relevant topics have entire texts and fields of study dedicated to their understanding. The presentation here is designed to present the breadth of applicable subject matter in a general sense highlighting significant contributions while delivering depth focused on the available cyber operations research. As the field of cyber operations is also relatively young, the depth of literature related to cyber operations reflects a time period of approximately twenty years.

2.1 Complexity Science Introduction

To defend those that use the internet, cybersecurity researchers and practitioners develop procedures and implement patches in an effort to thwart malicious actors, but to what end? The paradigm of cybersecurity is analogous to a dam that, despite best efforts, continues to spring leaks. While cybersecurity professionals develop solutions for today's botnet, ransomware, and rootkit, attackers continue to create and exploit software features that become tomorrow's zero-day exploits. Given the current paradigm of playing catch up to keep up, cybersecurity will remain a responsive approach to malicious attacks unless a radical change occurs in the current approach to understanding network, information, and computer security.

Although there is no silver bullet solution, an effort must be made to establish a better foothold in combating cyber attacks. Part of the challenge lies in the incredibly diverse and seemingly unpredictable nature of cyberspace. To better understand and, at some level, control cyberspace, researchers have begun examining cyber operations from a complexity science perspective.

The study of complexity science is a growing interdisciplinary field of research. While complexity science lacks a formal or universal definition, its key concepts include emergent phenomenon, dynamics, evolution and adaptation, collective non-deterministic behavior, and self-organization. Researchers of complex systems often find it necessary to develop simulations and models to understand and communicate the nature of a complex system effectively. Such complex systems span nearly all domains, including ecology, psychology, mathematics, economics, and computer science. Given the internet is a complex system and the nature of cybersecurity to safeguard its use, we believe examining cyber operations with a complexity science perspective is an important and necessary step to producing a safe and secure internet.

2.1.1 Early History

The formal study of complex systems as a modern scientific endeavor took root in the twentieth century and was established as a recognizable field of study in the 1980s [24]. Prior to that, complexity science evolved from contributions across multiple disciplines. Five areas of early study are cited as fundamental to the current state of complexity science: (1) mathematics of complexity, (2) general systems theory, (3) complex systems theory, (4) cybernetics, and (5) artificial intelligence [25]. Each of these areas has impacted the understanding of complex systems both independently and interdependently, which continues to result in definitions and applications of complexity theory that vary from one domain to the next.

The development of general system theory (GST), founded by Austrian biologist Ludwig von Bertalanffy was an early contributor to generalizing system analysis [26]. In the mid-20th century, Bertalanffy identified the increased isolation of scientific fields. He also noted that despite little communication across these evolving boundaries, researchers from different domains were independently tackling challenges derived from the chaotic nature of nonlinear systems [27]. First proposed in the 1940s and then published in 1968, Bertalanffy’s general system theory suggested that complex systems share fundamental universal principles across all domains that can be understood and mathematically modeled. His theory was rooted in examining systems characterized by autonomy, creativity, and dynamism and has produced theoretical developments across multiple fields, including complexity, cybernetics, systems theory, and systems engineering.

Building on similar principles at the time, cybernetics and artificial intelligence produced important contributions to studying complex systems. Norbert Wiener, using the term cybernetics in his 1948 text on the subject [28], proposed that feedback loops are fundamental to learning and that such dynamic systems could be leveraged in developing machine learning. Walter Pitts, the founder of artificial intelligence, was a student of

Wiener at MIT. Working with Wiener and Warren McCulloch, a neurophysiologist, Pitts developed computational models forwarding the concept of neural networks [29].

Through the efforts of the Santa Fe Institute (SFI) established in 1984, complexity science exposure grew across scientific domains and international borders. SFI founders and early contributors, many from the Los Alamos Laboratory, represented a swath of scientific fields, including economics, physics, biology, and chemistry. Hosting international conferences for discussion and collaboration, research focused on complex systems quickly expanded. Researchers such as Goerge Cowan, Murray Gell-Mann, and David Pines provided the foundation from which complexity science grew to the paradigm-altering scientific field it is today [30].

2.1.2 Terminology

Complexity science continues to be an interdisciplinary field of research, and its definitions and applications vary from one domain to the next. Across each research field vested in understanding complexity, different terms are used to underscore the principles of complex science, including complexity theory, complex systems analysis, system of a system, system dynamics, chaos theory, systems thinking, complex networks, and complex adaptive systems. Nonetheless, the significance of complex systems analysis is so profound that the scientific method itself has been refined to account for the insights provided through our increased understanding of complexity science [26].

As a baseline for cyber operations research, we provide the following definitions to clarify complex systems terms for current and future research. The field of complexity science, or complex systems science, encompasses the entirety of complexity research and represents the domain as a whole. A complex system is a system in which interdependent elements interact to produce characteristics that define the system beyond the characteristics present when analyzing the individual elements independently. This phenomenon, known as emergence, underlines the distinction between complex systems and simple



Figure 2.1: Complex Systems Topics

systems (elements and the system characteristics are consistent, behavior/relationship of elements are fixed) or disparate systems (the behavior/relationship of elements within the system are not related or truly random). Regarding cyber security, [31] asserted: “A security property of a cybersystem exhibits emergent behavior if the property is not possessed by the underlying lower-level components of the cybersystem” (p.1). Complex systems analysis provides a means of understanding systems beyond classical mathematical tools, such as differential equations and statistics, emphasizing the complexity and correlation of elements within the system [26].

Self-organization, the rise of emergence spontaneously over time, can be observed in

complex adaptive systems when analyzing the element's interdependent effects on the system as a form of non-directed system evolution over time. These systems will typically adapt as their interdependent elements develop new responses based on changes within the system. Often a complex system will be referred to as dynamic, highlighting the sometimes radical changes observed within a complex system when its nonlinear nature becomes apparent. A simple example is that of the double pendulum, the motion of which is bound by differential equations resulting in a dynamic and chaotic track of movement.

2.1.3 Fields of Study

Within the scope of complexity science are different areas of study that represent the various disciplines from which they evolved. Sayama, Director of the Center for Collective Dynamics of Complex Systems at Binghamton University, defined seven areas of focus within complexity science as topical clusters: (1) game theory, (2) nonlinear dynamics, (3) systems theory, (4) pattern formation, (5) evolution & adaptation, (6) networks, and (7) collective behavior [4]. Adopted from [4], Figure 2.1 provides a visual perspective of the various research fields related to Sayama's topical areas. The following paragraphs outline a concise introduction and correlation of the topical areas related to cyber research examples.

Behavioral game theory is a multidisciplinary field based on mathematical models representing rational or irrational decision-making across human populations. Cyber researchers utilize game theory to simulate cyber operations dynamics [32], [33], [34], adapt machine learning for cyber security [35] and as a tool for creating cybersecurity assessments [36]. Findings from [32] demonstrated the utility of a complexity science perspective in cyber research to quantify the impact of network misconfiguration across attacker types and network setups. [36] highlighted the advantage of decision support gained through cyber operations complexity analysis. Recent cyber-based research contributions [37], [38] provide excellent continued reading beyond this initial introduction.

Nonlinear dynamics, popularly known as chaos theory, focuses on systems in which a change of input is not mathematically proportional to the output. Related research has aimed at understanding cyber incident frequency to improve short-term incident prediction [39], managing cyber-emergencies [40], and interpreting cyber warfare law [41]. As early as 2006, researchers positioned chaos theory to predict the outcome of cyber operations through the recognition that the average of hundreds of simulations can normalize results [40].

Systems theory research is the application of understanding and problem-solving complex systems challenges. Given this is the broadest of the complex science areas, it can be applied across all domains, though the term is commonly used in social sciences research such as psychology, business management, and organizational behavior. Research directly related to system homeostasis (a system’s steady state of equilibrium), feedback (or cybernetics, when a system’s output influences the inputs), and system dynamics (measurement of a system’s change over time) are based on systems theory. Fundamentally, this is the foundation for research regarding solutions dedicated to cyber-physical systems (CPS) and the internet of things (IoT).

Pattern formation is the recognition and research of complex systems based on self-organization into naturally occurring identifiable patterns. From pattern-formation rises cellular automata, well known for the “game of life” example [42]. A cellular automaton is a cellular grid bound by explicit rules defining a finite set of states and how states update over time. Cyber research has used cellular automata to calculate cybersecurity risk based on CPS [43], smart grids [44], and cascading failures [45].

The area of complex evolution and adaptation is dedicated to understanding how adaptation occurs in biological and technological systems. Specific areas tangent to cyber operations include artificial intelligence, artificial life, and machine learning. Given the current popularity of artificial intelligence and machine learning, examples of relevant cyber research are ubiquitous throughout the literature.

The study of networks is fundamental to cyber operations; however, not all networks are complex networks. Specific areas of complex networks include dynamic networks, adaptive networks, scaling, and graph theory. A great deal of research regarding complex networks is found throughout the literature based on cyber attacks [46], [47] as well as various areas of cybersecurity [48], [49].

The final topical cluster is collective behavior. In addition to social dynamics, collective intelligence, and synchronization, agent-based modeling is a key research area of complex science and underlines many cyber research articles based on complexity science. Agent-based modeling (ABM) simulates dynamic systems through the use of interdependent agents who influence one another and the system according to a set of predetermined rules. Agent-based modeling provides three considerable benefits: it illustrates emergent phenomena associated with complex systems, models are relatively simple to design and observe, and results can be gained quickly across many runs of the simulation. Cyber-related research includes both offensive and defensive models, discussed in further detail below.

2.1.4 Complex Systems Tools

Given the breadth and depth of complex systems subject matter, developing a meaningful appreciation can be aided through familiarization with the field’s tools. Shalizi [50] presented a comprehensive approach to organizing complexity science tools by categorizing them into three areas based on purpose: building and understanding models, measuring complexity, and analyzing data (Figure 2.2). The remainder of this section is dedicated to reviewing complex systems tools that can be leveraged in cyber operations research and performance.

Artificial intelligence (AI) continues to experience significant growth and development across all domains, including cyber operations. Statistical learning theory is a framework for developing and evaluating algorithms and models fundamental to the prediction re-

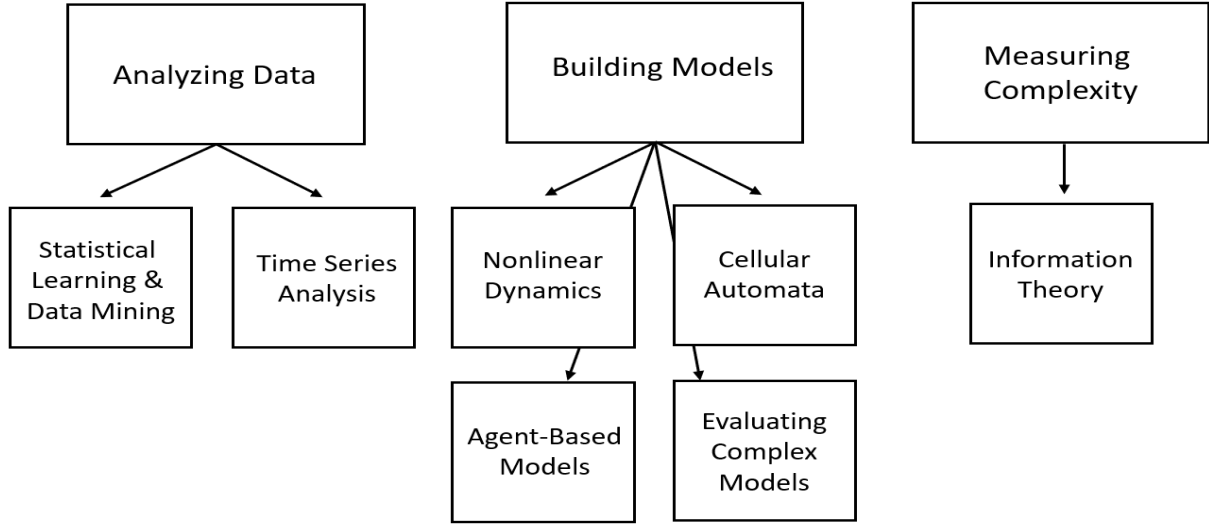


Figure 2.2: Complex Systems Tools

quired for systematic learning. Although modeling complex systems may reveal system phenomena not evident with statistical analysis, statistics can still provide meaningful insight in data analysis relevant to complexity science [51].

Due to the complicated nature and number of variables in many complex systems, a valid and reliable approach to analyzing output causality can provide significant insight into the system's operations. Research by Razak and Jensen [52] demonstrated the use of transfer entropy, a time series statistical model designed to analyze complex systems, as a means to infer causation from correlations that accurately forecast future values. Through the use of applicable models and tools, complex systems can be visually displayed through graphs and measured using probability models to facilitate interpreting system characteristics. As each of these areas of analysis is a field of research unto itself, we present recently published research applicable to cyber operations as a foundation for discussion.

Naturally, statistical learning is inherent in cyber research based on artificial intelligence. Throughout the literature, there are notable cyber research examples dedicated to statistical learning theory, including fake website detection [53], cybersecurity and biometrics [54], and cyber threat detection [55], [56]. These examples each explicitly identify the

relevant contribution statistical learning provided to developing an effective framework within a cyber context.

While there are variations in modeling complex systems, the most prominent are cellular automata, complex network models, and agent-based models (ABM). Agent-based modeling represents the most common approach of the three within the cyber literature and has significantly increased in popularity over the last twenty years as a reliable tool for analyzing and presenting complex phenomena. Based on agents that are autonomous and interactive, ABMs enable observing the complexity within a system that would otherwise be difficult to extract and understand. This approach is used in modeling phenomena across a wide range of fields, including political science [57], sociology [58], economics [7], epidemiology [59], [60], biology [8], and chemistry [61]. Specific to each system and phenomenon being studied, agents may take the form of individual people, infections, fire, flooding, or independent systems [62]. Agents within an ABM may vary from a few to millions, and while initial programming defines basic agents with identical characteristics and learning rules, the variations experienced due to interactions with one another and the environment often result in a wide variety of agent actions and system adaptation.

Information theory, specifically information fluctuation, provides a quantitative measure of complexity within a given system [63]. While an in-depth analysis is beyond the scope of this paper, a cursory introduction will help provide context to further applicable measurements on a macroscopic scale. As presented in [64], complexity is a matter of probability, and its presence can be measured based on the states of information available, presented in Equation 2.1:

$$\begin{aligned}
 I &= \log N = -\log P \\
 P &= 1/N
 \end{aligned}
 \tag{2.1}$$

Where I is information, P is probability, and \log is a logarithmic operation, symbolic

of the reverse exponential function defined by the probability equation that represents the content of information in a complex system. The negative $\log P$ produces increased information through decreasing probability. In a complex system, order and chaos occur in variations of alternating states producing a system that, at times, may be either predictable or unpredictable. [64] illustrated these states within a system using a diagram similar to that presented in Figure 2.3. Arrows converge on the circles representing stability and order. When arrows diverge from the circle, it represents chaos. The numbers within the circles represent various potential states within the system. The arrows then have a forward conditional probability $P_i \rightarrow_j$ and a reverse conditional probability $P_i \leftarrow_j$ (not displayed), indicating the probability of the current state and future or past state respectively [64].

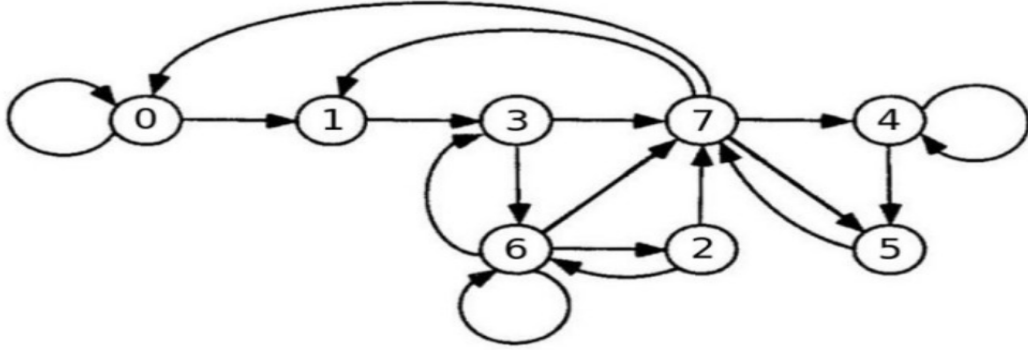


Figure 2.3: Illustration of a complexity state diagram.

Once the probability of a state is defined, net gain information Γ represents transitions from the present to the next state, and when balanced against the weighted mean or average using standard deviation $\langle \Gamma \rangle$ and calculated for multiple transitions, the complexity of a system can be measured using Equation 2.2:

$$\Gamma_{ij} = \log P_i - \log P_j = I_j - I_i \quad (2.2)$$

The above introduction is only a summary based on [63] [64]. Interested readers are encouraged to review these papers for a detailed examination and illustrated examples.

2.1.5 Cybersecurity Dynamics Framework

Across the cyber-complex literature landscape, a notable and prominent contribution emerged: Xu’s cybersecurity dynamics framework [65]. The following subsection outlines specifics regarding cybersecurity dynamics and its potential influence on cyber research.

Recognizing the need for a formal scientific foundation in cyber research, Xu developed cybersecurity dynamics [65], [66], [67]. While the name implies a strict focus on cybersecurity, its principles, and applications apply to both offensive and defensive cyber. Based on a macroscopic perspective, or *macroscopic cybersecurity*, Xu applied many of the principles of complexity science, including a systems-level analysis, acknowledging the emergent, adaptive, and dynamic nature of cyber systems, and leveraging applicable models to interpret system-level characteristics.

Within Xu’s cybersecurity dynamics framework, two core principles define the scope of cybersecurity dynamics: core research objectives and the triple research axis (Figure 2.4). The core research objectives are based on understanding, managing, and forecasting cyber phenomena [66]. As such, cyber dynamics drives researchers to develop descriptive, prescriptive, and predictive models. When examined holistically, each of these objectives supports and ultimately drives forward one another, providing a means to interpret and validate data observed across cyber systems. Descriptive modeling provides an abstraction and simplification of model characteristics to better grasp agent-level influences and system adaptation. Often these models are preliminary simulations designed to ensure the simplest approach to modeling system functionality. Cyber descriptive models can be used to understand attack-defense scenarios in a variety of instances, including botnets [68] mobile networks [69] and organizational manufacturing [36]. From descriptive models evolve prescriptive and predictive models. Prescriptive models interpolate cyber datasets to evaluate cyber operations and are often used in team simulations [70], and security evaluations [32]. Predictive models extrapolate cyber datasets to forecast the impact that

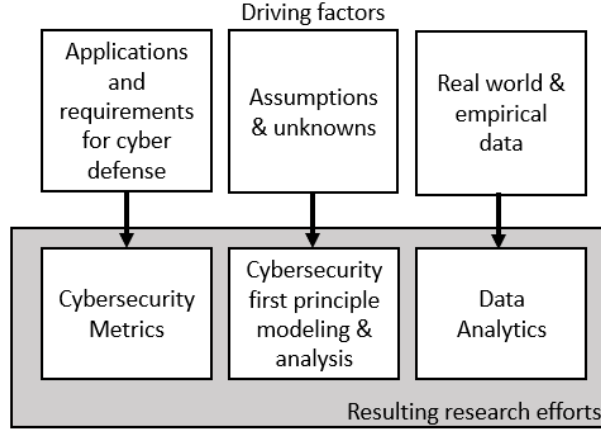


Figure 2.4: Triple Axis of Cybersecurity Dynamics Research

threats and security measures make on cyber operations. Through the use of prescriptive and predictive modeling, researchers can evaluate policy and validate assumptions posed through the descriptive modeling process [66].

2.1.6 Agent-Based Modeling

As early as 1999, Fred Cohen, the father of computer virus defense and pioneer of network modeling [71], identified the lack of research regarding complex systems and cyber, despite applicable advantages in modeling and simulation already determined for cyber operations [72]. Citing challenges such as the complexity of cyberspace, lack of quality data, inconsistency in practice and research, and the rate of evolving technology, Cohen recognized hurdles that remain true to this day. Furthering the point, Cohen emphasized the shortfalls of statistical analysis, the standard modeling and simulation tool of the time, for its inability to demonstrate attacks in parallel or simulate attacks based on timing. To improve on the few previously published models simulating cyber operations, Cohen actively balanced accuracy in complexity with computational performance limitations. Leveraging a cause and effect approach, Cohen et al. developed a novel model simulating attack and defense of a cyber environment approximating the time to attack and defend as a prominent characteristic [73]. Although the model is not explicitly agent-based, at-

tackers and defenders (or *agents*) are defined by various characteristics, and the cause and effect and timing-based nature of the model produces an emergent quality of the system/-model in which elements of the cyber environment become more or less exposed to attack throughout the simulation. Through analysis and multiple simulations, the researchers identified a critical point within the system in which the defense of a system increases radically despite minimal increase in defender ability. They also recognized that a perfect defender does not always succeed, allowing an organization to fall victim to attacks when certain time and ability thresholds of attackers are set. Ultimately, they presented the nonlinear results of attacker success, implicitly highlighting the applicability of cyber operations to complex systems analysis and demonstrating its power through modeling and simulation.

Building from Cohen et al.'s model, cyber research applying formal agent-based modeling techniques began to grow, including influential contributions by Kotenko to include examining various attack and defense scenarios and simulating cyber-wars across the internet [74], [75], [76], [77].

Kotenko et al. [70] followed up this research with agent-based modeling approaches to defending against botnet attacks and analyzing cooperation versus competition of teams [78] to determine how best to simulate their actions and an analysis of cyber attack and defense for homeland security [77]. Additional agent-based modeling research analyzing cyber operation teams of note include [79], and [80].

In 2011, Grunewald et al. [81] introduced NeSSi2, an agent-based network security simulation framework designed to illustrate various attack vectors versus security solutions. Built on a three-component architecture, the framework consists of a GUI, agent-based simulation back-end, and results database. The framework adopted three context models: the network, attacker, and related interdependencies and simulated attacker intent, opportunity, capability, and preferences demonstrated through attacker actions. The researchers reported successful findings when leveraging NeSSi2 to deter-

mine effective intrusion detection strategies versus malicious worm propagation. While recognizing the lack of real-world validation, the results provide insight beyond historical reference and enable cybersecurity professionals to assess security strengths beyond the ubiquitous threat assessment frameworks solely relied on by many organizations. Further research highlighted NeSSi2’s scalability, fidelity, and extendable nature [82], in addition to research on specific cyber threats such as large-scale DDoS attacks [83].

Following an analysis of the current state of cybersecurity illustrated through application whitelisting, Norman et al. [84] aptly illustrated the importance of applying complex scientific principles to cybersecurity to analyze and solve cyber challenges. Through the use of a fictional government, the researchers used agent-based modeling to run simulations of a fictional government conducting cybersecurity via whitelisting applications from a top-down (all programs whitelisted with the exception of those approved for network use) versus bottom-up (programs are whitelisted once a known vulnerability is identified) perspective. Despite expecting a top-down approach to significantly bottleneck operational productivity, the agent-based model demonstrated that the two approaches had very similar time throughput measuring organizational success via application processing times. While an exceptionally simple model, it grounded organizational decision-making regarding cybersecurity through empirical evidence rather than simply leaning on common sense or existing precedent.

Across the literary landscape, a single cyber-based *performance* model was identified, the Cyber-Forces Interaction Terrain (FIT) Simulation Framework [22]. The cyber-fit model is designed specifically to support military operations, modeling military forces and terrain (computer systems). Using NetLogo, an agent-based modeling software, the authors simulated three terrain types (military base, tactical or industrial location) and cross-threaded them against three terrains (analogous to computer systems) to define three vulnerability rates (Figure 2.5).

To implement the agents, Dobson and Carly defined offensive and defensive forces.

Terrain type	Base	Tactical	Industrial
Type 1 (Networking)	Low	Medium	High
Type 2 (Servers)	Low	High	Medium
Type 3 (Users)	High	Medium	Low

Figure 2.5: Cyber-fit Vulnerability matrix

Defensive forces take action to update vulnerable terrain to secure terrain. Offensive forces conduct one of four attack standards associated with one of the terrain types (1) Random - attacks all types, (2) routing protocol attack - attacks Type 1 (Networking Systems), (3) denial of service - attacks Type 2 (Server Systems) and (4) phishing - attacks Type 3 (User Systems) [22]. Through running simulations, the researchers answered a series of questions regarding operational logistics, such as ideal force allocation for cyber defense and impact on network security based on changes to attack scenarios.

The Cyber-FIT simulation framework represents an important significant step forward in performance modeling for cyber operations. As noted by Dobson and Carly, it lacks rigorous validity via empirical data (inputs don't reflect real-world values), and while the model demonstrates proof of concept, it's unable to be applied to real-world applications. The cyber operations performance framework builds on these developments by applying input from the Cyber Operations Self-Efficacy Scale (COSES) to reflect cyber operator behavioral characteristics while also accepting inputs regarding real-world network status and operator skills and capabilities.

As noted by Wilensky and Rand [10], agent-based models may be employed for the following eight use cases: (1) description, (2) explanation, (3) experimentation, (4) providing sources of analogy, (5) communication/education, (6) providing focal objects or centerpieces for scientific dialogue (7) as thought experiments or (8) prediction. As seen across the cyber literature, each of these use cases is applicable to examining and understanding the cyber environment.

While not the only modeling and simulation framework for demonstrating complex systems, the ease with which researchers can develop, observe, and experiment with emergence compared to alternatives can not be overstated. Researchers with little programming experience can develop agent-based models with free open-source software, including NetLogo [85], Repast Suite [86], and StarLogo Nova [87]. These and additional software options with varying strengths and learning curves are available to examine and simulate cyber operations across all OS platforms. With a combination of community support, in-depth online tutorials, and resources such as <https://www.comses.net>, readers are encouraged to explore the incredible potential of computational modeling beyond the scope of this article.

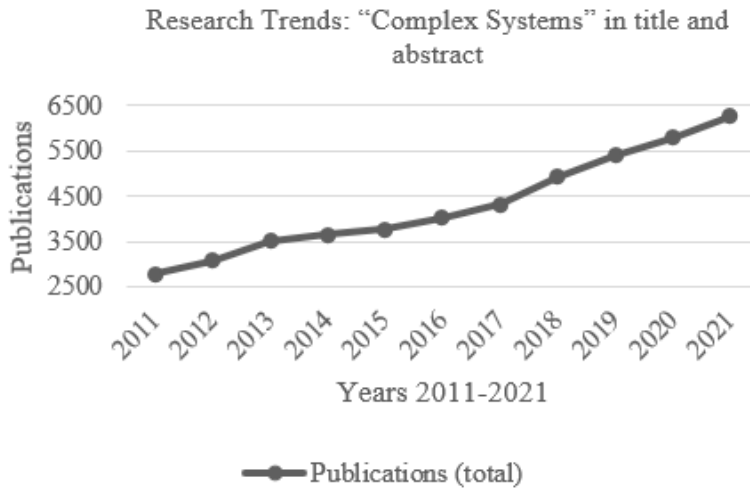


Figure 2.6: Research trends across the literature

2.1.7 Trends in Research

Despite the clear applicability, cyber operations represents only a fraction of the literature based on agent-based modeling. We conducted a trends analysis through Dimensions.ai, a site dedicated to providing comprehensive data on published research. To establish context, we begin with the key phrase “complex systems” (quotes applied), where there were approximately 38,000 articles filtering for “complex systems” within the title and

abstract. Fig. 2.6 presents our findings across each year of the last decade of research.

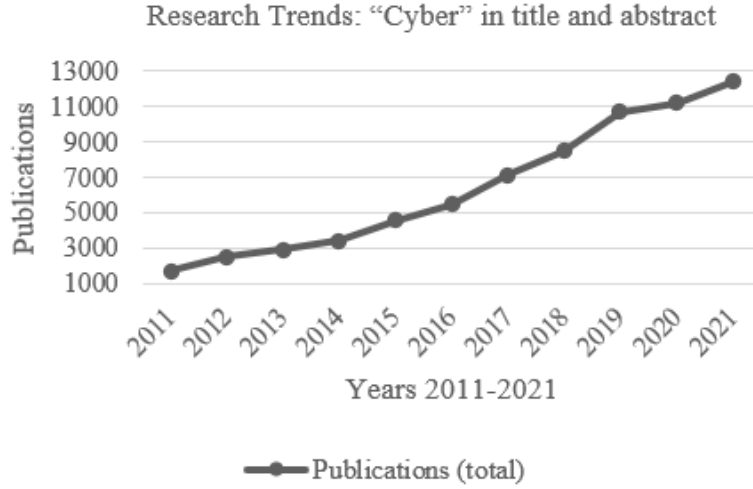


Figure 2.7: Research trends across the literature

Due to the nature of complexity science jargon across its various domains of applied research, this surely represents a small sample of all research reflective of the subject matter. Despite this, the number of publications and applicable citations has grown on average over the last ten years, as depicted in the graph. During the same period, the term “cyber” appears in the title and abstract of just over 76,000 journal articles (Fig. 2.7).

Conducting the search inclusive of both “complex systems” and “cyber”, we discovered 736 publications across the decade of research with both our terms (Fig. 2.8), a small fraction of the overall literature. As was identified by reviewing the terms independently, the trend reflects a general increase in publications year over year. It’s also important to note that not every article within our combined search is specific to cyber operations applied through a complexity science perspective. Ultimately, the findings indicate thousands of research articles in which either concept is central but relatively few at the intersection of both fields. When analyzing the publication source classifications, a majority of the research articles are related to artificial intelligence and information systems, with only a small fraction represented by distributed computing. While we have

demonstrated the impact of applying complex systems in cyber research, only a fraction of the cyber literature explicitly references or examines complexity principles.

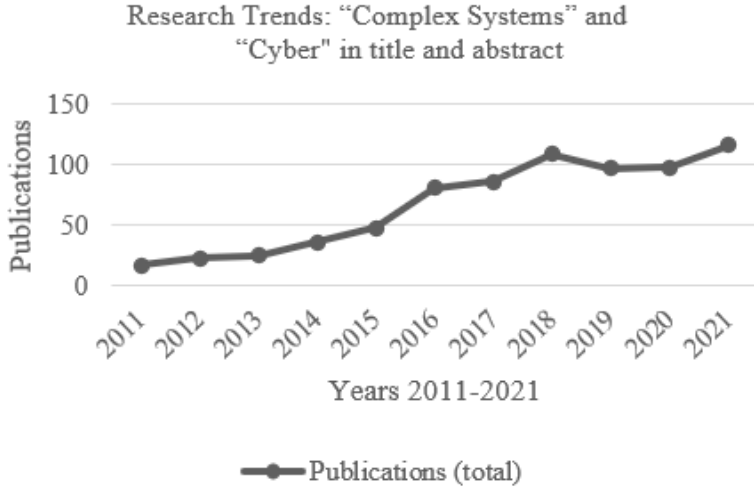


Figure 2.8: Research trends across the literature

2.2 Self-Efficacy and Performance

Self-efficacy is a personal behavioral construct developed as an element of the social learning theory proposed by Albert Bandura [14]. Bandura’s social learning theory expanded on classical and operant conditioning by asserting that learning often occurs not only from direct reinforcement but also via observation, as was demonstrated via the well-known Bobo doll experiment [88]. The social cognitive theory, an expansion of the social learning model, defines four key cognitive processes applicable to goal realization: self-observation, self-evaluation, self-reaction, and self-efficacy. While each of these interrelated behavioral constructs plays an important role in performance, Bandura asserted self-efficacy as having a causal impact on performance [89]. Since then, research efforts have continued to demonstrate the power of self-efficacy to predict performance across a wide range of areas, including academics [90], mental health [91], [92], first responders [93], entrepreneurship [94], [95], and physical activity [96] to name a short few. It’s important to note that self-

efficacy is not a universal construct but rather a belief specific to a distinct domain or task and should not be confused with self-esteem, locus of control, or outcome expectancies [97].

While there are a number of self-efficacy assessments developed for cybersecurity, previous research has focused almost exclusively on lay-person confidence in achieving cybersecurity policy requirements [98], [99], [100], cyberbullying [101], [102], [103], or student learners [104], [105]. As noted by recent research developing a needs assessment for cyber operations performance, organizations currently lack an effective method to evaluate cyber team performance that accounts for team behavioral characteristics [106] such as self-efficacy. Although there are self-efficacy scales aimed at evaluating staff confidence in effective cybersecurity practices, the behavioral dimensions (e.g. “password security skills” and “learning security skills” [100] do not apply to cybersecurity operators well-versed in cybersecurity requirements conducting high-level cyber operations.

Chapter 3

Research Methodology

At the heart of every research endeavor is the methodology, defining the direction, distance, and trajectory of one’s scientific inquiry. From the methodology springs the experimental design, as does the foundation of a house support the frame of a home.

3.1 Introduction

Within the Cyber Operations Performance Framework are three fundamental constructs: (1) the Cyber Operations Self-Efficacy Scale (COSES), an independent tool providing users a behavioral assessment of operator confidence in achieving cyber mission success, (2) the computational model, an agent-based modeling framework designed to simulate cyber operations, and (3) the performance dashboard, a visual representation of the computational model results.

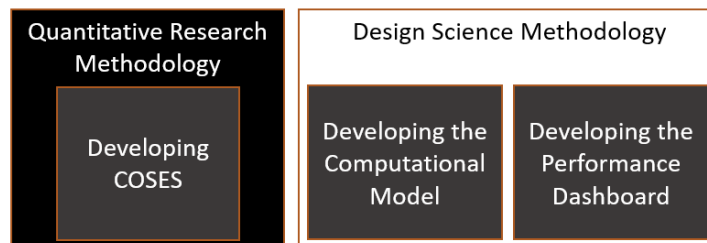


Figure 3.1: Two Research Methodologies Applied

Given the distinct nature of each of these tools, the research methodology used was

two-fold, recognizing that each interdependent construct benefits from a methodology applicable to the specific tool being developed. In accordance with a standard psychometric approach, the COSES was developed using a quantitative methodology, while the computational model and dashboard were developed using design science methodology (Figure 3.1). This chapter outlines applied research methodologies and experimental design and provides additional context for the development of the Cyber Operations Performance Framework, including construction and validation.

3.2 Cyber Performance Metrics

The Cyber Operations Performance Framework is based on three fundamental performance factors, each of which directly influences the computational model: (1) behavioral characteristics, (2) skills & expertise, and (3) the target’s network characteristics. Commonly known as KPIs (key performance indicators), determining and leveraging applicable metrics ensures the performance model more closely replicates a given real-world scenario. Equation 3.1 introduces this as the fundamental equation defining primary influences in cyber operations performance.

$$Cp = Bc + Sk + Ns \quad (3.1)$$

Where Cp is cyber performance, Bc is behavioral characteristics, Sk is skills and expertise, and Ns is the target network strength. While there are more specific KPIs that provide key data points for organizations to understand, assess, and benchmark network status and cyber capability, the Cyber Operations Performance Framework is intentionally defined by the above basic metrics to ensure its applicability is broad. Within each of these factors, a set of quantifiable metrics are passed to the computational model to influence the simulation. These factors, when combined with the number of active cyber operators, form the computational model’s algorithm. The output of the model’s algorithm results

in the performance dashboard (Figure 3.2). The following paragraphs illustrate how each of these metrics is defined.

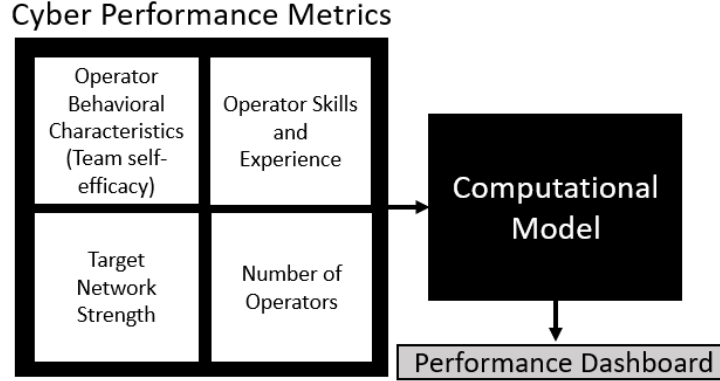


Figure 3.2: Cyber Operations Performance Framework

The behavioral characteristics factor is derived from the Cyber Operations Self-Efficacy Scales (COSES). Discussed in greater detail below, the COSES consists of separate self-efficacy assessments for cybersecurity and cyber offensive operators. Scoring is achieved by adding each response based on the 10-point Likert scale. The total score for each operator is then divided by the number of items on the scale (arithmetic mean) to produce a final numerical score of one through ten. Once the total score of all operators is obtained, one can obtain the team self-efficacy by adding the total scores of each member and dividing by the number of operators (arithmetic mean of the team member’s scores). Through successful and failed attempts and inter-agent interaction throughout the simulation, an operator’s self-efficacy will increase or decrease incrementally. Equation 3.2 defines the function of team self-efficacy as an element in the cyber performance equation.

$$Tse = \frac{1}{n} \sum_{i=Se}^n x_{Se} \quad (3.2)$$

Where Tse is team self-efficacy, and Se is each operator’s individual score divided by the number of answers evaluated. Adversary operators receive an initial team self-efficacy of seven and also increase or decrease throughout the simulation.

Friendly and adversarial skills and expertise are selectable inputs based on a scale of

one through ten. As the counterpart to self-efficacy, the skill and expertise characteristic allows users to influence a team’s performance by applying a numerical score based on their capabilities. This factor is distinct from the self-efficacy score as an operator may be confident but low in real-world skill or have a great deal of real-world experience and skill but lack confidence in completing a specific task. Contrary to public opinion, the Dunning-Kruger Effect [107] demonstrated individuals with limited knowledge and skill often overestimate their ability while those with greater skill and experience may downplay their capabilities. While developing and prescribing specific criteria for rating skills and expertise (as has been achieved for self-efficacy) might improve metric reliability, a rigid, forced fit was not desired. Applying an organization’s existing standard for evaluating skill and expertise will both expand and simplify the use of the framework. Estimating an adversary’s skills and expertise allows end-users to evaluate this influence as they would their own skills and team self-efficacy.

Network strength represents the technological environment as a measure of the network’s overall security. This characteristic is user-defined and, as with the skills and expertise element, is entrusted to organizations to leverage existing measures or develop applicable tools to evaluate target network security.

The remaining subsections provide greater detail regarding the development and theory of each of these significant elements. Chapter Four discusses the results following development, analysis, and implementation.

3.2.1 COSES Methodology & Design

The Cyber Operations Self-Efficacy Scales (COSES) represent an important cornerstone of the Cyber Operations Performance Framework and directly distinguish this research from previous efforts at cyber performance modeling, as discussed in Chapter Two. The COSES ensures the computational model accounts for the behavioral characteristics of friendly cyber operators conducting the operation being simulated. The COSES is a

simple, standardized survey tool able to capture sufficient and reliable data points to inform the computational model of influential behavioral characteristics, specifically, an operator’s cyber self-efficacy in achieving cyber operational success. The design goal is a tool simple and short enough to quickly collect behavioral characteristics that mitigate historical data challenges such as team churn or operator growth and changed confidence with experience. The value of the COSES is in its simplicity and significance, able to accurately capture the applicable behavioral characteristics critical to cyber operations.

Psychometric scales and assessments rely on a quantitative methodology to leverage the appropriate statistical analysis to meet the high standards of scientific rigor. Albert Bandura, responsible for developing the social cognitive theory and general self-efficacy construct [14], published a guide specifically for developing self-efficacy scales [97]. This publication, along with recent research within the fields of cyber operations, social sciences, and psychometrics, directly guided the construction of the Cyber Operations Self-Efficacy Scales (COSES). The following diagram is an overview of the applied steps to complete the construction and validation of the COSES (Figure 3.3). The diagram highlights the three phases of scale development introduced by Boateng et al., including item development, scale development, and scale evaluation [108]. Each of these phases is further divided into steps that ensure rigorous scale validation and reliability. The following subsections describe each phase in greater detail. The results achieved from developing the COSES are outlined in Chapter Four.

3.2.1.1 Phase I: Initial Development

Initial development consisted of a literature review, item development, expert analysis, and scale design. To begin, a thorough review of the literature determined no applicable scale existed for the intended practical and research applications. Once confirmed, item development began.

Given cyber operations is a multi-dimensional field, COSES development required a

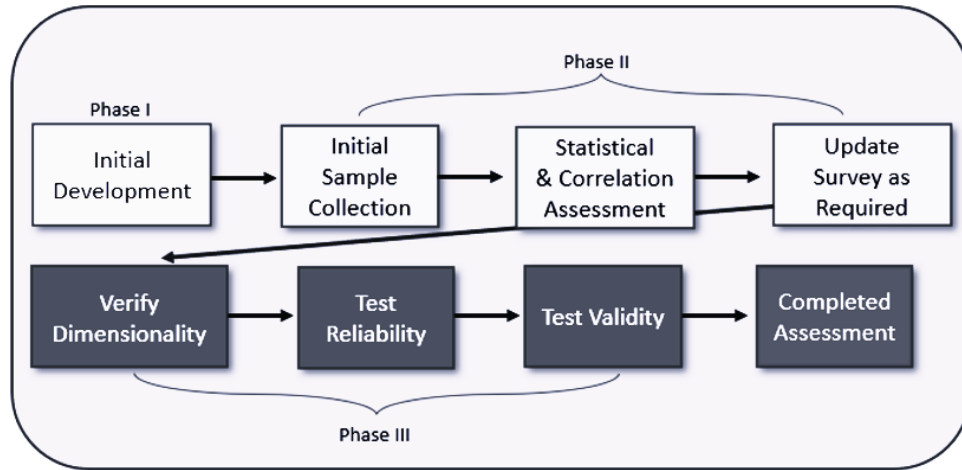


Figure 3.3: Cyber Operations Self-Efficacy Assessment Construction Overview

balance between accounting for domain elements across the entirety of applicable cyber functions and maintaining manageable scope. To achieve this, preliminary domain elements for question development were established using a deductive approach through the literary review, which illustrated domain areas applicable for the subject to master [108]. Three primary domains were selected to provide the appropriate breadth of context in determining cyber operation self-efficacy:

1. Access Control
2. Server and Network Security
3. Software Security Architecture and Design

As cyber operators are typically trained primarily as defenders or attackers, scale items were developed and characterized with this in mind. From this list, survey questions in a Likert format were developed with a situational style orientation to direct the respondent to answer based on a potential real-world scenario. As cyber operations generally occur in tangible phases, the COSES is presented based on three phases of operation:

1. Preparation & Weaponization
2. Intrusion

3. Active Breach

An overview of the COSES design outlining domains, dimensions, and phases is presented in Figure 3.4. As a matter of logistical constraints and practicality, a cross-sectional internet survey design was selected, using Jotform (jotform.com/surveys) as the platform for survey delivery and data collection. Research identifying the reliability and validity of e-survey design dates back to the late 1990s [109] with findings highlighting equivalent reliability and validity across online and paper-based formats [110] with more recent research reflecting a ten times cost savings using online formats with a negligible difference in response rate [111]. Though the internet survey method is not without challenges related to coverage, sampling, measurement, and bias [112], the advantages in cost, design, and ease coupled with the mitigations discussed within Chapter Five drove the choice to leverage this approach.

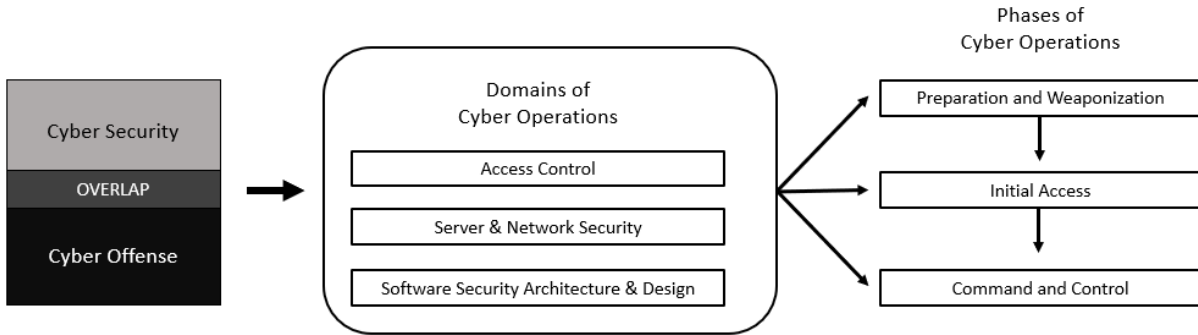


Figure 3.4: COSES Conceptual Model

In accordance with guidance from [97], items were also developed to reflect an increase in difficulty across these phases of operation. Initial item generation resulted in twenty-five questions covering each of the above domains across the three phases. Both inductive (qualitative, informal individual interviews) and deductive (literature review and content analysis) approaches to item generation were applied as recommended by [108]. Following question generation, face, and content validity were confirmed via expert evaluation provided by five anonymous Ph.D. contributors (see Appendix C for the data collection tool

used), meeting the appropriate threshold recommended by [108]. Experts were fielded from Dakota State University faculty across the Beacom College of Computer and Cyber Sciences, and their responses were received via the aforementioned collection tool at jotform.com. Quantitative feedback was received in a five-point Likert format assessing the applicability and clarity of each item in addition to qualitative feedback on question quality, brevity, and applicability. Statistical analysis was conducted to ensure inter-rater reliability across the quantitative data. Inter-rater reliability is a determination of the consistency of grading across two or more evaluators. Krippendorff's Alpha provides a robust assessment of rater response correlation with a great deal of flexibility in selecting measurement types, mitigating missing data and smaller data sets [113] [114]. To conduct Krippendorff's Alpha within SPSS, a macro developed by Hayes presented in [114] distributed on his website [115] was uploaded into SPSS. Through the analysis of expert feedback, questions were updated, resulting in a reduction to twenty questions in total. A detailed description of statistical analysis and results follows in Chapter Four.

3.2.1.2 Phase II: Scale Development

The second phase of survey creation consisted of data collection and analysis by exercising the developed scale. A sample collection of approximately ten respondents per survey question, or 200 in total, is desired for statistical analysis [108]. Survey participants were solicited through online groups and email and filtered to ensure respondents were experienced in cyber operations. Responses were obtained anonymously via jotform.com (Appendix D is the final version of the COSES). A total of 227 respondents provided sufficient answers for analysis and represented a broad swath of cyber professionals across various age groups, education, certification, and employment fields.

Despite domain determination a priori in developing the COSES, factor analysis provided an opportunity to statistically analyze and confirm the survey questions correlate with one another, aren't redundant, and accurately represent the desired trait [116]. Fac-

tor analysis allows the identification of unobservable or latent factors by examining the correlated variance of items within the developed scale. While a variety of approaches to conducting factor analysis exist across a wide range of research fields, path analysis followed by oblique rotation was applied to the COSES in light of the potential correlation between the cybersecurity and cyber offensive domains with the intent to complete an orthogonal rotation if required [117]. Figure 3.5 provides the proposed theoretical construct of the three cyber operations domains (access control, server & network security, and software security, architecture, & design) with respect to cybersecurity separate from cyber offense, resulting in the homogenous unidimensional latent factor of cyber operation performance for each cyber domain.

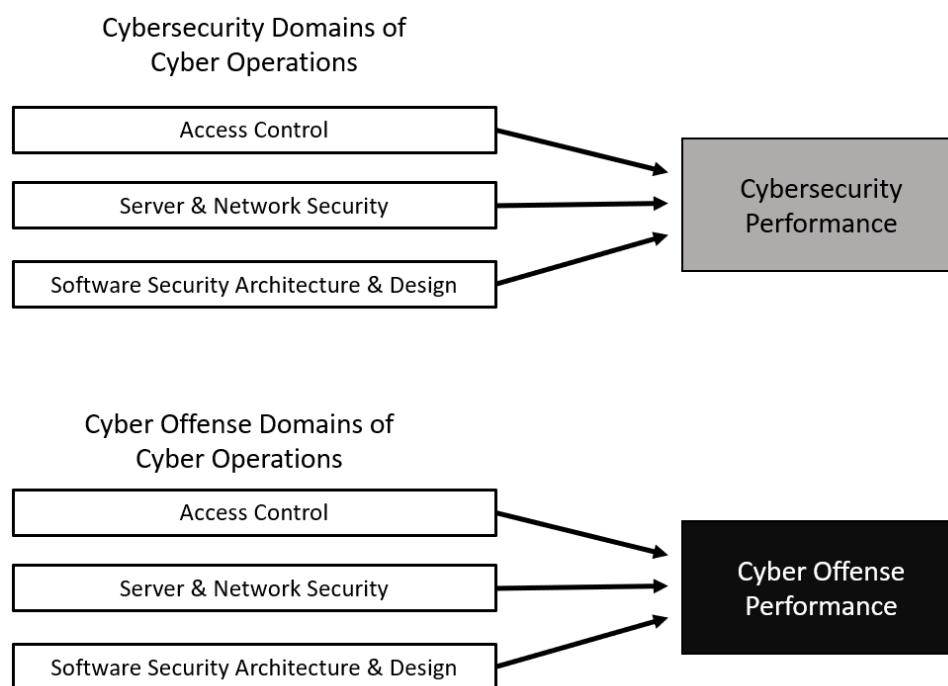


Figure 3.5: Theoretical Cyber Performance Latent Factor

Principle component analysis (PCA) is an alternative to factor analysis in which the goal is specifically aimed at item reduction, vice factor interpretation and overall latent factor analysis [117]. Despite this, PCA is commonly used to interpret the number of latent factors necessary to conduct effective factor analysis. Two approaches are applied

to testing the theoretical constructs of cyber performance presented in figure 3.5, the kaiser rule and evaluation of a PCA scree plot of the eigenvalues [117].

In all forms of measurement, an element of error is presumed. [117] presented measurement error as either systematic (consistent error) or random error (inconsistent error). Both types are produced beyond the development of the scale to include the time of evaluation, interpretation of scale items, and racial bias. To evaluate scale error, two primary models are available, Classic Test Theory (CTT) and Item Response Theory (IRT). Classic test theory, also known as true score theory, was developed throughout the early twentieth century, representing the foundational statistics used to measure, validate, and interpret scale measurements [118]. Based on the sample and item size, CTT was selected as the preferred method of analysis [119]. Utilizing CTT techniques, inter-item and item-total correlations were tested, the results of which are presented in Chapter Four.

3.2.1.3 Phase III: Scale Evaluation

The final phase of scale development is to confirm the dimensionality, reliability, and validity of the assessment. Dimensionality was verified via confirmatory factor analysis. Measurement invariance testing confirmed the factor and dimension as consistent across different samples. Internal reliability and consistency using Cronbach's α remained consistent across phase two and phase three of development. Testing criterion validity, the final step within phase III was achieved via multivariate regression. Results are presented in Chapter Four.

3.2.2 Friendly and Adversary Skills and Expertise

Skills and expertise represent an important element in an operation's success or failure [120]. As previously mentioned, many organizations have internal tools dedicated to evaluating skill and expertise that provide a more specific approach than can be provided here. For context, the following items are those relevant to the agent-based model and can

provide a baseline for organizations wishing to develop or expand current cyber assessment tools.

Cybersecurity activities:

- Maintain threat awareness
- Install software updates
- Detect traffic anomalies
- Identify software deviations
- Identify network intrusion
- Recover impaired services
- Digital forensics

Cyber offense activities:

- Target selection
- Adversary OPSEC
- Infiltrate a network
- Defense evasion
- Identify software vulnerabilities
- Exploit software vulnerabilities undetected
- Command and control
- Lateral Movement

These skills represent the agent's skill level throughout the course of the model's simulations. This input is provided via a ten-point slider by the user to indicate the friendly and adversary cyber operator skill levels.

3.2.3 Network Characteristics

There exists a diverse collection of strategies to quantify cybersecurity network characteristics ranging from MITRE’s extensive cyber resiliency [121] to network-type specific tools such as an assessment used for railway signals communications [122]. The goal of the cyber operations performance framework is practical use alongside an applicable strategic approach such as Lockheed Martin’s Cyber Kill Chain [123]. To leverage a simple, comprehensive, and reliable set of metrics to meet this challenge, the standards established within the NCCIC (National Cybersecurity and Communications Integration Center) Cyber Security Evaluation Tool (CSET) [124] are recommended to evaluate the target network and organization.

Alternative options for assessments include, but are not limited to:

- DHS Catalog of Control Systems Security
- NERC Critical Infrastructure Protection (CIP) Standards 002-009
- NIST Special Publication 800-82, Guide to Industrial Control Systems Security
- NIST Special Publication 800-53, Recommended Security Controls for Federal Information Systems
- NIST Cybersecurity Framework
- Committee on National Security Systems Instruction (CNSSI) 1253
- NISTIR 7628 Guidelines for Smart Grid Cyber Security

For users simulating defensive performance, recent research by Zavala et al. provides automation guidance for the CSET, simplifying data entry and analysis via PowerShell [125]. Using this or any security assessment tool that generates a network security strength score between 0 to 100, the user can provide the computational model input for the

target network strength. Each step taken across the computational model will then impact the network’s security based on actions taken by the attacking and defending agents.

3.2.4 Alternative Cyber Performance Metrics

There are undoubtedly additional factors that influence operational success, including preparation and execution, timeline requirements, additional intelligence and resource availability, and various go/no-go criteria. The key to an effective agent-based model is using the fewest factors to demonstrate reliable results even when dealing with highly complex systems [10]. Chapter Five discusses recommendations for future research that can expand on the computational model, including additional cyber characteristics and measurements.

3.3 Computational Modeling Methodology & Design

While the self-efficacy assessment is a critical element to account for the behavioral characteristics of cyber operators, the computational model is the defining feature for answering the research question: can we provide real-world assessments and predictions of performance for cyber operations? Because computational modeling is used across nearly every field of scientific research as both a tool for experimental exploration as well as a means to analyze complex adaptive systems, there is no silver-bullet choice for selecting a research methodology.

In part, this research works to capture the results of behavioral data. As noted by Wilson and Collins, the selection of qualitative or quantitative methodology for a computational model of behavioral data is dependent on the researcher’s goals [126]. A wide range of goals across research interests has resulted in both significant diversity and challenges in developing reliable computational models [127]. Alternatively, the research also applies the technical standards characteristic of cyberspace to a tool designed to solve

a problem. Design science is predicated on developing and evaluating an artifact within the problem context, be it social, technical, organizational, or an applicable combination [128]. While there are formal frameworks for mixed methods methodology, at the heart of each is the combination of qualitative and quantitative research - for which there is a great deal of value in analyzing cyber performance [129]. Just as there are numerous legitimate approaches to complexity and cyber research, so too are the options for selecting a research methodology for solving this research problem. While each approach may be effectively applied in developing a computational model, design science provides an ideal methodology as the research problem is grounded in case research applying the developed framework and resulting tools to solve both theoretical and applied challenges.

Both design science and action research methodologies have significant use throughout computer science research, especially in efforts combining industry and academia [130]. The single-case mechanism experiment method was applied as a means to develop an artifact and test it by solving a bonafide problem in information and or organizational systems [131]. As is the standard with agent-based modeling, the researcher is directly involved in conducting experimentation, evaluating results, and iterating for improvements, often within a lab setting [128]. Engineering cycles of iterations test initial assumptions, which are developed into practical assumptions and ultimately into validated findings (Figure 3.6). With this design methodology in mind, the development of the Cyber Operations Performance Framework computational model evolved through multiple engineering cycles using an investigation, development, testing, redesign, and repeat approach [128]. As results were validated or challenges identified, the framework and associated tools improved, as is the goal with design science methodology.

3.3.1 Agent-Based Model Introduction

Agent-based models are developed through an iterative process of conceptualization, model development, simulation runs, and analysis of findings - often followed by revisiting



Figure 3.6: Design Science Methodology

initial assumptions and error checking both modeling programming and execution validation. [132] emphasized four primary approaches for developing an empirical agent-based model: (1) via statistical distribution from empirical data, (2) direct observation and comparison, (3) laboratory testing, (4) case study analysis. Due to the common challenge of obtaining existing or observational data for statistical analysis of cyber attacks or laboratory facilities for testing, this research relied on case study data to develop the practical and numerical elements of the agent-based model. While a variety of case studies were reviewed and elements applied, two examples of note include the Doncaster Teslacrypt ransomware attack in 2018 [133] and the DarkSide ransomware attack on Colonial Pipeline in 2021 [134].

To conceptualize the model for real-world applicability, a ransomware attack use case by a cybercriminal hacking group, DarkSide, will be applied in the development and analysis of the model’s functions. DarkSide is a cybercriminal group that conducts ransomware cyber attacks on targets throughout the world. The group, or “affiliates”, leverage an attack framework that includes a ransomware as a service (RaaS) to execute exploitation. While the exact number of attackers or specific skills and training is unknown, their operations indicate the group is well organized, skilled, and effective. Once access is gained, the group will remain on the network for only a few days before encrypting assets and demanding ransomware. Quick action, infected asset separation, and preplanned recovery of affected assets has resulted in successfully avoiding complete network loss from

ransomware.

The primary components of an agent-based model are the agents within the model and the environment. Agents are programmed with a set of characteristics and initial actions. Through the result of inter-agent influence and agent-environment influence, agents take actions throughout the simulation that change and evolve. While there are a variety of approaches to defining human behavior within an agent-based model, the Cyber Performance Framework adopts the PECS approach. PECS is an acronym that stands for (1) physical conditions, (2) emotional state, (3) cognitive capabilities, and (4) social status [135]. The physical state is defined by network strength, emotional state by self-efficacy, and cognitive capabilities through the skills and expertise factor. Social status can play an important role in inter-agent communication and evolution, but as offensive and cybersecurity agents are of the same social status, implementation of a social status characteristic was not applicable. Agents do, however, interact with one another on a social level when requesting assistance from one another in their efforts.

3.3.2 Agent Types

There are two agent types within the Cyber Operations Performance Framework: Operators and Nodes. Operators are defined as friendly or adversary forces. The choice of friendly forces attacking or defending and the number of defender and attacker agents is a user-defined input. Cybersecurity agents work to identify, patch, and protect organizational computers and network systems. Their ability to do so is influenced by their confidence (self-efficacy), skills and expertise regarding cyber defense activities, and the quality of the organization’s network (Table 3.1).

Table 3.1: Degree of Influence on Agent Performance Across Model Characteristics

	Self-Efficacy	Skills & Abilities	Network Characteristics
Cyber Forces	15%	65%	25%

Cyber offensive agents work to identify, create, and abuse target computer and network systems. Their ability to do so is influenced by their self-efficacy, skills, and expertise regarding cyber-attack activities and the quality of the target’s network. Cybersecurity agents monitor, update, repair, and recover the network and likewise are influenced by the same metrics. Adjusting these factors for organizational implementation and real-world fit is discussed in Chapter Five.

3.3.3 Environment Types

Visually, the computational model is presented on a 35 x 35 torus grid (edges wrap) that represents the target network. In addition to cybersecurity and offensive agents, users can observe network nodes that represent the attack surface for exploiting and securing the network. These nodes are the fundamental environmental type upon which all cyber operators interact. The number of nodes is a user-defined input that plays a significant role in the overall results of the simulation. The nodes are defined by varying degrees of security exposure/status, indicated in Table 3.2. A Network node’s security state can be (1) offline, (2) zero-day secure, (3) secure, (4) exposed, (5) infiltrated, and (6) exploited. The shifts between these states is explained in greater detail below.

Table 3.2: Network Node Status

Network Node Status					
Offline	Zero-Day Secure	Secure	Exposed	Infiltrated	Exploited

to facilitate the implied impact (a high network strength positively impacts defenders, negatively impacts attackers; likewise, a lower network strength positively impacts attackers and negatively impacts defenders, the following equations were applied to determine the impact of network strength (Equation 3.3)

$$\begin{aligned}
Defender &= Nts * .02 \\
Attacker &= abs(Nts - 10) * .02
\end{aligned}
\tag{3.3}$$

A network security value of 50 (the default) benefits neither attacker nor defender.

3.3.4 Simulation Start-up

As previously highlighted, there are a number of user-defined inputs that allow the model to reflect real-world operations that can be modified prior to simulation start. The following list represents the initial user settings, minimum, maximum, and default values:

- Number of Network Access Nodes: Minimum: 1, Maximum: 100, Default: 15
- Network Initial Security Strength: Minimum: 1, Maximum: 100, Default: 50
- Friendly Forces Mission: "Defend" or "Attack", Default: "Defend"
- Number of Cyber Friendly Operators: Minimum: 1, Maximum: 100, Default: 2
- Friendly Forces Skills, Minimum: 1, Maximum: 10, Default: 7
- Friendly Forces Team-Efficacy: 1, Maximum: 10, Default: 7
- Number of Adversary Operators: Minimum: 1, Maximum: 100, Default: 4
- Adversary Skills, Minimum: 1, Maximum: 10, Default: 7
- Dollar Cost of an Outage per Day: Minimum: \$1,000, Maximum: \$200,000, Default: \$100,000
- Dollar Cost of a Defender per Year: Minimum: \$30,000, Maximum: \$120,000, Default: \$85,000

To this point, each of these settings has been introduced, with the exception of the dollar cost settings, used to facilitate user assessment of value regarding adding additional operators versus a security breach that takes elements of the network offline. Additionally, the computational model provides a graphical line chart that depicts the overall state of the network based on the status of each node. The computational model at simulation start-up is presented in Figure 3.7. The position of the attackers and nodes is both random and arbitrary. The defenders are always positioned along the sides of the display; their location is also arbitrary.

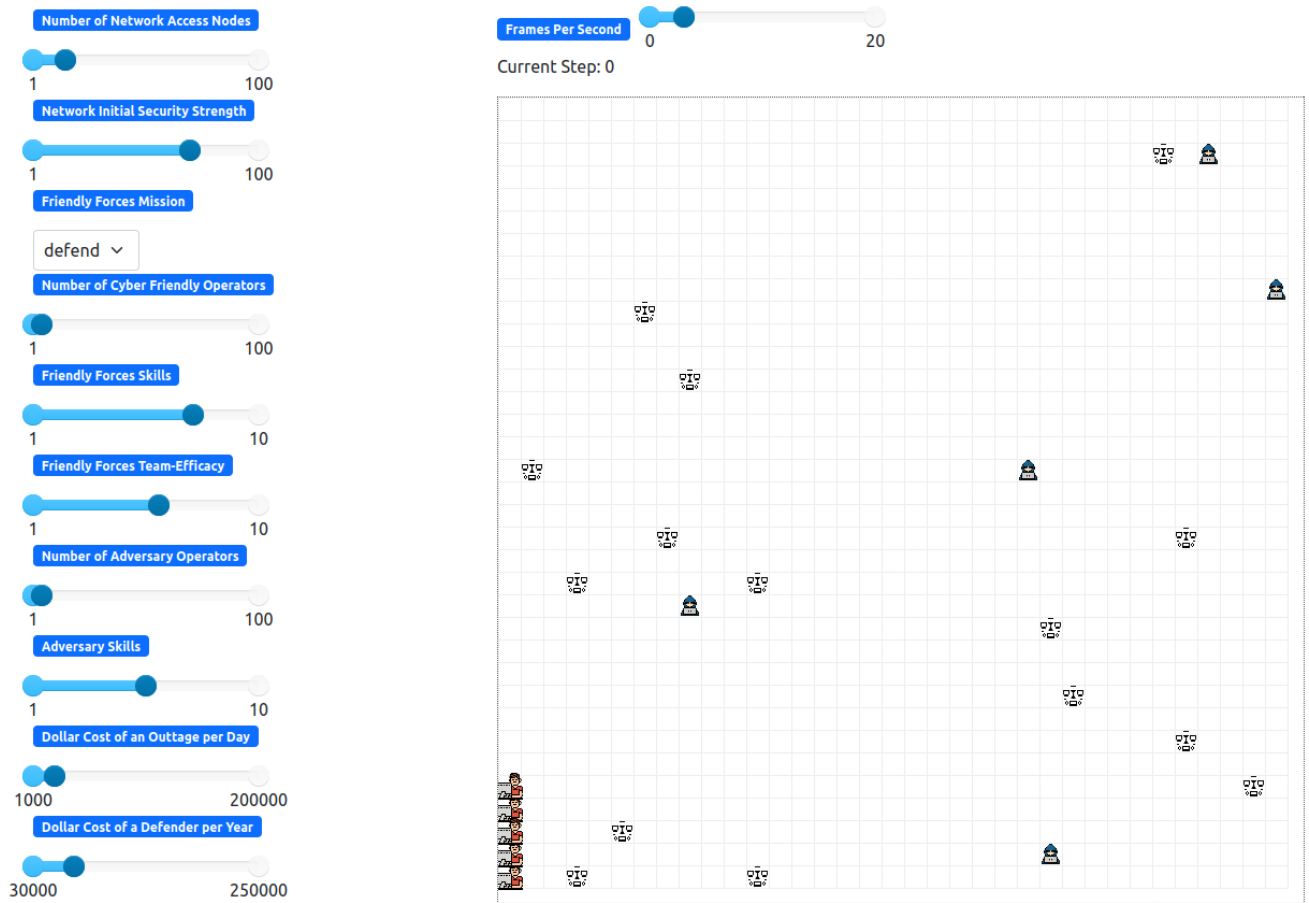


Figure 3.7: Cyber Performance Computational Model at Start-up

3.3.5 Model Execution

Agent-based models run simulations as a matter of “steps” or “ticks”, each representing a turn (or a day) in which all agents take an action. The user can initiate the model to run automatically or one step at a time and can set the speed to run between 1 and 20 frames per second. The order of the agent’s turn is random for each step. The actions the agents take in the computational model are dependent on the phase of cyber operation they are currently in. As noted earlier in this chapter, agents within the computational model operate in three distinct phases: preparation and weaponization, initial access, and command and control. Figure 3.8 illustrates these phases characterized by agent type and their simulated actions.

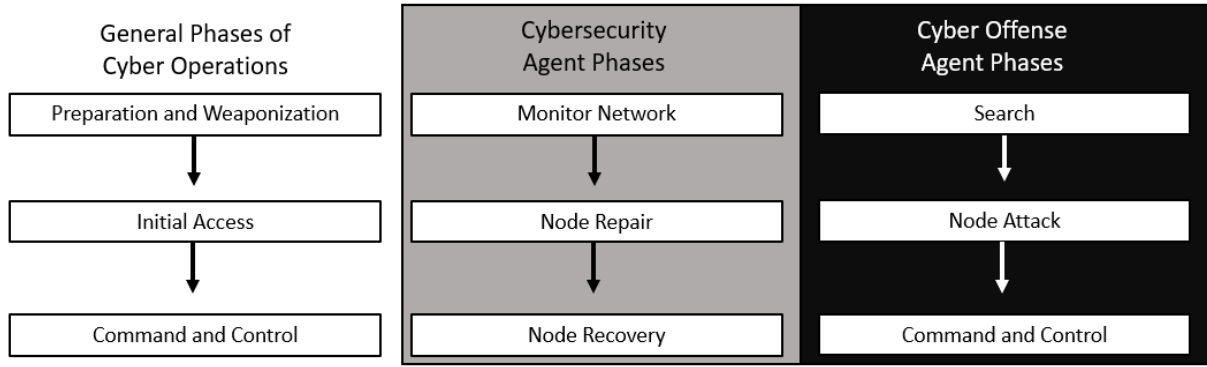


Figure 3.8: Computational Model Agent Phases

3.3.6 Defender’s Turn

Upon model start-up, all defenders enter into the monitor phase, simulating updating, patching, and network traffic observation. Each defender monitors an assigned set of nodes, evenly distributed across all defenders based on the number of defenders and the total number of nodes to monitor. Only if there are more defenders than nodes will more than one defender monitor the same node. Defenders monitor network nodes based on the following function (Equation 3.4):

$$Dc = Sk * .65 + Tse * .15 + Nst * .02 \quad (3.4)$$

Where Dc is the operator's defense capability, applied as the maximum number of nodes a defender is capable of monitoring per step based on skill (Sk), self-efficacy (Tse) and network strength (Nst) with the degree of influence introduced in Table 3.1. The resultant factor is a value between .82 and 10. If the defender has more nodes assigned then is able to monitor in a single day, then the operator prioritizes nodes based on node status, focusing first on the node with the lowest value. A patch is applied to every node the operator is able to monitor based on Equation 3.5:

$$Ns = Ns + Dc/7 \quad (3.5)$$

Where Ns is node status and Dc remains the defender's overall ability. The divisor of seven results in an overall change in node status between 1% and 14%, directly dependent on the Dc factors of skill, self-efficacy, and network strength. If, after the update, the defender agent observes the node in a non-secure status, the defender will add the node to the repair list and move to repair it next turn. If the cybersecurity operator is confident enough (self-efficacy is greater than a random integer between one and ten), they will inform another cybersecurity agent to help assist in repairing the node.

A defender in the repair phase will focus on a single node per step, conducting a repair based on Equation 3.6:

$$Ns = Ns + Dc/3 \quad (3.6)$$

With a divisor of three, the node status, Ns , will improve by a rate up to 33%, reflecting an increase from the monitor phase based on the dedicated focus of the agent. If an agent in the repair phase observes a node that is infiltrated or exploited following the repair,

they will notify all defenders currently in the monitor phase to assist in recovery.

Recovery is the third and final phase available to defenders. A defender in the recovery phase will also focus on a single node. The defender will recover a node that is infiltrated or exploited by taking it offline. The defender will attempt to identify any of the attackers by searching the network location and identifying if an attacker is present. If found, the attacker is placed in an offline status for ten to twenty steps, representing the disruption of being identified by local or federal law enforcement. The defender that recovered the node and the node are taken offline for a random number of steps between four and ten, representing the time required to repair the attacked node.

3.3.7 Attacker's Turn

At model start-up, all offensive agents begin in the search phase. During the search phase, an attacker is seeking to identify a node to attack. Searching is a probability-based function presented in Equation 3.7:

$$\begin{aligned} Ac &= Sk * .65 + Tse * .15 + Nst * .02 \\ Srch &= (1 + Nc/30) * Ac * .015 \end{aligned} \tag{3.7}$$

Ac is the attacker's capability, an attacking agent's attack potential based on Table 3.1. Sk Tse and Nst are skill, team self-efficacy, and network strength, respectively. $Srch$ is the function of Nc , the node count, and Ac . As the node count increases, the probability of finding a node increases as well. The function was conceived by analyzing the likelihood that a skilled non-expert (seven of ten) attacker could find a computer access point to exploit in a network of 15 potential access points out of 1,225 possible attack points (The grid's total attack surface: 35 x 35). To begin, the probability of selecting one of 15 of 1,225 possibilities is 1.22%. As the attack search is expected to be both intentional and with (varying degrees of) skill, the resulting probability needed to scale accordingly. Based

on the above equation, with an Attacker Sk of 7, Tse of 7, and Nst of 50, (resulting Ac 6.6), the search probability $Srch$ is .1485 resulting in a 14.85% chance for success. Using the same Ac of 6.6, the probability ranges from 10.23% (a single node) to 33% (100 nodes). An expert, where Sk is ten, Se is ten, and Nst is 50) has a probability search range of 13.95% through 58.5%.

While a successful search from one agent does not directly impact the search of another, if an agent does not detect an active network node, a confidence check is conducted (self-efficacy greater than a random integer between one and ten) to obtain a known node location from another random attacker (who may or may not have a known node location). When a node is identified, it is added to an attack list, assigned a random float status between five and seven representing the degree of degradation, and the attacker moves into the attack phase starting on the following step.

During the attack phase, attacker agents move to the target node on the grid and degrade computer nodes in an attempt to exploit the network based on Equation 3.8:

$$Ns = Ns - Ac/7 \quad (3.8)$$

Here, Ns is node state, and Ac is attack capability. Similar to the above defender patching equation, this equation represents a steady-state attempt to uncover a means to exploit the node. As the node state Ns is reduced through multiple iterations (one per step), it moves through states of exploitation. Revisiting the previous node status identifiers, the following table includes numerical values to facilitate each node's movement from secure to exploited (Table 3.3).

Table 3.3: Network Node Status with Values

Network Node Status						
Title	Offline	Zero-Day Secure	Secure	Exposed	Infiltrated	Exploited
Value	N/A	11 <= Status > 9	9 <= Status > 7	7 <= Status > 4	4 <= Status > 2.5	2.5 <= Status > 0

An expert attacker with a target network strength of 50 will have a Ac of 9, resulting in a per/step node degradation of 1.29. It's important to bear in mind that one or two defenders may work to repair the node, unaware of the attacker's attempts to exploit it. If the node's security value falls below four, it is infiltrated, and the attacker has achieved access to the network. At this stage, if a defender agent monitors the node, the defender will move directly to the recovery phase (as noted above). From here, the attacker is required to reduce the node to the exploited status (below 2.5) to achieve exploitation. Once achieved, the attacker moves into the final phase, command and control.

Once in the command and control phase, the attacker agent leverages network access to laterally move throughout the organization's network, further degrading each network node. Simulating the exploitation of an entire network to achieve a ransomware attack (the entire network is compromised to prevent access until a ransom is paid). Based on the attacker's current foothold in the network, the search function to move laterally is improved for the attacker agent in command and control (Equation 3.9):

$$\begin{aligned} Ac &= Sk * .65 + Tse * .15 + Ns * .02 \\ Srch &= (2.5 + Nc/30) * Ac * .015 \end{aligned} \tag{3.9}$$

Attacks are also improved while in the C2 phase (equation 3.10):

$$Ns = Ns - Ac/7 \tag{3.10}$$

Note the 2.5 factor now used versus the previously applied factor of 1 for search and the improved attack with a divisor of 3. This mirrors the advantage of lateral movement through a network vice an external attack. The previously noted $Srch$ for an expert (Ac is 9 where Sk is ten, Se is ten, and an Ns of 50, 15 nodes) has a probability search range of 20%. While in command and control a lateral search under the same constraints, the

search is 40.5%.

Recognizing a nearly infinite number of variables influence the probability of the search, attack, repair, and recovery of a network, these numbers were defined and refined through many runs to determine simulated real-world operations. Chapter Five presents challenges and opportunities for future work in further developing the Cyber Operations Performance Framework and implementing it for use in an organizational environment.

3.3.8 Simulation End

When running a single simulation, the model will continue to run until all network nodes are either offline or exploited. This can result in an indefinite run when defenders are able to ensure, at a minimum, one node is not exploited or offline. Once all network nodes are either offline (in an outage status being recovered by defenders) or exploited (organizational access to the node is limited due to node corruption), the network is considered completely compromised, and the simulation ends. In batch runs, a back-end number of multiple simultaneous runs of the model is conducted across a fixed or range of parameter inputs. Through testing, it's been determined that very likely that runs will run indefinitely if reaching 1000 steps. Computational power is another important consideration when determining the maximum steps and number of iterations in batch runs.

3.3.9 Model Sensitivity Analysis

With any complex system, modeling and simulation can demonstrate emergent properties not evident or even counter-intuitive at the agent or micro level. Computational models, while perhaps a departure from standard statistical analysis, benefit from statistical analysis of the model itself for model assessment. Due to the nature of complex system modeling, a direct quantitative analysis within the "black box" of the model and its formulas is often misleading, but sensitivity analysis can provide model developers a means

to confirm and analyze emergent properties while quantifying parameter variability, and impact [136] [137]. As defined by [137], sensitivity analysis can be achieved through a six-step process: (1) defining the output of interest, (2) goal confirmation, (3) selecting parameters for analysis, (4) defining the sensitive method or design, (5) assigning values, and (6) visualizing and analyzing the results. As previously defined, the outputs of interest include the dependent variables: vulnerability rate and capability rate. Goals for sensitivity analysis include model calibration, factor impact, direction of change, and an analysis of emergent properties. Each user input serves as an independent variable for model analysis. Two separate designs are tested against the cyber computational performance model, the one-factor-at-a-time (OFAT) or local sensitivity and global sensitivity analysis (GSA). The results of the OFAT demonstrated the specific parameters to be analyzed through GSA based on decomposition [136]. Inputs selected represent the available range of user-available options while maintaining consideration of computational costs and limitations of the virtual machine. Results are discussed within Chapter Four and presented in Appendix E.

3.3.10 Code Development

Various agent-based modeling software platforms and frameworks have evolved over the last two decades, including the notable off-the-shelf standalone and educational standard Net-Logo and commercial software Anylogic. While these provide a simple interface and a relatively small learning curve, they lack the customization necessary for advanced solutions and implementations. To develop the programming for the computational model and implement the aforementioned elements, the Cyber Operations Performance Framework was developed leveraging the open-source software Mesa [138], an agent-based model development software built on a Python 3 framework (<https://github.com/projectmesa/>). A VMWare Ubuntu (22.04.1) distribution hosted development, and Python 3.10.6 via the Mesa framework was used to develop the Cyber Operations Performance Framework

computational model.

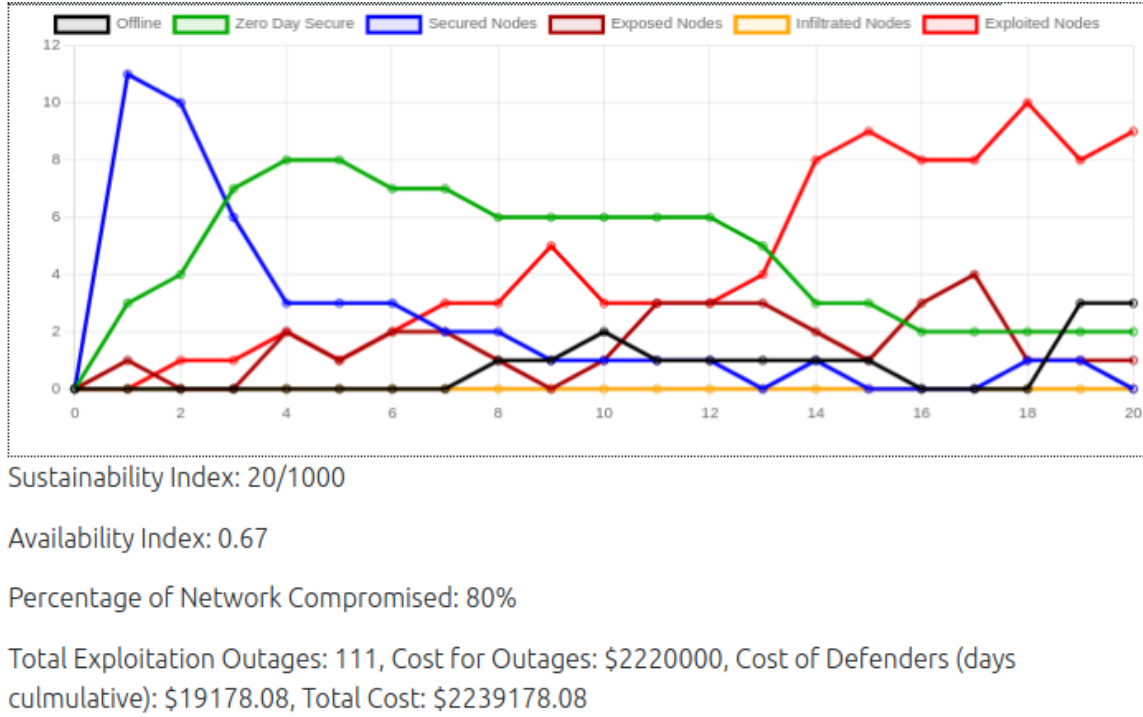


Figure 3.9: Cyber Performance Dashboard

3.3.11 Dashboard Development

In addition to the computational model display is the dashboard and text results, presented in Figure 3.9 at step twenty of a simulation. Through the dashboard, users observe the network's status on a step-by-step basis with feedback regarding the previously defined sustainability and availability index, percentage of the network currently compromised, total exploitation and outages, outage total cost (based on user input), the total daily cost of defenders (based on user input), a total of both factors on a step by step (or day by day) basis. The node status line chart depicts in real time the number of nodes and their associated security exposure. The intent of the dashboard is a visual representation of node and network status that users can evaluate to assess model implications with financials to facilitate calculating return on investment for cybersecurity defenders (or financial impact from cyber offensive operations).

Chapter 4

Research Results

4.1 Introduction

As noted in Chapter Three, the Cyber Operations Performance Framework consists of three major elements, the Cyber Operations Self-Efficacy Scales (COSES), the computational model, and the performance dashboard. This chapter outlines the results of the development, research, and testing of each of these research efforts in applicable sections.

4.2 COSES Research Results

The results of the Cyber Operations Self-Efficacy Scales are presented here based on the phases of development introduced in Chapter Three: Phase I: Initial Development, Phase II: Scale Development, and Phase III: Scale Evaluation.

4.2.1 Phase I: COSES Initial Development

The first steps in scale development are to define the desired domain of study and specify the dimensions of the domain to measure. Following an extensive literature review, initial item development resulted in twenty-five items across three primary domains (1) access control, (2) server and network security, and (3) software architecture and design. To ensure practicality in conducting the COSES, the dimensions of these domains included

both cybersecurity and cyber offensive techniques in a single scale. The scale is presented across three phases of a cyber operation: (1) preparation & weaponization, (2) intrusion, and (3) active breach. Limitations produced by implementing this scope are discussed in Chapter Five. Based on the intent and basis for the Cyber Operations Performance Framework, item domains for the COSES were developed a priori, directly driving the dimensions of the domains within the scale [108].

Experts were fielded from Dakota State University faculty across the Beacom College of Computer and Cyber Sciences to conduct face and content validity, focusing on content applicability and completeness, item clarity, brevity, and structure. Five experts provided data, completely answering each of the questions provided. Responses were collected anonymously via jotform.com (see Appendix C to review the Expert Evaluation & Content Validity Tool). Data collected via Likert responses were treated as ordinal, quantitative data for statistical analysis. Verbal and written feedback was reviewed and compared for consistency and treated as qualitative data. To confirm inter-rater reliability, Krippendorff's α bootstrap was selected as the most flexible appropriate measure for two or more raters. Table 4.1 indicates the obtained results across 10,000 samples of a .8944 Krippendorff's α , indicating a high degree of inter-rater reliability across rater responses.

Table 4.1: Expert Analysis Feedback Krippendorff's α Results
Krippendorff's Alpha Reliability Estimate

	Alpha	LL95%CI	UL95%CI	Units	Observrs	Pairs
Ordinal	.8944	.8486	.9353	25.0000	5.0000	250.0000
Probability (q) of failure to achieve an alpha of at least alphamin:						
alphamin	q					
.9000	.5932					
.8000	.0000					
.7000	.0000					
.6700	.0000					
.6000	.0000					
.5000	.0000					
Number of bootstrap samples:						
	10000					

Expert feedback regarding item clarity and applicability promoted reducing the scale from twenty-five to twenty questions (Appendix D contains the resulting twenty-item version distributed for analysis).

4.2.2 Phase II: COSES Scale Development

Following the analysis of expert feedback, the second phase of scale development was conducted, including survey administration and exploratory factor analysis (EFA).

4.2.2.1 Survey Administration

A sample collection of approximately ten respondents per survey question is desired for statistical analysis [108]. Responses to survey requests were obtained directly from jotform.com (Appendix D is the final draft of the COSES scales). To confirm the appropriate sampling of the target population, a demographics survey preceded the COSES that included questions to determine appropriate cyber familiarisation in work experience, education, and certifications. Only results indicating the respondent had work or educational experience in cyber operations were applied. A random subsample of N=55 provided initial analysis and a basis for separating the COSES cybersecurity and cyber offensive surveys. A total of 227 respondents provided usable data in the updated version of the COSES. Of that, 86% of respondents were male, the mean age was 34 years old, and the mean applicable work experience was five years.

4.2.2.2 Item Reduction & Factor Analysis

Though the Pearson correlation represents the standard in coefficient correlation, analysis of ordinal data, such as that produced by Likert scales, requires an alternative approach. As recommended by Watkins, exploratory factor analysis was conducted based on polychoric correlations [139]. Leveraging the SPSS Heterogenous Correlation extension (ver. 2.0), a high polychoric correlation was observed across items when cyberse-

curity items were separated from offensive items. Table 4.2 depicts the results of the inter-item correlation with a mean correlation of .782 for cybersecurity-based questions. Question-to-question correlation demonstrated strong unidimensional consistency across the cybersecurity portion of the COSES.

Table 4.2: Polychoric Correlation of Cybersecurity items

Variables	Statistics	PCS1	PCS2	PCS3	PCS5	PCS6	ICS1	ICS2	ICS3	ICS4	ABCS1	ABCS3
PCS1	Correlation	1.000	.799	.806	.898	.664	.881	.795	.711	.811	.934	.823
	Std. Error	.000	.057	.057	.036	.079	.042	.055	.084	.057	.022	.057
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
PCS2	Correlation	.799	1.000	.822	.756	.740	.656	.865	.875	.781	.826	.926
	Std. Error	.057	.000	.049	.073	.067	.081	.034	.039	.069	.054	.028
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
PCS3	Correlation	.806	.822	1.000	.897	.839	.634	.600	.829	.765	.826	.871
	Std. Error	.057	.049	.000	.036	.058	.088	.097	.054	.065	.057	.043
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
PCS5	Correlation	.898	.756	.897	1.000	.789	.787	.723	.865	.904	.931	.787
	Std. Error	.036	.073	.036	.000	.062	.059	.071	.041	.036	.022	.057
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
PCS6	Correlation	.664	.740	.839	.789	1.000	.618	.466	.751	.693	.624	.883
	Std. Error	.079	.067	.058	.062	.000	.088	.107	.066	.078	.089	.038
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
ICS1	Correlation	.881	.656	.634	.787	.618	1.000	.684	.590	.753	.763	.661
	Std. Error	.042	.081	.088	.059	.088	.000	.077	.095	.070	.067	.082
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
ICS2	Correlation	.795	.865	.600	.723	.466	.684	1.000	.831	.863	.848	.698
	Std. Error	.055	.034	.097	.071	.107	.077	.000	.049	.038	.048	.071
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
ICS3	Correlation	.711	.875	.829	.865	.751	.590	.831	1.000	.914	.842	.780
	Std. Error	.084	.039	.054	.041	.066	.095	.049	.000	.033	.049	.070
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
ICS4	Correlation	.811	.781	.765	.904	.693	.753	.863	.914	1.000	.843	.715
	Std. Error	.057	.069	.065	.036	.078	.070	.038	.033	.000	.057	.078
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
ABCS1	Correlation	.934	.826	.826	.931	.624	.763	.848	.842	.843	1.000	.776
	Std. Error	.022	.054	.057	.022	.089	.067	.048	.049	.057	.000	.065
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000
ABCS3	Correlation	.823	.926	.871	.787	.883	.661	.698	.780	.715	.776	1.000
	Std. Error	.057	.028	.043	.057	.038	.082	.071	.070	.078	.065	.000
	N	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000	176.000

Correlations computed by R Heter package

When analyzing the cyber offense items, it was clear numerous respondents with significant cybersecurity experience lacked confidence in completing cyber offensive tasks. This aligns with standard pedagogical approaches in cyber education in which students are expected to have a foundation in cybersecurity prior to mastering cyber offensive tasks. However, many who achieve cybersecurity mastery never learn cyber offensive techniques. The data collected represented the disparity of responses to cyber offensive self-efficacy and was unsuitable for coefficient correlation analysis. To address this, the final version

of the COSES was divided into two separate surveys; one focused on cybersecurity, the other on cyber offense. This not only shortened the time and effort required to administer the scale but also simplified analyzing results when the results are specific to the cyber domain of interest (cybersecurity versus cyber offense). Appendix D contains the final version of the COSES, with eleven cybersecurity items and ten cyber offense items, with one item reflected on both scales. To complete data analysis, responses indicating cyber offensive experience (N = 63) were isolated and examined independently. Table 4.3 depicts the results of the inter-item correlation with a mean correlation of .834 for cyber offense-based questions. Following scale separation, question-to-question correlation once again demonstrated strong unidimensional consistency, now with the cyber offense portion of the COSES.

Table 4.3: Polychoric Correlation of Cyber Offense items

Variables	Statistics	PCO3	PCO2	PCO5	ICO1	ICO2	ICO3	ICO4	ABCO1	ABCO2	ABCO4
PCO3	Correlation	1.000	.877	.932	.917	.869	.951	.875	.841	.874	.885
	Std. Error	.000	.081	.036	.043	.067	.028	.074	.083	.064	.063
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
PCO2	Correlation	.877	1.000	.861	.811	.839	.883	.916	.915	.856	.858
	Std. Error	.081	.000	.088	.079	.084	.058	.037	.042	.074	.064
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
PCO5	Correlation	.932	.861	1.000	.749	.782	.838	.847	.825	.885	.873
	Std. Error	.036	.088	.000	.099	.104	.083	.074	.085	.066	.068
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
ICO1	Correlation	.917	.811	.749	1.000	.760	.872	.869	.793	.782	.817
	Std. Error	.043	.079	.099	.000	.109	.069	.063	.066	.077	.082
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
ICO2	Correlation	.869	.839	.782	.760	1.000	.881	.770	.743	.673	.762
	Std. Error	.067	.084	.104	.109	.000	.070	.106	.113	.133	.111
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
ICO3	Correlation	.951	.883	.838	.872	.881	1.000	.791	.820	.813	.835
	Std. Error	.028	.058	.083	.069	.070	.000	.106	.094	.095	.091
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
ICO4	Correlation	.875	.916	.847	.869	.770	.791	1.000	.799	.896	.797
	Std. Error	.074	.037	.074	.063	.106	.106	.000	.102	.064	.103
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
ABCO1	Correlation	.841	.915	.825	.793	.743	.820	.799	1.000	.814	.802
	Std. Error	.083	.042	.085	.066	.113	.094	.102	.000	.094	.094
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
ABCO2	Correlation	.874	.856	.885	.782	.673	.813	.896	.814	1.000	.682
	Std. Error	.064	.074	.066	.077	.133	.095	.064	.094	.000	.132
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000
ABCO4	Correlation	.885	.858	.873	.817	.762	.835	.797	.802	.682	1.000
	Std. Error	.063	.064	.068	.082	.111	.091	.103	.094	.132	.000
	N	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000	63.000

Correlations computed by R Hecor package

4.2.3 Phase III: COSES Scale Evaluation

The final phase of scale development is to confirm the dimensionality, reliability, and validity of the assessment. Dimensionality was now verified via confirmatory factor analysis, applying principle component analysis (PCA) to the cybersecurity and cyber offense datasets via SPSS. Primary methods of identifying dimensions via PCA include confirmation of eigenvalues above 1.0, otherwise known as the Kaiser rule, and evaluations of a PCA scree plot of the eigenvalues [117]. Values on the scree plot that flatten out following a sharp drop are discounted as not significant in determining number of dimensions. Table 4.4 provides the cybersecurity total variance indications depicting eigenvalues.

Table 4.4: Cybersecurity PCA Total Variance

Component	Total	Initial Eigenvalues		Total	Extraction Sums of Squared Loadings		Rotation Sums of Squared Loadings ^a
		% of Variance	Cumulative %		% of Variance	Cumulative %	
1	6.740	61.273	61.273	6.740	61.273	61.273	6.044
2	1.000	9.092	70.365	1.000	9.092	70.365	5.133
3	.830	7.541	77.906				
4	.707	6.427	84.333				
5	.530	4.821	89.154				
6	.347	3.155	92.308				
7	.287	2.610	94.918				
8	.214	1.948	96.866				
9	.143	1.301	98.167				
10	.108	.985	99.152				
11	.093	.848	100.000				

Extraction Method: Principal Component Analysis.

a. When components are correlated, sums of squared loadings cannot be added to obtain a total variance.

While the Eigenvalue above 1.0 approach suggests two dimensions, the second dimension is not significant in its distinction from the others as demonstrated when analyzing the associated scree plot (Figure 4.1). The results indicated a single dimension of cybersecurity self-efficacy across the three primary domains: access control, server & network security, and software security architecture & design. The same analysis was conducted across the cyber offense dataset.

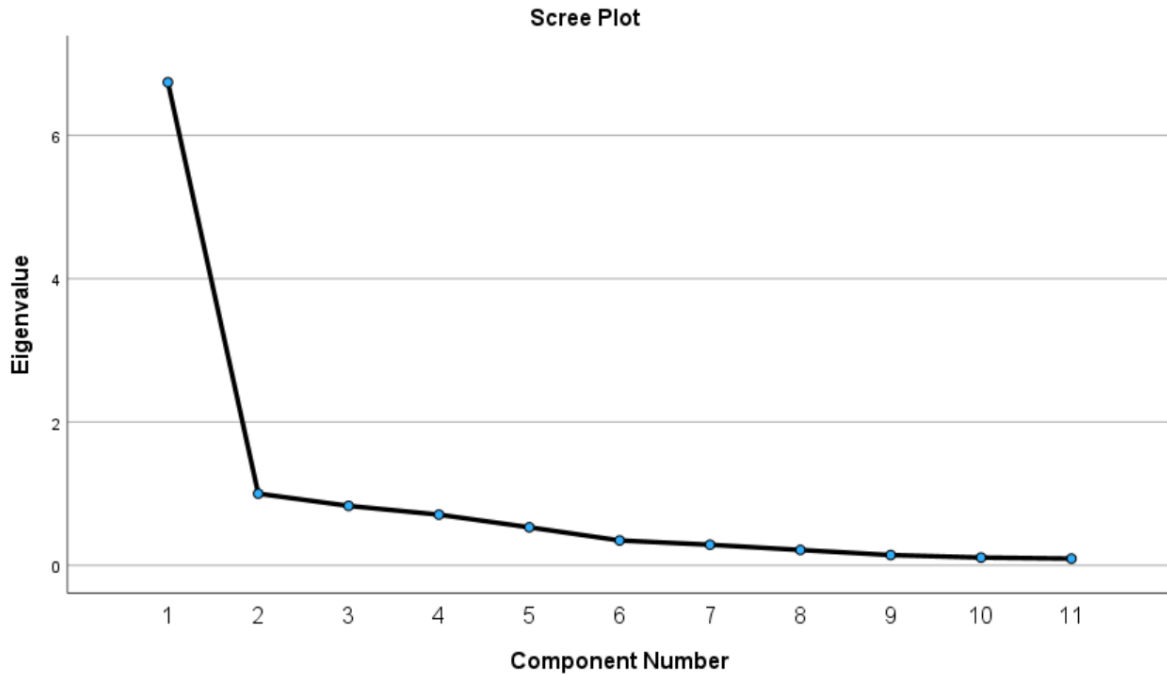


Figure 4.1: Cybersecurity PCA Scree Plot

There are numerous standards for evaluating internal scale reliability, the most common for psychometric scales being the Cronbach's α , a measure of the scale's item's average intercorrelation. A coefficient above 70 is the acceptable standard [140]. The COSES achieved a Cronbach's α of .93 for cybersecurity and .94 for cyber offense when the scales were adjusted and reduced to the final twenty items, leveraging IBM SPSS for analysis. Expert feedback indicated the remaining items achieved satisfactory face and content validity. Measurement invariance testing confirmed the factor and dimension as consistent across different samples.

Testing concurrent criterion validity, the final step within phase III, was achieved via linear multi-variable regression to determine causal effects and predictive validity across assessment items. A weighted approach was assigned to the participant demographic data to create a criterion for assessment. Cyber security results are presented in Table 4.5 and cyber offensive in Table 4.6. Data sets for both cybersecurity and cyber offense were confirmed for model fit against heteroskedasticity (inconsistent variation from variables

across time) and normal data distribution - both applicable assumptions for this statistical model. Both scales indicated overall statistical significance in predicting the dependent criterion (weighted cyber experience) with strong overall predictive power.

The Cybersecurity mutli-variable regression model (Table 4.5) resulted in an overall statistically significant result and an adjusted R square of .874 or 87%. A few of the items within the cybersecurity scale failed to meet the statistical significance standard of .05, and notably, P-CS-5 was exceptionally high at .4. It's worth noting that failing to meet statistical significance for this test doesn't immediately suggest the question is inappropriate for cyber self-efficacy, simply that the results don't align with the criterion developed based on user provided demographics. The overall positive correlation with each item and overall statistical significance provides confirmation of the scale's criterion validity.

The Cyber offense mutli-variable regression model (Table 4.6) also resulted in an overall statistically significant finding with an overall adjusted R square of .867 or 87%. This scale had additional items outside the bounds of statistical significance, suggesting a more rigorous criterion or further review of item analysis conducted through continued research. Additionally, the AB-CO-4 item resulted in a very slight negative relationship with the criterion, further calling into question the relationship between the criterion (overall cyber experience) and cyber operations proficiency. Additional testing may reveal interesting results given the previous observations regarding the development of cyber offense operators and the overall pedagogy of cyber operations education. Nonetheless, the overall scale is consistent with the criterion and results in a strong positive predictive relationship confirming criterion validity.

Table 4.5: Cybersecurity Multi-Variable Regression - Criterion Validity

Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.939 ^a	.882	.874	.44275

a. Predictors: (Constant), I-CS-4, AB-CS-3, I-CS-1, AB-CS-1, P-CS-6, P-CS-3, I-CS-2, I-CS-3, P-CS-5, P-CS-1, P-CS-2

b. Dependent Variable: Criterion

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	240.710	11	21.883	111.631	<.001 ^b
	Residual	32.148	164	.196		
	Total	272.858	175			

a. Dependent Variable: Criterion

b. Predictors: (Constant), I-CS-4, AB-CS-3, I-CS-1, AB-CS-1, P-CS-6, P-CS-3, I-CS-2, I-CS-3, P-CS-5, P-CS-1, P-CS-2

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	-.047	.454		-.104	.917	-.943	.849
	AB-CS-1	.263	.143	.100	1.834	.068	-.020	.546
	AB-CS-3	.261	.137	.101	1.899	.059	-.010	.532
	P-CS-1	.249	.150	.098	1.662	.099	-.047	.546
	P-CS-2	.445	.156	.174	2.842	.005	.136	.754
	P-CS-3	.316	.134	.115	2.362	.019	.052	.580
	P-CS-5	.123	.146	.048	.845	.400	-.165	.412
	P-CS-6	.252	.118	.095	2.127	.035	.018	.486
	I-CS-1	.257	.097	.100	2.645	.009	.065	.449
	I-CS-2	.194	.138	.077	1.406	.162	-.079	.468
	I-CS-3	.459	.159	.164	2.882	.004	.145	.774
	I-CS-4	.338	.138	.123	2.444	.016	.065	.612

a. Dependent Variable: Criterion

4.3 Computational Model Research Results

The aim of this research effort was to develop a computational model able to predict cyber operational performance and facilitate theory testing. The use case for developing and analyzing the computational model is a cybercriminal organization leveraging ransomware (such as DarkSide). Based on the minimal public quantified data regarding such cybercriminal enterprises and operations, validation of the model fit to real-world cyber attacks is limited to a review of results and comparison to available qualitative data. The

Table 4.6: Cyber Offense Multi-Variable Regression - Criterion Validity

Model Summary^b

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	R Square Change	Change Statistics			
						F Change	df1	df2	Sig. F Change
1	.943 ^a	.888	.867	.47226	.888	41.423	10	52	<.001

a. Predictors: (Constant), AB-CO-4, AB-CO-2, I-CO-2, AB-CO-1, I-CO-4, I-CO-3, I-CO-1, P-CO-5, P-CO-2, P-CS-3/P-CO-3

b. Dependent Variable: Criterion

ANOVA^a

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	92.384	10	9.238	41.423	<.001 ^b
	Residual	11.597	52	.223		
	Total	103.982	62			

a. Dependent Variable: Criterion

b. Predictors: (Constant), AB-CO-4, AB-CO-2, I-CO-2, AB-CO-1, I-CO-4, I-CO-3, I-CO-1, P-CO-5, P-CO-2, P-CS-3/P-CO-3

Coefficients^a

Model		Unstandardized Coefficients		Standardized Coefficients Beta	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error				Lower Bound	Upper Bound
1	(Constant)	1.165	.699		1.667	.101	-.237	2.567
	P-CS-3/P-CO-3	.268	.345	.098	.775	.442	-.425	.961
	P-CO-2	.620	.286	.243	2.170	.035	.047	1.193
	P-CO-5	.275	.263	.103	1.045	.301	-.253	.803
	I-CO-1	.405	.227	.173	1.783	.080	-.051	.862
	I-CO-2	.287	.174	.111	1.644	.106	-.063	.636
	I-CO-3	.245	.212	.095	1.156	.253	-.180	.670
	I-CO-4	.170	.244	.065	.699	.488	-.319	.659
	AB-CO-1	.055	.227	.022	.244	.808	-.400	.510
	AB-CO-2	.582	.221	.220	2.633	.011	.138	1.025
	AB-CO-4	-.003	.212	-.001	-.015	.988	-.429	.422

a. Dependent Variable: Criterion

remaining focus of this chapter is a discussion of statistical analysis, patterns produced, and empirical observations, followed by many iterations of development in fit forming and testing.

4.3.1 Model Exploration

For model assessment and exploration, a multi-factorial experimental approach is leveraged to analyze the computational model's results. Independent variables are tested to determine an ideal distribution of manpower, training, and network quality against an anticipated adversarial attacker. The independent variables for experimentation are the user inputs, including the number of network nodes, number of attackers and defenders,

quality of the network, skills and abilities of each force, and self-efficacy of the friendly force (the adversary force is a predetermined seven of ten) (Table 4.7). The dependent variables are availability index and sustainability index. The availability index is an indication of the network security based on a summation of available, non-exploited, or offline nodes per step over until max iterations (1000 steps) or full network compromise (Equation 4.1).

Table 4.7: Computational Model Independent Variables

Model Parameters			
Type	Characteristics	Characteristics	Characteristics
Environment	Quantity (Access nodes) 1-100	Network (Security Strength) 1-100	
Defender	Quantity 1-100	Skill 1-10	Self-Efficacy 1-10
Attacker	Quantity 1-100	Skill 1-10	Self-Efficacy 1-10

$$Ai = \sum_{i=Do}^{steps} n/Do \quad (4.1)$$

Where Ai is availability index, n is total node count, and Do is nodes not in *offline* or *outage* status. The availability index asks the question: if directly attacked, how much of the network remains effective prior to full network compromise? An important metric, but the availability index doesn't account for a network that is compromised earlier or later than an alternative set of parameters (a key consideration for network defenders). To account for this, the sustainability index is the number of steps achieved during a direct attack prior to network compromise. If the defenders maintain the network for 1000 steps, an index of 1000 is achieved; if 100 steps are achieved prior to full network compromise, the sustainability index is 100. Simply stated: if under cyber attack, how long can network connectivity be maintained? Each step in the simulation can be considered a real-world day for model-to-real-world comparison. Recognizing that robust capability and

vulnerability rates quantify node and connectivity qualities [141], the following metrics provide the necessary data for applicable statistical analysis. Discussion in Chapter Five includes modification and expansion of applicable formulas for increased fidelity.

Table 4.8: Baseline Parameters for Model Analysis

Baseline Parameters			
Type	Characteristics	Characteristics	Characteristics
Environment	Node Count 15	Network Security 50	
Defender	Defender Count 2	Skill 7	Self-Efficacy 7
Attacker	Attacker Count 4	Skill 7	Self-Efficacy 7

Due to the nature of agent-based models demonstrating nonlinear, emergent phenomena, a single simulation of the model is insufficient to interpret model results. A determination must be made regarding simulation runtime length. Using complete network compromise as the explicit event for model completion leveraging the baseline parameters defined in Table 4.8, a runtime analysis is conducted to determine the ideal runtime length. Based on 1,000 iterations, the results indicate all runs end in full network compromise prior to 225 steps. Simulations over variations on other parameters indicated that a simulation extending beyond 1000 steps appears to run indefinitely. The remaining baseline simulations will run to network corruption or a maximum of 1000 steps.

4.3.2 Model Sensitivity Analysis

Model sensitivity analysis is conducted to determine the parameters of greatest significance and how each parameter impacted the model's overall functionality. Specific goals for sensitivity analysis include model calibration, factor impact, direction of change, and an analysis of emergent properties. Each user input serves as an independent variable for model analysis. Two separate designs are tested against the cyber computational

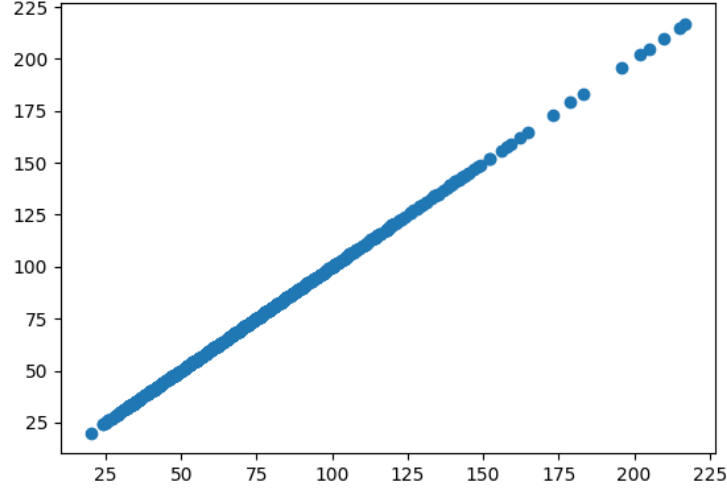


Figure 4.2: Baseline Parameters Runtime Length Analysis. Model length (steps) on both x and y axis

performance model, the one-factor-at-a-time (OFAT or local sensitivity) and the global sensitivity analysis (GSA) design. The local sensitivity tests included each of the independent variables for the full range of input, by increments of five when the range was 1-100 or increments of one when the range was 1-10. Following simulation runs of both designs, multivariate analysis of variance (MANOVA) was conducted to determine statistical significance, mean, standard deviation (σ), correlation via regression, and overall direction and degree of correlation to the dependent variables [142], [143]. The results of the OFAT demonstrated which parameters and range to be analyzed through GSA based on decomposition and tipping point [136]. The Confidence level for all tests was set to 95% (displayed in each of the following graphs). Test inputs selected represented a broad sample from the range of user-available options while maintaining consideration of computational costs and limitations of the VMWARE virtual machine. A full set of statistical results and associated graphs is presented in Appendix E, while graphs of note are presented below for further review and discussion.

Grid sampling, or variation of parameters, determines the impact and relationships across the agents and dependent variables. To determine the effective use of logical resources, the independent variables are tested with the intent of assessing the distribution

of manpower, training, and network quality against a known adversarial attacker. Baseline fundamentals for the model were previously presented in Table 4.8. The baseline values consisted of two friendly defenders, four adversarial attackers, and fifteen nodes, reflecting a DarkSide affiliate group attacking a small government computer infrastructure or public organization. This baseline defined the initial model fit for formula assessment, allowing variations in model inputs to demonstrate emergent qualities. Upon completion of 500 batch simulations per parameter, the results were analyzed through a MANOVA model via SPSS to determine statistical significance, factor impact, mean results, standard deviation σ , and correlation.

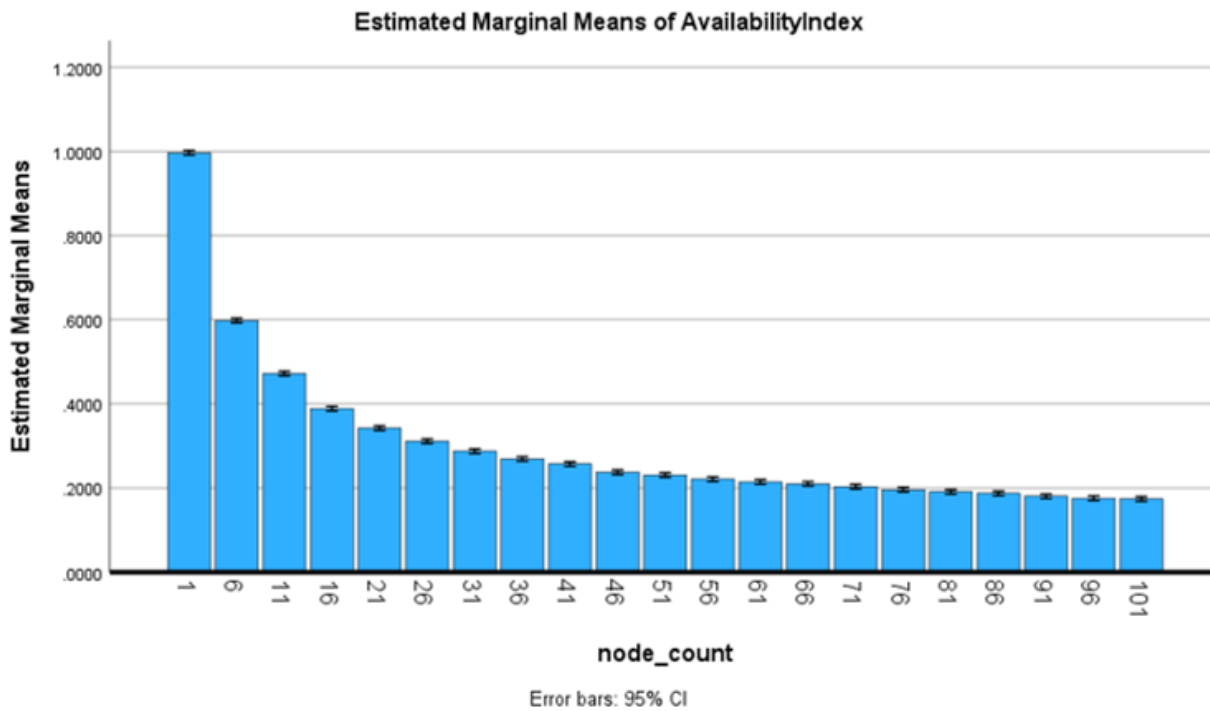


Figure 4.3: Node Count MANOVA against Availability

Node Count is a user input that allows a network size variation between one and one hundred access points. Based on baseline parameters varying the node count from 1 to 101 with increments of five, the model demonstrated statistical significance ($p < .001$), a mean availability index of .302, SD σ .197, and mean sustainability index of 253.956, SD σ 184.345, negative directional results with the availability index and positive directional

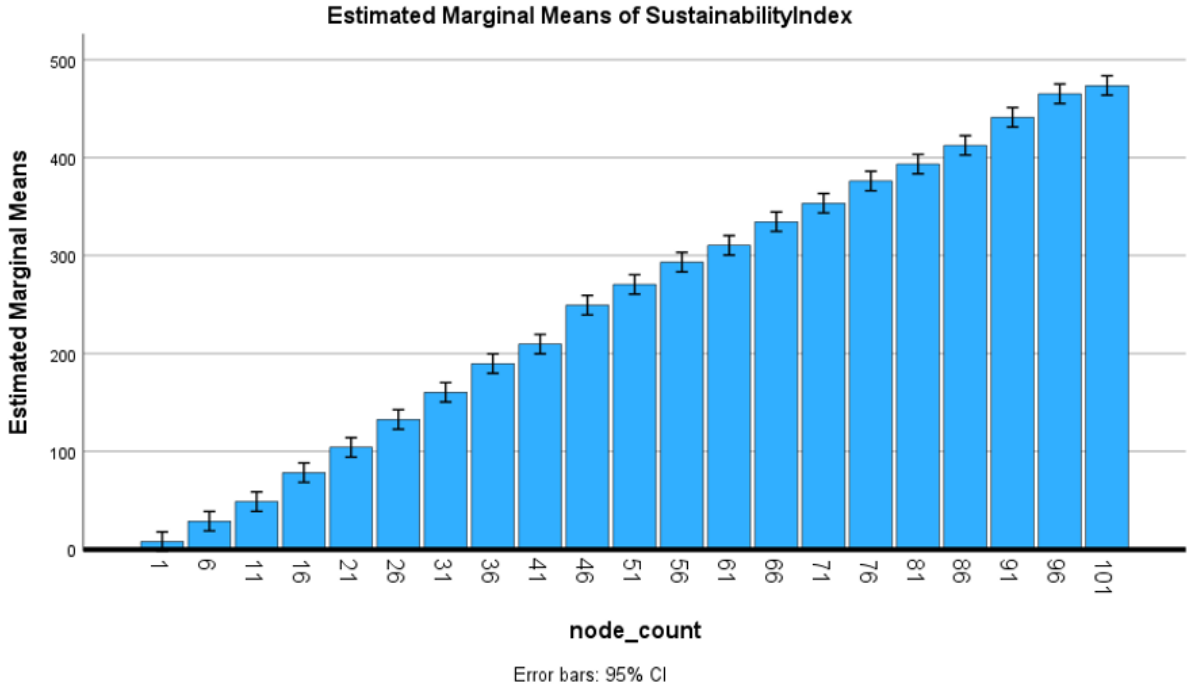


Figure 4.4: Node Count MANOVA against Sustainability

results with the sustainability index. In review, the impact on availability levels off significantly above 25 nodes (Figure 4.3) with a positive linear relationship between node count and sustainability (Figure 4.4). Intuitively, the greater the number of network nodes (access points), the lower the overall percentage of network connectivity (availability index) maintained while under direct attack. However, despite a lower availability index with increased nodes, increasing nodes results in a positive linear correlation in availability, up to 500 steps at the base parameters.

Security strength is a user input that defines a network's security status on a scale of 1 to 100. Based on baseline parameters varying the security strength from 1 to 101 with increments of five, the model demonstrated statistical significance ($p < .001$), a mean availability index of .396, SD σ .087, and mean sustainability index of 71.754, SD σ 28.06. The results indicated very little directional impact across the range of inputs for security strength. While analysis of the network security measure indicated a statistical

significance, the relationship is not nearly as great as node count.

Friendly operator count is a user input that defines the number of friendly operators attacking or defending on a scale of 1 to 100. Based on baseline parameters, the friendly operator is set to defending, and the parameter is varied from 1 to 101 with increments of five. The model demonstrated statistical significance ($p < .001$), a mean availability index of .859, SD σ .148 and mean sustainability index of 935.462, SD σ 226.450. Within the bar chart presented here is a potential emergent phenomenon with a tipping point between 6 and 11 defenders in relation to the number of nodes and attackers (Figure 4.6). Little surprise that twice the number of attackers and a two-to-three ratio of defenders to nodes produced diminishing returns. Additional analysis of the defender count as an independent variable was conducted and presented below.

SustainabilityIndex

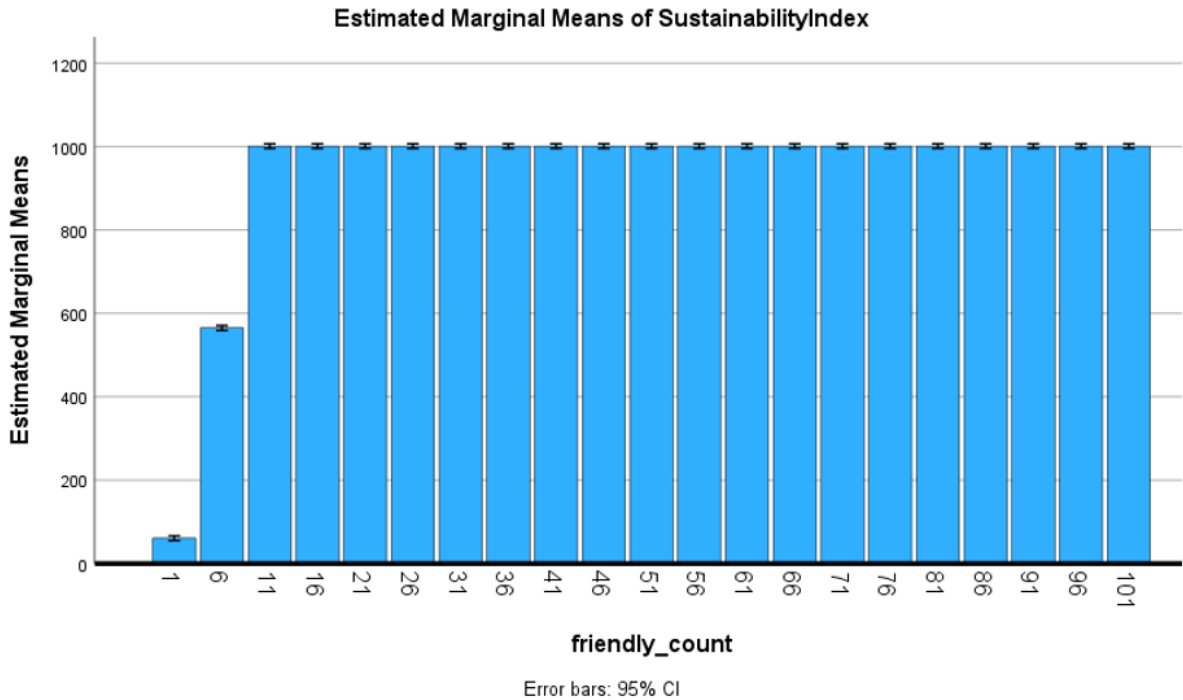


Figure 4.5: Friendly Count MANOVA against Sustainability

Friendly efficacy is a user input that defines the behavioral characteristic of friendly operators attacking or defending on a scale of 1 to 10. Derived from the COSES, this

value can easily be obtained and applied to the model for real-world application. Based on baseline parameters, the friendly operator is set to defending, and the parameter is varied from 1 to 10 with increments of one. The model demonstrated statistical significance ($p < .001$), a mean availability index of .407, SD σ .080, and mean sustainability index of 69.554, SD σ 26.929, with negligible correlation across the range of inputs.

Friendly skills is a user input that defines the measure of an attacker or defender's cyber proficiency and experience on a scale of 1 to 10. Based on baseline parameters, the friendly operator is set to defending, and the parameter is varied from 1 to 10 with increments of one. The model demonstrated statistical significance ($p < .001$), a mean availability index of .413, SD σ .081, and a sustainability index of 66.360, SD σ 26.003, with little directional impact across the range of inputs. An interesting phenomenon was observed in both friendly skill and efficacy, in which the highest rate of each (set to ten) displayed a lower mean result than a setting of nine, specifically for the sustainability index (Figure 4.6 is friendly skills vs sustainability). While the impact of friendly skill and efficacy is fairly small compared to the operator or node count, an error can account for some of these unexpected results, and additional model fitting may be appropriate through future tests and analysis of specific case studies.

Adversary operator count is a user input that defines the number of adversary operators attacking or defending on a scale of 1 to 100. Based on baseline parameters, the adversary operator is set to attacking, and the parameter is varied from 1 to 101 with increments of five. The model demonstrated statistical significance ($p < .001$), a mean availability index of .431, SD σ .115, and mean sustainability index of 65.528, SD σ 209.247. As with the previously discussed friendly operator count, the adversary count presented a clear tipping point between 1 and 6 attackers (Table 4.7). Further testing and results are presented below.

The last primary GSA MANOVA test conducted was on adversary skills, a user input that defines the proficiency of an adversary's cyber knowledge and experience while at-

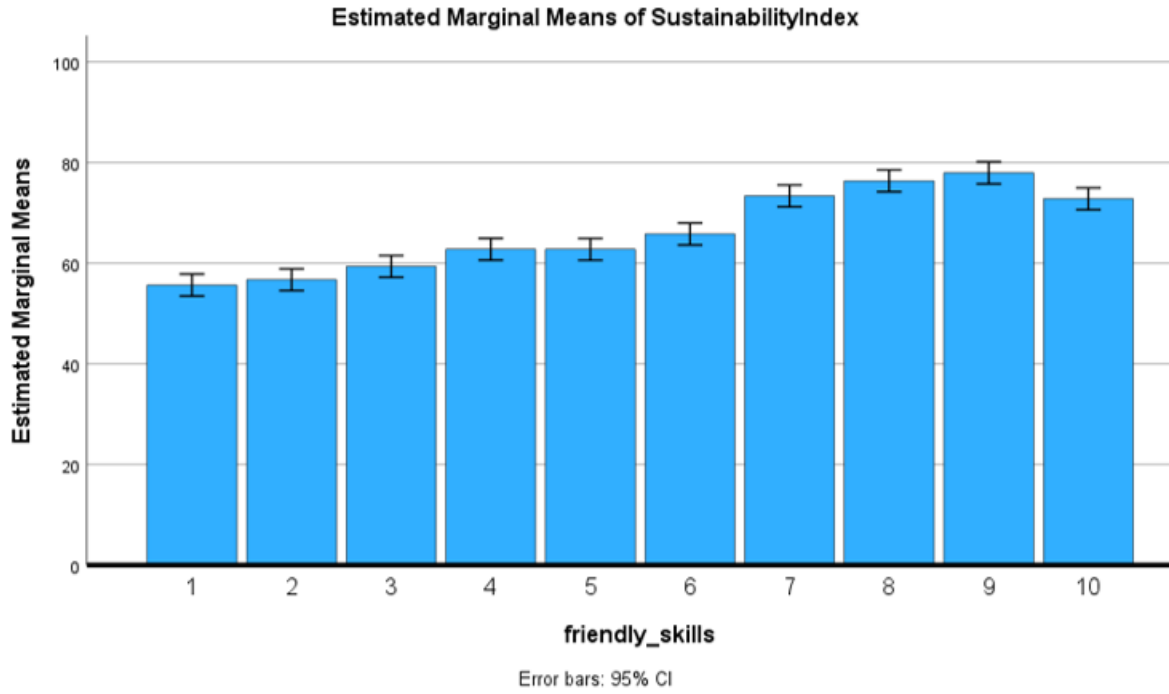


Figure 4.6: Friendly Skills MANOVA against Sustainability

tacking or defending on a scale of 1 to 10. Based on baseline parameters, the adversary operator is set to attacking, and the parameter is varied from 1 to 10 with increments of one. The model demonstrated statistical significance ($p < .001$), a mean availability index of .480, SD σ .191, and a sustainability index of 228.595, SD σ 299.119, with a much stronger directional impact across the range of inputs than the defender skills variable (Figure 4.8).

Based on findings observed through OFAT model analysis, operator count (both attacker and defender) produced results indicative of emergent qualities. Additionally, the operator count and node count parameters demonstrated the greatest sensitivity across the model. To examine more closely, the simulations were run again, focusing on the respective tipping points. Both sets of simulations were run on a scale of 1 to 15 in increments of one. The results (available in full detail in Appendix E) indicate the tipping point between five and seven for defending units and one and two for attacking units.

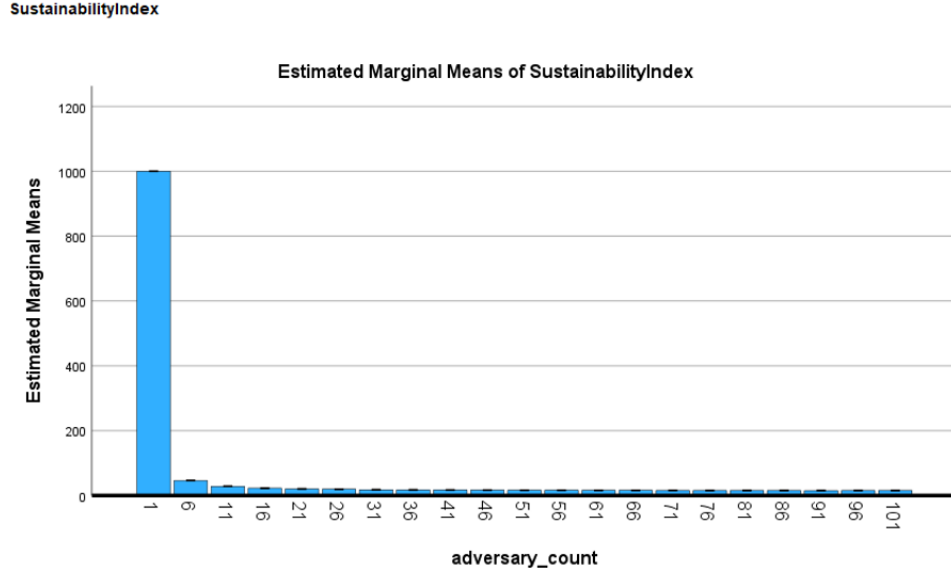


Figure 4.7: Adversary Count MANOVA against Sustainability

Here it's worth reemphasizing the baseline parameters of four attackers and two defenders. The influence between attacker and defender count is illustrated more clearly via the GSA analysis. Based on the range of input demonstrating potential emergent qualities, these data points were selected for GSA analysis.

While local analysis demonstrates which parameters have the most significant sensitivity to model output, it does not reveal the impact across independent variables or total model variance. For this, global sensitivity analysis is required. With GSA, each of the selected independent variables is varied to determine overall model variance and sensitivity. To conduct this analysis, the targeted areas of 1 to 16 in increments of one are selected for both friendly and adversary count while also applying node count from 1 to 30 in increments of five. The results presented below are achieved via MANCOVA statistical analysis with friendly and adversary (defender and attacker) count as independent variables, availability, and sustainability indexes as dependent variables, and node count as the covariate. To obtain a manageable data set, the number of simulation iterations was reduced to ten and run multiple times to minimize small sample size error. Unsurprisingly, the results indicated statistical significance for each of the independent variables

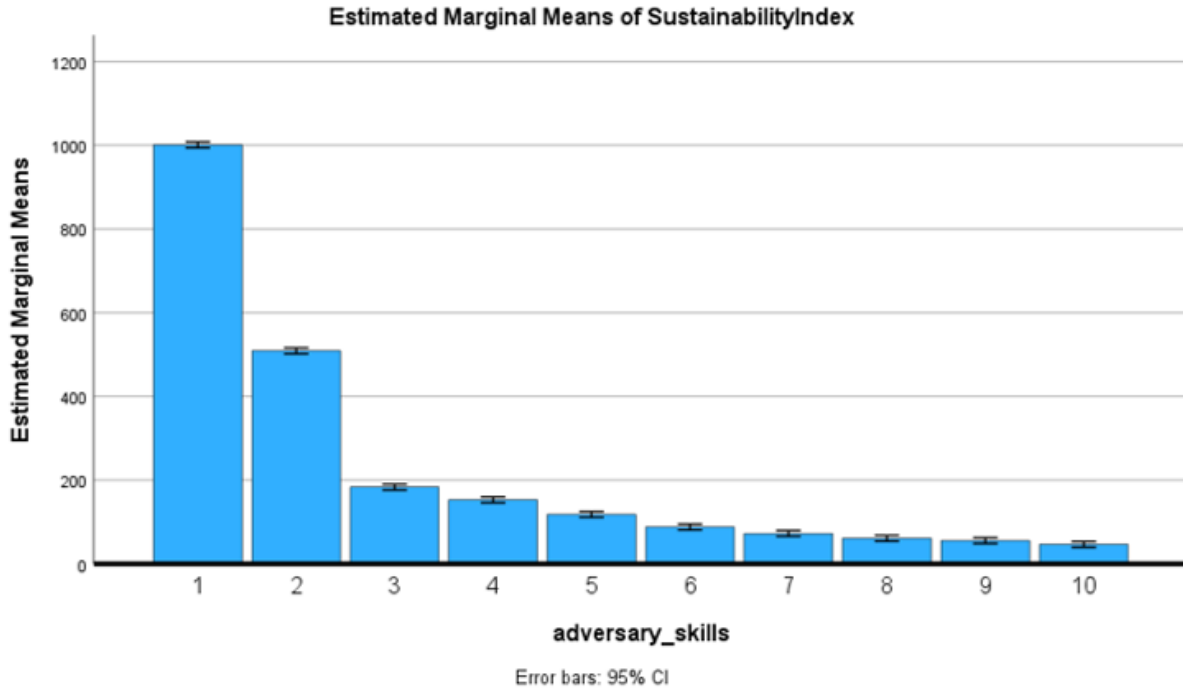


Figure 4.8: Adversary Skills MANOVA against Sustainability

($p < .001$) and positive correlation. Examining the multivariate line chart with friendly count versus adversary count (defender versus attacker) through the targeted variable range of interest, the results demonstrated the impact on model simulation of a single attacker or defender against the range of inputs (a clear tipping point) and illustrated the bootstrapped median availability (.586, SD σ .267) and sustainability (431, SD σ 447.975) indexes for the selected range (Figure 4.9 and 4.10).

Visually analyzing data on the compound line graphs highlighted the emergent quality of the data as the increase in friendly and adversary operators directly alters the shape and power of the line graph while demonstrating the influence each variable has on one another and the model as a whole. Notice how the shape of the line, represented by adversary count, changes as the adversary count increases. It's also worth observing the nonlinear growth of the curve as friendly count increases across the horizontal axis. While this may not appear a practical observation, without modeling the system and analyzing

AvailabilityIndex

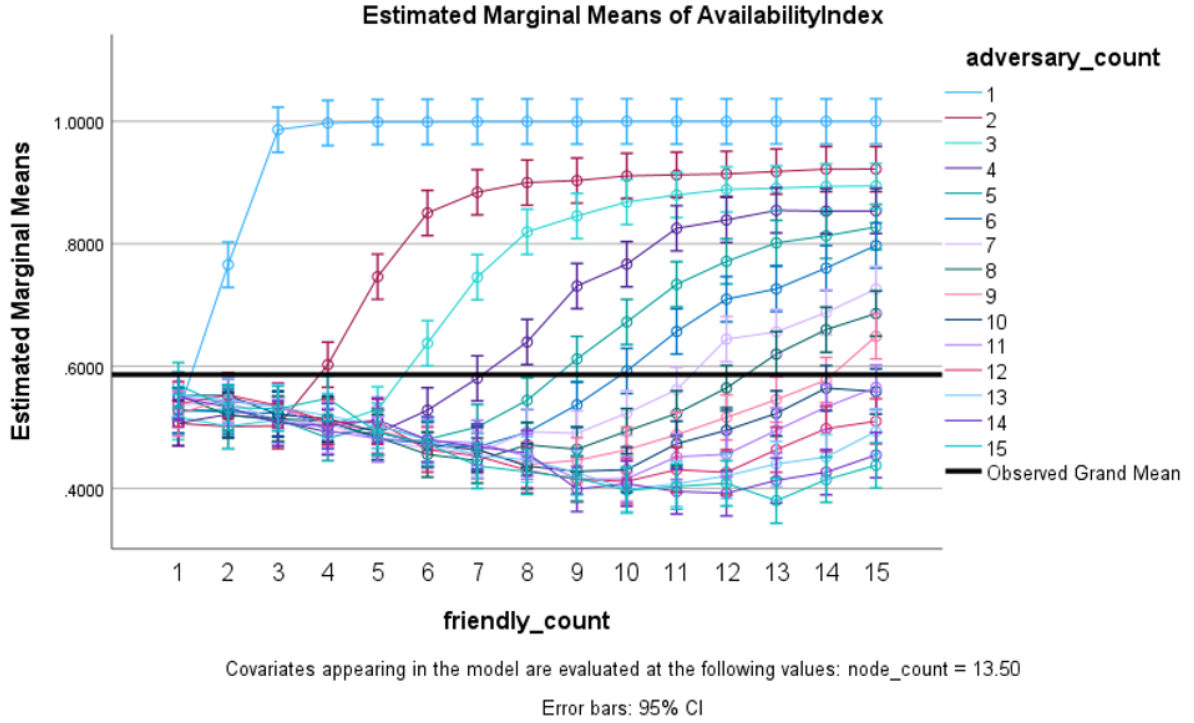


Figure 4.9: Node Count MANCOVA against Availability

the results, one might assume a linear or exponential growth across the adversary, friend, and node count variables. Doing so would lead to investing in network and cybersecurity resources without appreciating the impact that increasing elements within the system can have on overall functionality. As previously discussed in Chapter Two, modeling and simulation is the definitive approach for confirming emergent qualities within a complex system.

As one might hypothesize, increasing defender count strengthened the overall defense of the network with two important considerations. A leveling effect, or diminishing return, occurred directly in relation to the number of network access points and anticipated defenders. This will, of course, be influenced by operator skill efficacy and network strength, so should be calculated for the desired real-world variation of parameters. Second, when under direct attack by multiple, proficient cyber offense operators, some degree of network loss must be expected. Leveraging the ROI (return on investment) element of the dash-

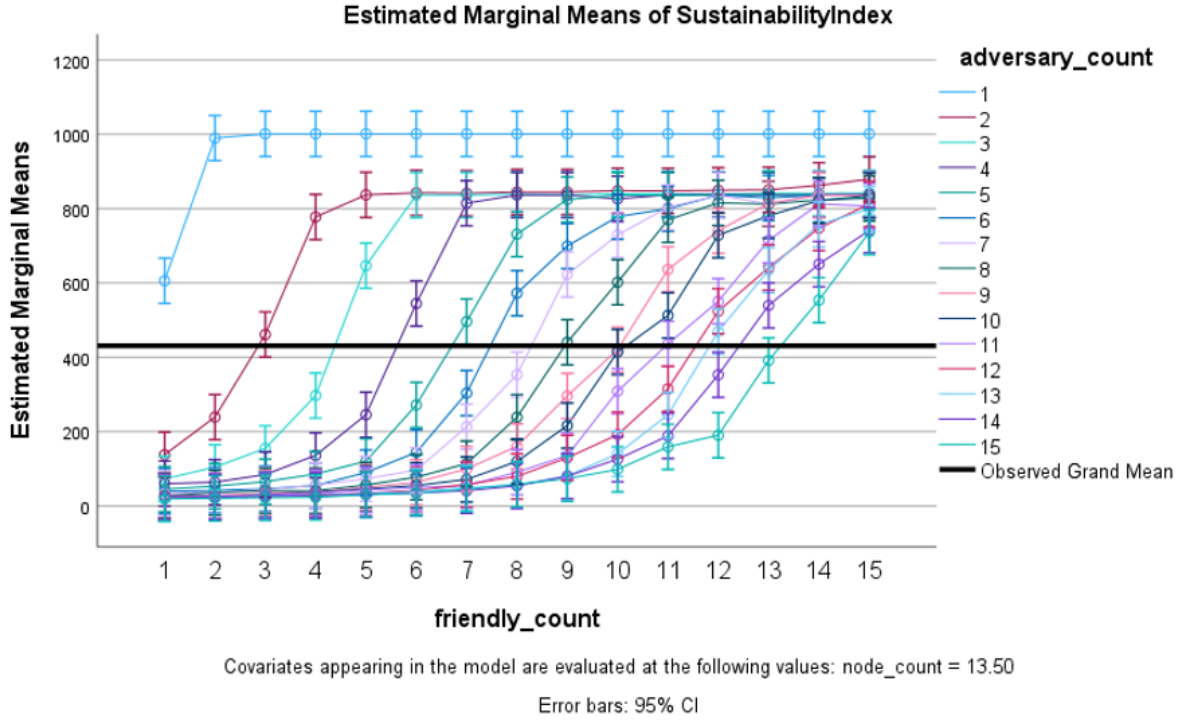


Figure 4.10: Node Count MANCOVA against Sustainability

board will enable organizational leaders to determine what cost/risk balance to target when considering the number of cybersecurity operators to employ. Based on relevant data and case study review, the model reflected the intended real-world fit and demonstrated applicable emergent properties. As noted by [132], the challenge in developing an effective computational model is balancing data fit versus applicability to real-world scenarios. To this end, the Cyber Operations Performance Framework achieved this balance through the flexibility to modify agent criteria while remaining relatively simple in design and execution. Variations of approaches in modeling cyber warfare are presented in Chapter Two, and considerations for this framework's strengths, areas for improvements, and future research are discussed in Chapter Five.

Chapter 5

Conclusions

5.1 Summary

Cyber attack operators work to identify, infiltrate and abuse target computer and network systems. Their ability to do so is influenced by their self-efficacy, skills, and expertise regarding cyber-attack activities and the quality of the target's network. Cybersecurity agents monitor, update, repair, and recover the network and likewise are influenced by the same metrics. The purpose of this research was to develop a framework in which real-world cyber performance could be assessed and predicted in real-time for a variety of organizational types and sizes. To achieve this, three research objectives were defined:

1. RO1 Develop a computational model that predicts cyber operational performance.
2. RO2 Develop a self-efficacy scale, the Self-Efficacy Cyber Operations Scales (COSES), as an input to the computational model for behavioral characteristic influence.
3. RO3 Develop a cyber performance dashboard that provides users with a simple-to-understand assessment of cyber performance based on user input.

The Cyber Performance Framework computational model is an agent-based model that allows users the flexibility to custom-tailor parameters to best fit the model to their real-world circumstances. When the model is adjusted based on an organization's cyber data

and details, the model’s simulations provide far more accurate results and predictions for the end user to apply than are currently available for effective cyber operations decision-making. Through the use of construction and validation techniques, the COSES is a two-survey scale used to provide accurate and relevant behavioral character data. Statistical analysis demonstrated its simple, easy, yet powerful capacity to capture cyber operator self-efficacy and deliver user-specific data to improve the model’s real-world accuracy.

The cyber performance dashboard allows users to observe in real-time the overall network status and changes within the network at each network access point throughout the model’s simulation. A text-based dollar costing feature allows users to establish cost estimates for defenders and outages. This feature greatly facilitates considerations regarding manpower, training, and network strength when determining how to obtain maximum return on investment for cyber expenditures.

Through the use of the Cyber Operations Performance Framework, organizations can compare real-world cyber operations with model results to determine areas for performance improvement, predict operational success and evaluate the potential for cyber investments. The framework consists of simple-to-use tools such as the COSES scales and computational model that better equip organizations to manage cyber operations and logistical investments to achieve cyber success.

5.2 Contributions

Agent-based models are designed across a spectrum of use cases, from hypothesis evaluation to prediction based on the model’s reflection of real-world conditions. The greater the detail in constructing the model, the greater accuracy, but at the expense of applicable scope. The Cyber Operations Performance Framework was developed based on the previously mentioned research objectives and moderates the challenge of prediction scope and applicability by allowing users to apply parameters specific to their intended use

case. The model's general scope is applicable to a wide range of organizations, while execution is specific to user-defined circumstances. This research has produced the following significant contributions:

1. A flexible computational model that demonstrates cyber operational performance.
2. A first of its kind, set of self-efficacy cyber scales (COSES)
3. An ROI-driven metrics dashboard for cyber operation decision-making.
4. An approach effectively leveraging both quantitative and design science methodologies,

In both research and practice, the model simulates cybersecurity and cyber offense with parameters that influence agents at the micro level while resulting in emergent phenomena at the macro level. The analysis of a cyber operation as it unfolds would otherwise be difficult to achieve without the ability to observe and analyze the system's characteristics through the agent-based model.

The Cyber Operational Self-Efficacy Scales (COSES) are the first of their kind in providing cyber operators and leaders insight into operator self-efficacy and performance. While a variety of tools exist to examine the self-efficacy of cyber lay-person performance, none until now measured the confidence of those trained in executing cyber operations. This was a critical gap across the cyber literature research and is now available for individual and organizational improvement and future research.

The cyber performance framework also provides considerable future research value in hypothesis testing. The complex nature of cyber attacks and security often results in false presumptions regarding the logistical and operational impacts of cyber operations. Researchers can use simulation results as a baseline for theory development and testing prior to lab development or large N research efforts.

5.3 Limitations & Future Research

5.3.1 COSES Data Collection

Collecting accurate feedback in a survey designed to quantify performance will always be faced with challenges. These challenges become significantly greater due to the secretive nature of cyber data and operator performance. Effective sampling, measurement, and bias control were impacted by the blind nature of the research’s request for survey participation across the internet. While the number of respondents that provided feedback to validate the overall survey was adequate for statistical analysis, a larger sample size derived from a controlled population may improve survey accuracy and honesty. Initial scale development relied on a population of varied respondent types to minimize bias, but future research in leveraging the COSES may seek to analyze target populations to confirm results are not skewed based on the nature of the blind collection of data obtained in this research.

5.3.2 Model Data Validation

As previously noted, metrics are only as reliable as the validity and reliability of their measurement. Without direct empirical data to compare, the computational model’s ability to accurately predict performance under various conditions deserves continued testing. When leveraged by organizations that have direct access to the sensitive data that’s relevant for such testing, the user can confirm modeling results and adjust the agent-based modeling parameters or underlying fundamentals as required. From there, simulation of future events can be achieved with a much higher degree of confidence and reliability.

Quantitative assumptions are a necessary requirement to develop computational models. The computational model presented here was aligned with previous type models in

developing quantitative characteristics to define network and node degradation and status. Over the last twenty years, relatively few improvements to network security have not resulted in parallel developments in offensive techniques. That withstanding, the Cyber Operations Performance computational model is developed and intended to be leveraged as a framework, and its use should be modified to reflect the dynamic landscape of cyber operations as it continues to evolve.

5.3.3 Attack Type & Attacker Variations

Existing cyber threat models provide varying details of motivation, techniques, and objectives that result in a wide degree of detail in types of attackers, types of attacks, and resulting impact on network functionality. As detailed throughout this research, the challenge to creating an effective agent-based model is balancing detail and specificity versus simplicity and broader application. This initial version of the cyber performance model was developed based on a structure assuming all agents of a type (offensive, defensive) have the same skills, self-efficacy, motivations, and objectives. While this facilitates a simple model of agent interaction, it lacks the potential nuance that could be helpful in decision-making aimed at particular types of attacks or attackers. Future development of the framework that provides options to select from a variety of attackers or attack types (such as DDOS, ransomware, rootkits, etc) could deliver greater fidelity and richer, though significantly more complex, results.

5.4 Organizational Adaptation

Organizations are encouraged to leverage the Cyber Operations Performance Framework as a feature of their cyber program. As a baseline, every organization must establish a cyber assessment program that consists of a cycle of steps to develop and maintain effective cyber operations (Figure 5.1):

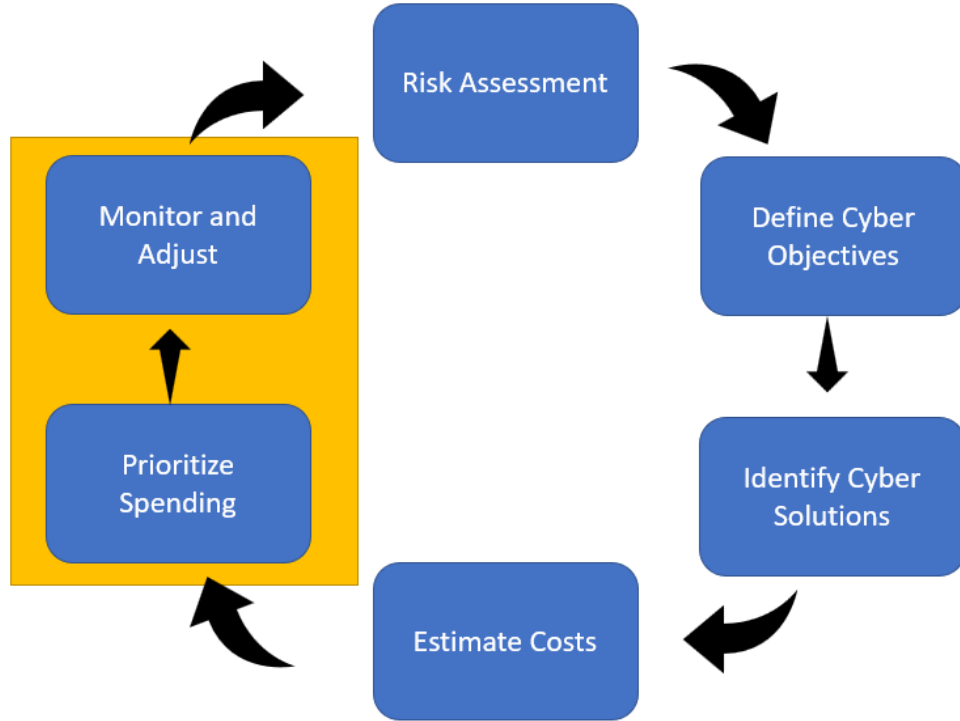


Figure 5.1: Cyber Program Management Cycle

As highlighted in the presented example, the Cyber Operations Performance Framework is designed to impact the prioritize spending and monitor and adjust phases. When used this way, the computational model can provide insight regarding potential performance based on threats and objectives established in the previous steps. As the cycle continues, fine-tuning objectives and solutions can further inform parameter settings. Conducting COSES assessments should occur as often as desired, based on team turnover, cyber training, or mission-set updates. Determining the appropriate timeline for an organization's cyber program management cycle will depend on the specific organization, but quarterly, biannually, or annually based on the above criteria is an appropriate starting point.

The computational model and COSES are both available through my personal GitHub page (<https://github.com/Bbecote>). Organizations are encouraged to pull and clone the computational model and leverage it with internally where sensitive data is protected. Publication of information gained through the use of the Cyber Operations Performance Framework or COSES should include attribution and citation to this or the applicable

publication. See the GitHub page for detailed guidance on running and updating the computational model.

5.5 Conclusions

The Cyber Operations Performance Framework represents an important step forward in leveraging complexity science to improve cyber operations. As a field, cyber operations is heavily reliant on practical application, often at the expense of the theoretical and experimental grounding that other research domains enjoy.

Complexity science is not without areas for further research. As a scientific framework, it is an evolutionary leap forward in understanding system adaptation and evolution, but a variety of challenges remain, including validity and reliability testing, stakeholder understanding of model implications, and the application of data science on large-scale models. Each of these areas directly impacts cyber operations research and practice, and dedicated efforts across both fields can help bridge current challenges and gaps in knowledge.

While still relatively young as a field, complexity science has proven a revolutionary force in understanding and interpreting the world around us. Cyber operations research and practice can employ complexity models including time series analysis and agent-based modeling to interpret and predict cyber operations. The very nature of cyberspace as a dynamic and continuously evolving environment will no doubt challenge researchers to create controlled conditions for experimentation or leverage modeling to simulate real-world systems. The way forward will combine the power and capability of complex systems modeling while continuing to build a foundation for effective research and policy-making through a formal association with complexity science. Further employing complexity science as a foundation for cyber operations will allow for scientifically robust testing, formalized metrics, and improved tool development through simulation and modeling

efforts.

While agent-based modeling is not the only modeling and simulation framework for demonstrating complex systems, the ease with which researchers can develop, observe, and experiment with emergence compared to alternatives can not be overstated. Cyber operations researchers and practitioners can quickly develop agent-based models with free open-source software. Software options with varying strengths and learning curves are available to examine and simulate cyber operations across all OS platforms. With a combination of community support, in-depth online tutorials, and free resources, cyber operations professionals have a great deal to gain from exploring the incredible potential of computational modeling and complexity science.

References

- [1] S. Morgan, *Global Cybersecurity Spending Predicted To Exceed \$1 Trillion From 2017-2021*, Section: Cybersecurity Market Report, Jun. 2019. [Online]. Available: <https://cybersecurityventures.com/cybersecurity-market-report/>.
- [2] J. Healey, *The Cyber Budget Shows What the U.S. Values—And It Isn't Defense*, Jun. 2020. [Online]. Available: <https://www.lawfareblog.com/cyber-budget-shows-what-us-values%E2%80%94and-it-isnt-defense>.
- [3] B. Becote and B. Rimal, “Complex Systems Science and Cyber Operations: A Comprehensive Survey and Analysis,” 2023.
- [4] H. Sayama, *Introduction to the Modeling and Analysis of Complex Systems*. Albany: Open SUNY Textbooks, 2015.
- [5] G. E. Mobus and M. C. Kalton, *Principles of Systems Science*. Springer, Nov. 2014.
- [6] P. Phister, “Cyberspace: The Ultimate Complex Adaptive System,” *International C2 Journal*, vol. 4, no. 2, 2010.
- [7] F. Chavez-Juarez, “On the Role of Agent-Based Modeling in the Theory of Development Economics,” *Review of Development Economics*, vol. 21, no. 3, pp. 713–730, 2017.
- [8] S. Bora and S. Emek, *Agent-Based Modeling and Simulation of Biological Systems*. IntechOpen, 2018.
- [9] K. Kollman and S. E. Page, “Chapter 29 Computational Methods and Models of Politics,” in *Handbook of Computational Economics*, L. Tesfatsion and K. L. Judd, Eds., vol. 2, Elsevier, Jan. 2006, pp. 1433–1463.
- [10] U. Wilensky and W. Rand, *An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*. Cambridge, Massachusetts: The MIT Press, 2015.
- [11] U.-M. O'Reilly, J. Toutouh, M. Pertierra, *et al.*, “Adversarial genetic programming for cyber security: A rising application domain where GP matters,” *Genetic Programming and Evolvable Machines*, 2020. DOI: 10.1007/s10710-020-09389-y.

- [12] G. A. Francia III, X. P. Francia, and C. Bridges, “Agent-Based Modeling of Entity Behavior in Cybersecurity,” in *Advances in Cybersecurity Management*, K. Daimi and C. Peoples, Eds., Cham: Springer International Publishing, 2021, pp. 3–18.
- [13] L. Ashiku and C. Dagli, “Agent Based Cybersecurity Model for Business Entity Risk Assessment,” in *2020 IEEE International Symposium on Systems Engineering (ISSE)*, ISSN: 2687-8828, Oct. 2020, pp. 1–6.
- [14] A. Bandura, “Self-efficacy: Toward a unifying theory of behavioral change,” *Psychological Review*, vol. 84, no. 2, pp. 191–215, 1977, Place: US Publisher: American Psychological Association, ISSN: 1939-1471. DOI: 10.1037/0033-295X.84.2.191.
- [15] H. Marsh, R. Pekrun, P. Parker, *et al.*, “The Murky Distinction Between Self-Concept and Self-Efficacy: Beware of Lurking Jingle-Jangle Fallacies,” *Journal of Educational Psychology*, 2019. DOI: 10.1037/edu0000281.
- [16] K. Bartimote-Aufflick, A. J. Bridgeman, R. A. Walker, M. D. Sharma, and L. D. Smith, “The study, evaluation, and improvement of university student self-efficacy,” 2016. DOI: 10.1080/03075079.2014.999319.
- [17] S. Cassidy, “Resilience Building in Students: The Role of Academic Self-Efficacy,” *Front. Psychol.*, 2015. DOI: 10.3389/fpsyg.2015.01781.
- [18] L. Gholami, “Teacher Self-Efficacy and Teacher Burnout: A Study of Relations,” *International Letters of Social and Humanistic Sciences*, vol. 60, pp. 83–86, 2015.
- [19] P. Drogkaris and A. Bourka, *Cybersecurity culture guidelines: behavioural aspects of cybersecurity*. Publications Office of the European Union, 2018.
- [20] M. A. Hameed and N. A. G. Arachchilage, “Understanding the influence of Individual’s Self-efficacy for Information Systems Security Innovation Adoption: A Systematic Literature Review,” *arXiv:1809.10890 [cs]*, Sep. 2018, arXiv: 1809.10890.
- [21] M. Bashir, C. Wee, N. Memon, and B. Guo, “Profiling cybersecurity competition participants: Self-efficacy, decision-making and interests predict effectiveness of competitions as a recruitment tool,” *Computers & Security*, vol. 65, pp. 153–165, Mar. 2017.
- [22] G. Dobson and K. Carley, “Cyber-FIT: An Agent-Based Modelling Approach to Simulating Cyber Warfare,” Jun. 2017, pp. 139–148, ISBN: 978-3-319-60239-4. DOI: 10.1007/978-3-319-60240-0_18.
- [23] Z. Maqbool, V. Pammi, and V. Dutt, “Computational Modeling of Decisions in Cyber-Security Games in the Presence or Absence of Interdependence Information,” in Jul. 2021, In press.

- [24] A. B. Downey, *Think Complexity: Complexity Science and Computational Modeling*, 2nd edition. Boston: O'Reilly Media, 2018.
- [25] B. Castellani and F. W. Hafferty, *Sociology and Complexity Science: A New Field of Inquiry*. Berlin: Springer, 2009, ISBN: 978-3-642-10013-0.
- [26] M. C. Jackson, *Critical Systems Thinking and the Management of Complexity*. Hoboken, NJ: Wiley, 2019.
- [27] L. V. Bertalanffy, *General System Theory: Foundations, Development, Applications*, Revised edition. New York, NY: George Braziller Inc., 1968, ISBN: 978-0-8076-0453-3.
- [28] N. Wiener, *Cybernetics: Second Edition: Or the Control and Communication in the Animal and the Machine*. Cambridge, Massachusetts: MIT Press, 1948.
- [29] B. Macukow, "Neural Networks – State of Art, Brief History, Basic Models and Architecture," in *Computer Information Systems and Industrial Management*, K. Saeed and W. Homenda, Eds., ser. Lecture Notes in Computer Science, Springer International Publishing, 2016, pp. 3–14.
- [30] G. A. Cowan, *Manhattan Project to the Santa Fe Institute: The Memoirs of George A. Cowan*. Albuquerque: University of New Mexico Press, 2010.
- [31] S. Xu, "Emergent behavior in cybersecurity," in *Proceedings of the 2014 Symposium and Bootcamp on the Science of Security*, New York, NY: Association for Computing Machinery, Apr. 2014, pp. 1–2.
- [32] S. Moskal, S. J. Yang, and M. Kuhl, "Cyber threat assessment via attack scenario simulation using an integrated adversary and network modeling approach," *The Journal of Defense Modeling and Simulation*, vol. 15, no. 1, pp. 13–29, Jan. 2018.
- [33] A. Attiah, M. Chatterjee, and C. C. Zou, "A Game Theoretic Approach to Model Cyber Attack and Defense Strategies," in *2018 IEEE International Conference on Communications (ICC)*, Kansas City, MO, May 2018, pp. 1–7.
- [34] R. Mitchell and B. Healy, "A game theoretic model of computer network exploitation campaigns," in *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*, Las Vegas, NV, Jan. 2018, pp. 431–438.
- [35] C. A. Kamhoua, C. D. Kiekintveld, F. Fang, and Q. Zhu, Eds., *Game Theory and Machine Learning for Cyber Security*. Hoboken, New Jersey: Wiley, 2021.

- [36] A. Zarreh, C. Saygin, H. Wan, Y. Lee, and A. Bracho, “A game theory based cybersecurity assessment model for advanced manufacturing systems,” *Procedia Manufacturing*, vol. 26, pp. 1255–1264, Jan. 2018.
- [37] A. Iqbal, L. J. Gunn, M. Guo, M. A. Babar, and D. Abbott, “Game Theoretical Modelling of Network/Cybersecurity,” *IEEE Access*, vol. 7, pp. 154 167–154 179, 2019. DOI: 10.1109/ACCESS.2019.2948356.
- [38] A. Chukwudi, U. Eze, and C. Ikerionwu, “Game Theory Basics and Its Application in Cyber Security,” *Advances in Wireless Communications and Networks*, vol. 3, pp. 45–49, Jul. 2017.
- [39] A. Mcleod, A. Dorantes, and G. Dietrich, “Modeling Security Vulnerabilities Using Chaos Theory: Discovering Order, Structure and Patterns from Chaotic Behavior in Complex Systems,” Las Vegas, Nevada, Jun. 2008.
- [40] A. Dorantes, A. Mcleod, and G. Dietrich, “Cyber-Emergencies: What Managers Can Learn From Complex Systems and Chaos Theory.,” vol. 3, Acapulco, Mexico: ACM, Jan. 2006.
- [41] D. Garrie and M. Simonova, “A Keystroke Causes a Tornado: Applying Chaos Theory to International Cyber Warfare Law,” *Brooklyn Journal of International Law*, vol. 45, no. 2, p. 497, Jun. 2020.
- [42] M. Gardner, “Mathematical games - the fantastic combinations of john conway’s new solitaire game life,” *Scientific American*, vol. 223, no. 4, pp. 120–123, 1970.
- [43] H. Qin, D. Liu, and J. Weng, “Cellular Automata Based Cyber Risk Conduction Mechanism of Cyber Physical Power Systems,” *2020 IEEE Sustainable Power and Energy Conference (iSPEC)*, pp. 1672–1677, 2020.
- [44] G. Cisotto and L. Badia, “Cyber security of smart grids modeled through epidemic models in cellular automata,” *2016 IEEE 17th International Symposium on A World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pp. 1–6, 2016.
- [45] J. Zhang, X. Xiong, Y. Wang, and J. Zhang, “Simulation Model for Cascading Failure in Complex Network: A Cellular Automata Approach,” *WSSE 2020: Proceedings of the 2020 The 2nd World Symposium on Software Engineering*, pp. 274–277, Sep. 2020. DOI: 10.1145/3425329.3425387.
- [46] K. Shi, J. Wang, S. Zhong, Y. Tang, and J. Cheng, “Hybrid-driven finite-time H sampling synchronization control for coupling memory complex networks with stochastic cyber attacks,” *Neurocomputing*, vol. 387, pp. 241–254, 2020. DOI: 10.1016/j.neucom.2020.01.022.

- [47] R. Pan, Y. Tan, D. Du, and S. Fei, “Adaptive event-triggered synchronization control for complex networks with quantization and cyber-attacks,” *Neurocomputing*, vol. 382, pp. 249–258, 2020.
- [48] D. Ionica, N. Popescu, D. Popescu, and F. Pop, “Cyber Defence Capabilities in Complex Networks,” in *Internet of Everything*, 2018. DOI: 10.1007/978-981-10-5861-5_10.
- [49] G. Wen, W. Yu, X. Yu, and J. Lu, “Complex cyber-physical networks: From cybersecurity to security control,” *J. Syst. Sci. Complex.*, vol. 30, pp. 46–67, 2017. DOI: 10.1007/s11424-017-6181-x.
- [50] C. R. Shalizi, “Methods and Techniques of Complex Systems Science: An Overview,” in *Complex Systems Science in Biomedicine*, ser. Topics in Biomedical Engineering International Book Series, T. Deisboeck and J. Y. Kresh, Eds., Boston, MA: Springer US, 2006, pp. 33–114.
- [51] T. Mary-Huard and S. Robin, “Introduction to Statistical Methods for Complex Systems,” in *Handbook of Statistical Systems Biology*, Hoboken, N.J: Wiley, 2011, pp. 15–38.
- [52] F. Razak and H. Jensen, “Quantifying ‘Causality’ in Complex Systems: Understanding Transfer Entropy,” *PloS one*, vol. 9, Jun. 2014.
- [53] A. Abbasi, Z. Zhang, D. Zimbra, H. Chen, and J. F. Nunamaker, “Detecting Fake Websites: The Contribution of Statistical Learning Theory,” *MIS Quarterly*, vol. 34, no. 3, pp. 435–461, 2010.
- [54] J. Kour, M. Hanmandlu, and A. Q. Ansari, “Biometrics in Cyber Security,” *Defence Science Journal*, vol. 66, no. 6, pp. 600–604, Oct. 2016.
- [55] J. Ashraf, M. Keshk, N. Moustafa, *et al.*, “IoTBoT-IDS: A novel statistical learning-enabled botnet detection framework for protecting networks of smart cities,” *Sustainable Cities and Society*, vol. 72, p. 103 041, Sep. 2021.
- [56] K. Pei, “Bridging statistical learning and formal reasoning for cyber attack detection,” Ph.D. dissertation, Purdue, 2016. [Online]. Available: https://docs.lib.purdue.edu/cgi/viewcontent.cgi?article=1857&context=open_access_theses.
- [57] S. de Marchi and S. E. Page, “Agent-Based Models,” *Annual Review of Political Science*, vol. 17, no. 1, pp. 1–20, 2014, <https://doi.org/10.1146/annurev-polisci-080812-191558>.

- [58] M. W. Macy and R. Willer, “From Factors to Actors: Computational Sociology and Agent-Based Modeling,” *Annual Review of Sociology*, vol. 28, no. 1, pp. 143–166, 2002.
- [59] A. M. El-Sayed, P. Scarborough, L. Seemann, and S. Galea, “Social network analysis and agent-based modeling in social epidemiology,” *Epidemiologic Perspectives & Innovations*, vol. 9, p. 1, Feb. 2012.
- [60] S. Galea, M. Riddle, and G. A. Kaplan, “Causal thinking and complex system approaches in epidemiology,” *International Journal of Epidemiology*, vol. 39, no. 1, pp. 97–106, Feb. 2010.
- [61] A. Troisi, V. Wong, and M. A. Ratner, “An agent-based approach for modeling molecular self-organization,” *Proceedings of the National Academy of Sciences*, vol. 102, no. 2, pp. 255–260, Jan. 2005.
- [62] CoMSES, *Computational model library*, 2021. [Online]. Available: <https://www.comses.net/codebases/> (visited on 09/20/2021).
- [63] J. Bates and H. K. Shepard, “Measuring complexity using information fluctuation,” *Physics Letters A*, vol. 172, pp. 416–425, 1993.
- [64] J. Bates, *Measuring complexity using information fluctuation: A tutorial*, Mar. 2020. [Online]. Available: https://www.researchgate.net/publication/340284677_Measuring_complexity_using_information_fluctuation_a_tutorial.
- [65] S. Xu, “Cybersecurity Dynamics: A Foundation for the Science of Cybersecurity,” in *Proactive and Dynamic Network Defense*, ser. Advances in Information Security 74, Springer, 2019, pp. 1–31.
- [66] S. Xu, “The cybersecurity dynamics way of thinking and landscape,” in *Proceedings of the 7th ACM Workshop on Moving Target Defense, 2020*, Dec. 2020.
- [67] S. Xu, “Cybersecurity dynamics,” in *Proceedings of the 2014 Symposium and Bootcamp on the Science of Security*, ser. HotSoS ’14, New York, NY, Apr. 2014, pp. 1–2.
- [68] I. Kotenkov, A. Konovalov, and A. Shorov, “Simulation of Botnets: Agent-Based Approach,” in *Intelligent Distributed Computing IV*, M. Essaaidi, M. Malgeri, and C. Badica, Eds., ser. Studies in Computational Intelligence, Berlin, Heidelberg: Springer, 2010, pp. 247–252.

- [69] B. Thompson and J. Morris-King, "An agent-based modeling framework for cybersecurity in mobile tactical networks," *The Journal of Defense Modeling and Simulation*, vol. 15, no. 2, pp. 205–218, Apr. 2018.
- [70] I. Kotenko, "Simulation of agent teams: Application of a domain independent framework to computer network security," *ECMS 2009*, Jun. 2009.
- [71] S. Chi, J. S. Park, K. C. Jung, and J. S. Lee, "Network Security Modeling and Cyber Attack Simulation Methodology," in *Information Security and Privacy*, V. Varadharajan and Y. Mu, Eds., Berlin, Heidelberg: Springer, 2001, pp. 320–333.
- [72] F. Cohen, "Simulating cyber attacks, defences, and consequences," *Computers & Security*, vol. 18, no. 6, pp. 479–518, Jan. 1999, ISSN: 0167-4048. DOI: 10.1016/S0167-4048(99)80115-1. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0167404899801151>.
- [73] F. Cohen, C. Phillips, L. P. Swiler, *et al.*, "A Preliminary Classification Scheme for Information System Threats, Attacks, and Defenses; A Cause and Effect Model; and Some Analysis Based on That Model," Sandia National Laboratories, Tech. Rep., Sep. 1998. [Online]. Available: <http://all.net/journal/ntb/cause-and-effect.html>.
- [74] I. Kotenko and E. Man'kov, "Experiments with Simulation of Attacks against Computer Networks," in *Computer Network Security*, V. Gorodetsky, L. Popyack, and V. Skormin, Eds., Berlin, Heidelberg: Springer, 2003, pp. 183–194.
- [75] I. Kotenko, "Teamwork of Hackers-Agents: Modeling and Simulation of Coordinated Distributed Attacks on Computer Networks," in *Multi-Agent Systems and Applications III*, V. Mařík, M. Pěchouček, and J. Müller, Eds., ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer, 2003, pp. 464–474.
- [76] I. Kotenko, "Agent-Based Modeling and Simulation of Cyber-Warfare between Malefactors and Security Agents in Internet," *ECMS 2005*, Jan. 2005.
- [77] I. Kotenko, "Multi-agent Modelling and Simulation of Cyber-Attacks and Cyber-Defense for Homeland Security," in *2007 4th IEEE Workshop on Intelligent Data Acquisition*, 2007, pp. 614–619.
- [78] A. M. Konovalov, I. V. Kotenko, and A. V. Shorov, "Simulation-based study of botnets and defense mechanisms against them," *Journal of Computer and Systems Sciences International*, vol. 52, no. 1, pp. 43–65, Jan. 2013.
- [79] K. Sycara and M. Lewis, "Agent-based Approaches to Dynamic Team Simulation," *Navy Personnel Research, Studies, and Technology Division Bureau of Naval*

Personnel, 2008. [Online]. Available: <https://apps.dtic.mil/sti/pdfs/ADA487741.pdf>.

- [80] P. Rajivan, M. A. Janssen, and N. J. Cooke, “Agent-Based Model of a Cyber Security Defense Analyst Team,” *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 57, no. 1, pp. 314–318, Sep. 2013.
- [81] D. Grunewald, M. Lützenberger, J. Chinnow, R. Bye, K. Bsufka, and S. Albayrak, “Agent-based Network Security Simulation,” Taipei, Taiwan, 2011.
- [82] Y. Zhao, Y. Wang, H. Zhang, C. Zhang, and C. Yang, “Agent-based Network Security Simulator Nessi2 | Semantic Scholar,” Qingdao, China, 2015.
- [83] A. Kosowski and V. Mosorov, “Nessi2 simulator for large-scale DDoS attack analysis,” *Perspective Technologies and Methods in MEMS Design*, 2011.
- [84] M. D. Norman and M. T. Koehler, “Cyber Defense as a Complex Adaptive System: A model-based approach to strategic policy design,” in *CSS 2017: CSSSA*, Santa Fe NM: ACM, Oct. 2017.
- [85] *NetLogo Home Page*, 2016. [Online]. Available: <http://ccl.northwestern.edu/netlogo/> (visited on 08/16/2022).
- [86] *Repast Suite Documentation*, 2021. [Online]. Available: <https://repast.github.io/index.html> (visited on 08/16/2022).
- [87] *StarLogo Nova*, 2022. [Online]. Available: <https://www.slnova.org/> (visited on 08/16/2022).
- [88] J. L. Nolen, *Bobo doll experiment*, May 2020. [Online]. Available: <https://www.britannica.com/event/Bobo-doll-experiment> (visited on 02/13/2022).
- [89] A. Bandura, “Toward a Psychology of Human Agency,” *Perspectives on Psychological Science*, vol. 1, no. 2, pp. 164–180, Jun. 2006, Publisher: SAGE Publications Inc, ISSN: 1745-6916. [Online]. Available: <https://doi.org/10.1111/j.1745-6916.2006.00011.x>.
- [90] E. Alqurashi, “Self-Efficacy In Online Learning Environments: A Literature Review,” *CIER*, vol. 9, pp. 45–52, 2016. DOI: 10.19030/CIER.V9I1.9549.
- [91] P. Sheeran, A. Maki, E. Montanaro, *et al.*, “The impact of changing attitudes, norms, and self-efficacy on health-related intentions and behavior: A meta-analysis,” *Health psychology : official journal of the Division of Health Psychology, American Psychological Association*, 2016. DOI: 10.1037/HEA0000387.

- [92] L. Blackburn and G. P. Owens, "The effect of self efficacy and meaning in life on posttraumatic stress disorder and depression severity among veterans," *Journal of Clinical Psychology*, vol. 71, no. 3, pp. 219–228, Mar. 2015, ISSN: 1097-4679. DOI: 10.1002/jclp.22133.
- [93] B. Becote, "Construction and Validation of the First Aid Initial Response Self-Efficacy Assessment," New York, NY: Association for Psychological Science, 2006.
- [94] M. S. Cardon and C. P. Kirk, "Entrepreneurial Passion as Mediator of the Self Efficacy to Persistence Relationship," *Entrepreneurship Theory and Practice*, vol. 39, pp. 1027–1050, 5 2015. DOI: 10.1111/etap.12089.
- [95] P. Piperopoulos and D. Dimov, "Burst Bubbles or Build Steam Entrepreneurship Education, Entrepreneurial Self-Efficacy, and Entrepreneurial Intentions," *Journal of Small Business Management*, vol. 53, pp. 970–985, 4 2015. DOI: 10.1111/jsbm.12116.
- [96] S. N. Sweet, M. S. Fortier, S. M. Strachan, and C. M. Blanchard, "Testing and integrating self-determination theory and self-efficacy theory in a physical activity context.," *Canadian Psychology/Psychologie canadienne*, vol. 53, no. 4, pp. 319–327, 2012.
- [97] A. Bandura, "Guide for Constructing Self-Efficacy Scales," in *Self-Efficacy of Adolescents*, Greenwich, Conn: Information Age Publishing, Feb. 2006, pp. 307–337.
- [98] N. C. Zainal, M. Puad, and N. F. M. Sani, "Moderating Effect of Self-Efficacy in the Relationship Between Knowledge, Attitude and Environment Behavior of Cybersecurity Awareness," *Asian Social Science*, 2021.
- [99] T. Chen, M. Stewart, Z. Bai, E. Chen, L. A. Dabbish, and J. Hammer, "Hacked Time: Design and Evaluation of a Self-Efficacy Based Cybersecurity Game," *Conference on Designing Interactive Systems*, 2020. DOI: 10.1145/3357236.3395522.
- [100] W. He, X. Yuan, and X. Tian, "The Self-Efficacy Variable in Behavioral Information Security Research," *Proceedings - 2nd International Conference on Enterprise Systems, ES 2014*, pp. 28–32, Dec. 2014. DOI: 10.1109/ES.2014.52.
- [101] V. Sheanoda and K. Bussey, "Victims of Cyberbullying: An Examination of Social Cognitive Processes Associated with Cyberbullying Victimization," *Journal of School Violence*, 2021. DOI: 10.1080/15388220.2021.1984933.
- [102] A. Peker, Y. Eroğlu, and M. N. Yıldız, "Does High Self-Efficacy in Adolescents Minimize Cyber Bullying Behaviour?," 2021. DOI: 10.33808/CLINEXPHEALTHSCI.864038.

- [103] T. Y. Bingöl, “Determining The Predictors of Self-Efficacy and Cyber Bullying,” *International Journal of Higher Education*, vol. 7, no. 2, p. 138, Mar. 2018, Number: 2.
- [104] K. Jeon and Y.-W. Kim, “The effect of self-expression, emotion regulation, and self-efficacy on burnout of cyber university students,” *Journal of Next-generation Convergence Information Services Technology*, 2021. DOI: 10.29056/jncist.2021.08.05.
- [105] A. Konak, “Experiential Learning Builds Cybersecurity Self-Efficacy in K-12 Students,” *Journal of Cyber Education, Research and Practice*, p. 16, 1 2018.
- [106] R. Van der Kleij, G. Kleinhuis, and H. Young, “Computer Security Incident Response Team Effectiveness: A Needs Assessment,” *Frontiers in Psychology*, vol. 8, p. 2179, Dec. 2017.
- [107] J. Kruger and D. Dunning, “Unskilled and unaware of it: How difficulties in recognizing one’s own incompetence lead to inflated self-assessments,” *Journal of Personality and Social Psychology*, vol. 77, no. 6, pp. 1121–1134, Dec. 1999.
- [108] G. O. Boateng, T. B. Neilands, E. A. Frongillo, H. R. Melgar-Quinonez, and S. L. Young, “Best Practices for Developing and Validating Scales for Health, Social, and Behavioral Research: A Primer,” *Frontiers in Public Health*, vol. 6, p. 149, Jun. 2018.
- [109] K. Jansen, K. Corley, and J. Jansen, “E-Survey Methodology,” *Pennsylvania State College of Information Sciences and Technology*. Retrieved on March 1, 2010, from, Jan. 2007. DOI: 10.4018/978-1-59140-792-8.ch001.
- [110] R. N. Davis, “Web-based administration of a personality questionnaire: Comparison with traditional methods,” *Behavior Research Methods, Instruments, & Computers*, vol. 31, no. 4, pp. 572–577, Dec. 1999.
- [111] J. F. Ebert, L. Huibers, B. Christensen, and M. B. Christensen, “Paper- or Web-Based Questionnaire Invitations as a Method for Data Collection: Cross-Sectional Comparative Study of Differences in Response Rate, Completeness of Data, and Financial Cost,” *Journal of Medical Internet Research*, vol. 20, no. 1, e24, Jan. 2018. (visited on 05/02/2022).
- [112] J. D. Smyth, “Internet survey methods: A review of strengths, weaknesses, and innovations,” in *Social and behavioral research and the internet*, New York, NY: Routledge, 2011.

- [113] A. Zapf, S. Castell, L. Morawietz, and A. Karch, “Measuring inter-rater reliability for nominal data – which coefficients and confidence intervals are appropriate?” *BMC Medical Research Methodology*, vol. 16, no. 1, p. 93, Aug. 2016.
- [114] A. F. Hayes and K. Krippendorff, “Answering the Call for a Standard Reliability Measure for Coding Data,” *Communication Methods and Measures*, vol. 1, no. 1, pp. 77–89, Apr. 2007.
- [115] A. D. Hayes, *SPSS, SAS, and R Macros and Code*, 2022. [Online]. Available: <http://afhayes.com/spss-sas-and-r-macros-and-code.html> (visited on 12/20/2022).
- [116] S. R. Briggs and J. M. Cheek, “The role of factor analysis in the development and evaluation of personality scales,” *Journal of Personality*, vol. 54, no. 1, pp. 106–148, Mar. 1986.
- [117] T. Raykov and G. A. Marcoulides, *Introduction to Psychometric Theory*, 1st edition. New York, NY: Routledge, Sep. 2010, ISBN: 978-0-415-87822-7.
- [118] L. Crocker and J. Algina, *Introduction to Classical and Modern Test Theory*. Holt, Rinehart and Winston, 1986, ISBN: 978-0-03-061634-1.
- [119] R. Jabrayilov, W. H. M. Emons, and K. Sijtsma, “Comparison of Classical Test Theory and Item Response Theory in Individual Change Assessment,” *Applied Psychological Measurement*, vol. 40, no. 8, pp. 559–572, Nov. 2016.
- [120] R. H. Hoyle and A. L. Dent, “Developmental trajectories of skills and abilities relevant for self-regulation of learning and performance,” in *Handbook of self-regulation of learning and performance*, 2nd ed, ser. Educational psychology handbook series, New York, NY, US: Routledge/Taylor & Francis Group, 2018, pp. 49–63.
- [121] D. J. Bodeau, R. D. Graubart, R. McQuaid, and J. R. W. Jr, “Cyber Resiliency Metrics, Measures of Effectiveness, and Scoring,” Aug. 2019.
- [122] F. Zhang and B. Bu, “A Cyber Security Risk Assessment Methodology for CBTC Systems Based on Complex Network Theory and Attack Graph,” *2021 7th Annual International Conference on Network and Information Systems for Computers (IC-NISC)*, 2021.
- [123] E. Hutchins, M. Cloppert, and R. Amin, “Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains,” *Leading Issues in Information Warfare & Security Research*, vol. 1, Jan. 2011.
- [124] D. of Homeland Security, *NCCIC ICS Cybersecurity Evaluation Tool*, 2022. [Online]. Available: <https://www.cisa.gov/uscert/ics/Assessments>.

- [125] S. Zavala, N. Shashidhar, and C. Varol, “Cybersecurity Evaluation with PowerShell,” in *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, Beirut, Lebanon: IEEE, Jun. 2020, pp. 1–6.
- [126] R. C. Wilson and A. G. Collins, “Ten simple rules for the computational modeling of behavioral data,” *eLife*, vol. 8, T. E. Behrens, Ed., p. 49 547, Nov. 2019.
- [127] K. Kording, G. Blohm, P. Schrater, and K. Kay, “Appreciating diversity of goals in computational neuroscience,” en-us, OSF Preprints, Tech. Rep., Sep. 2018, type: article.
- [128] R. J. Wieringa, *Design Science Methodology for Information Systems and Software Engineering*, 2014th ed. Springer, Nov. 2014.
- [129] M. Granasen and D. Granåsen, “Measuring team effectiveness in cyber-defense exercises: A cross-disciplinary case study,” *Cognition, Technology & Work*, vol. 18, Feb. 2016. DOI: 10.1007/s10111-015-0350-2.
- [130] C. Wohlin and P. Runeson, “Guiding the selection of research methodology in industry–academia collaboration in software engineering,” *Information and Software Technology*, vol. 140, p. 106 678, Dec. 2021.
- [131] R. Wieringa and A. Morah, “Technical Action Research as a Validation Method in Information Systems Design Science,” in *Design Science Research in Information Systems. Advances in Theory and Practice*, K. Peffers, M. Rothenberger, and B. Kuechler, Eds., ser. Lecture Notes in Computer Science, Berlin, Heidelberg: Springer, 2012, pp. 220–238, ISBN: 978-3-642-29863-9. DOI: 10.1007/978-3-642-29863-9_17.
- [132] M. Janssen and E. Ostrom, “Empirically Based, Agent-based models,” *Ecology and Society*, vol. 11, Dec. 2006. DOI: 10.5751/ES-01861-110237.
- [133] U. Doncaster, *Doncaster: Managing a cyber attack | Local Government Association*, 2018. [Online]. Available: <https://www.local.gov.uk/case-studies/doncaster-managing-cyber-attack>.
- [134] D. of Energy, “CyOTE Case Study: Darkside,” *Energy Security*, Feb. 2022.
- [135] B. Schmidt and Schneider, “Agent-based Modelling of Human Acting, Deciding and Behaviour-The Reference Model PECS,” Jan. 2004.
- [136] G. ten Broeke, G. van Voorn, and A. Ligtenberg, “Which Sensitivity Analysis Method Should I Use for My Agent-Based Model?” *Journal of Artificial Societies and Social Simulation*, vol. 19, no. 1, p. 5, 2016, ISSN: 1460-7425.

- [137] E. Borgonovo, M. Pangallo, J. Rivkin, L. Rizzo, and N. Siggelkow, “Sensitivity analysis of agent-based models: A new protocol,” *Computational and Mathematical Organization Theory*, vol. 28, no. 1, pp. 52–94, Mar. 2022.
- [138] J. Kazil, D. Masad, A. Crooks, *et al.*, “Utilizing Python for Agent-Based Modeling: The Mesa Framework,” in *Social, Cultural, and Behavioral Modeling*, Springer International Publishing, 2020, pp. 308–317.
- [139] M. W. Watkins, “Exploratory Factor Analysis: A Guide to Best Practice,” *Journal of Black Psychology*, vol. 44, no. 3, pp. 219–246, Apr. 2018, Publisher: SAGE Publications Inc.
- [140] M. Tavakol and R. Dennick, “Making sense of Cronbach’s alpha,” *International Journal of Medical Education*, vol. 2, pp. 53–55, Jun. 2011.
- [141] G. Qu, J. Rudraraju, R. Modukuri, S. Hariri, and C. Raghavendra, “A Framework for Network Vulnerability Analysis,” Jan. 2002, pp. 289–294.
- [142] J. Carifio and R. Perla, “Resolving the 50-year Debate Around using and Misusing Likert Scales,” *Medical education*, vol. 42, pp. 1150–2, Jan. 2009. DOI: 10.1111/j.1365-2923.2008.03172.x.
- [143] G. Norman, “Likert scales, levels of measurement and the “laws” of statistics,” *Advances in health sciences education : theory and practice*, vol. 15, pp. 625–32, Feb. 2010. DOI: 10.1007/s10459-010-9222-y.

Appendix A

Dakota State University IRB

Approval Letter



Institutional Review Board

DAKOTA STATE UNIVERSITY

820 N. Washington Ave

Madison, SD 57042

Expedited Review Determination

Date: 10/07/2022
To: Dr. Bhaskar Rimal and Briant Bocote
Approval #: 20221007

Dear : Dr. Rimal and Mr. Becote

The Dakota State University IRB has conducted an expedited review, in accordance with federal requirements under 45 CFR 46.110, of your project and approved it on 10/07/2022. This approval was based on your project's meeting the condition of: *Research that only includes no more than minimal risk to participants.*

To maintain its approved status, your research must be conducted according to the most recent plan reviewed by the IRB. You must notify the IRB in writing within **four** days of:

- Any changes to your research plan or departure from its description as stated in your application and/or other documents submitted;
- Any unexpected or adverse event that occurs in relation to your research project.

Within 364 days of the date of this letter, you must submit:

- A notice of closure once all project activities have concluded;
-- or --
- An application for extension of time to complete your research.

If you have any questions regarding this determination or during your study, please contact us at 605-256-5100 or irb@dsu.edu. Best wishes to you and your research.

Best Regards,

DocuSigned by:

932B2D779E0A4AF...

Stacey Berry, Chair

Appendix B

Research Participant Consent Form

Participant Consent Form

Construction, Reliability, and Validation of the Cyber Operations Self-Efficacy Scale (COSES)

The purpose of this research is to develop a measurement for the self-efficacy (confidence) of a cyber operations student or professional in accomplishing applicable cyber-related tasks. Your feedback will help to determine which questions are best suited to accomplish this. While demographic data is used to ensure comprehensive validation and reliability through the initial development of this research tool, your personal feedback will remain anonymous and confidential.

Initials

- ☐ • I voluntarily agree to participate in this research study.
- ☐ • I understand that even if I agree to participate now, I can withdraw at any time or refuse to answer any question without any consequences.
- ☐ • I understand that I can withdraw permission to use data from my interview within seven days after any participation, in which case the material will be deleted.
- ☐ • I have had the purpose and nature of the study explained to me in writing and I have had the opportunity to ask questions about the study.
- ☐ • I understand that participation involves answering questions regarding cyber operations tasks.
- ☐ • I understand that I will not be compensated for participating in this research.
- ☐ • I understand that all information I provide for this study will be treated confidentially.
- ☐ • I understand that in any report on the results of this research my identity will remain anonymous.
- ☐ • I understand that results from this research will be used for academic purposes and may be published in a research journal, presented at a research conference, and may be used to complete academic research requirements.
- ☐ • I understand that if I inform the researcher that myself or someone else is at risk of harm, they may have to report this to the relevant authorities, possibly without my permission.
- ☐ • I understand that signed consent forms will be retained digitally for a period of three years.

- ☐ • I understand that I am entitled to access the information I have provided at any time while it is in storage as specified above.
- ☐ • I understand that I am free to contact the researchers or the Dean of Graduate Studies for Dakota State University to seek further clarification or information.

Signature of research participant

Date

Researchers

Briant Becote, MS, PMP
briant.becote@trojans.dsu.edu

Bhaskar Rimal, PhD, SMIEEE
Bhaskar.rimal@dsu.edu

Dean of Graduate Studies, Dakota State University

Mark Hawkes, PhD
mark.hawkes@dsu.edu

Appendix C

Expert Evaluation & Content Validity Tool

Expert Analysis

Cyber Operations Self-Efficacy Scale (COSES)

While the COSES is a self-administered assessment, it should be conducted with the following guidance in mind:

This scale is designed to measure the self-efficacy of cyber operations students and professionals; it is not designed for assessing the untrained layperson. While developed as an integral part of the Cyber Operations Performance Framework, it's independent use is encouraged to inform and guide strategies to improve cyber operations learning and training within the classroom and organizational settings. Cyber offensive and defensive items are presented throughout the questionnaire. Results are characterized by these two primary domains across three separate phases of cyber operations: preparation, intrusion, and active breach.

The COSES is not a knowledge assessment, it's a self-efficacy measure to define user confidence in achieving cyber operations success. Users are encouraged to answer honestly regarding their confidence in their abilities as the results can provide important feedback for improving academic and organizational cyber success.

The following set of demographic questions is not part of the Cyber Operations Self-Efficacy Scale. It is included to ensure comprehensive item evaluation during scale development, validity, and reliability testing and will be used to report findings while ensuring individual anonymity.

Mark the applicable box.

1. Your age group.

18-24	25-30	31-40	41-50	51-64	65+
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2. Your gender.

Male	Female	Them/Theirs
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

3. Your current education.

H. S. Degree	Some Undergrad	Undergrad Complete	MA/MS Complete	PhD Complete
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

4. Indicate if are currently in a program or have a degree majoring in IT, Computer Science, Cybersecurity or other similar programs. Mark all applicable boxes.

Undergrad	Masters	PhD
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5. Your industry certifications. Mark all applicable boxes.

None	Cybersecurity	Cyber Offense
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

4. Your current cybersecurity work experience.

< 3 months	4-11 months	1-2 years	3-5 years	6-10 years	11+ years
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Expert Analysis Guidance

The COSES is a five-point Likert scale to determine the self-efficacy of a cyber professional in conducting cyber operations. To determine content validity and interrater reliability, the preliminary scale has been modified to provide the means to standardize and quantify expert feedback. For context, the standard scale appears as such:

1. I can identify a fake website phishing for information.

I cannot do this						I can do this easily
-------------------------	--	--	--	--	--	-----------------------------

The standard format has been modified so that your responses can be evaluated statistically against other experts. The scale is designed to measure one's confidence in completing cybersecurity and cyber offensive tasks. Some tasks are exclusively one or the other, some have overlap in both domains. Your responses are expected to reflect that.

A blank line has also been provided for open-ended feedback. Please take note to identify any issues with item clarity, brevity, applicability, content completeness, and structure (for example, avoiding questions that ask one's confidence to achieve multiple tasks or leads to a particular answer).

The instructions (passages throughout the assessment) remain to offer you context while evaluating the scale's items.

If you find that two items ask the same question, please make note of that in the available space for feedback below each question.

You are assigned a new role at work, responsible for assessing the attack and defensive strategies of local organizations. To determine your first assignment, you are asked to complete the following questionnaire. You want to provide a good impression, but you recognize that accurately reporting your skills and capabilities will lead to accurate expectations.

These local organizations have a wide range of cybersecurity capabilities. Some are highly sophisticated while others have no cybersecurity program at all. You're asked to indicate your confidence.

Mark the box that represents on a sliding scale how confident you are in completing the presented task.

1. I can identify a fake website phishing for information.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

2. I can update system or network patches based on vulnerabilities published to the internet.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

3. I can identify vulnerabilities with regards to safeguarding sensitive data practices on computer systems or networks.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

4. I can conduct a cybersecurity assessment to identify vulnerabilities on a system's threat surface.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

5. I can conduct active scans of network traffic and identify potential threats.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

6. I can configure firewall settings to limit inbound system access while allowing appropriate outbound access to the internet.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

7. I can use network sniffing to identify configuration or administrative data from a target network.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

8. I can create a phishing campaign to actively collect sensitive data from a target organization.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

9. I can use tools and techniques to extract system characteristics from a target system.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

Following weeks of preparation, the offensive team has completed their initial preparation phrase. An intrusion on a local organization is expected to occur next. You're asked to indicate your confidence of the following:

10. I can execute command scripts to manipulate system processes.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

11. I can evade debugger detection.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

12. I can modify directory permissions to allow access to protected files.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

13. I can use a computer's peripheral devices to capture video or audio and gain access to sensitive information.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

14. I can identify suspicious program execution through system process analysis.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

15. I can identify a malicious browser extension redirecting traffic.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

16. I can correct administrative privilege abuses.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

17. I can identify changes to the boot records or BIOS.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

The offensive team has breached the target network and is executing collection and exploitation techniques. You're asked to indicate your confidence of the following:

18. I can use remote access software to establish command and control of a targeted system across a network.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

19. I can leverage operating system software known vulnerabilities or the kernel to execute malicious code.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

20. I can leverage obfuscation techniques to conceal command and control traffic.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

21. I can wipe or corrupt raw disk data on a target system to interrupt availability of system resources.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

22. I can identify direct access read/write attempts.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

23. I can identify unusual driver installation activity.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

24. I can determine the source of software exploits used against a network or system. [AB-CS-2]

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

25. I can identify network traffic originating from unknown hardware devices.

The item reflects a key cybersecurity capability.						
Absolutely Not						Absolutely
The item reflects a key cyber offense capability.						
Absolutely Not						Absolutely
The item is clear and to the point.						
Absolutely Not						Absolutely

Additional Expert Feedback:

Please indicate if you feel this broadly represents the domains of cybersecurity and cyber offense. While working to maintain a scope specific to organizational cyber operations, are there any specific tasks you would add? Any additional comments or feedback would be greatly appreciated.

Your time and expertise is high valued, thank you for supporting my research.

Appendix D

Cyber Operations Self-Efficacy Scales

Preliminary Cyber Operations Self-Efficacy Scale (COSES)

While the COSES is a self-administered assessment, it should be conducted with the following guidance in mind:

This scale is designed to measure the self-efficacy of cyber operations students and professionals; it is not designed for assessing the untrained layperson. While developed as an integral part of the Cyber Operations Performance Framework, it's independent use is encouraged to inform and guide strategies to improve cyber operations learning and training within the classroom and organizational settings. Cyber offensive and defensive items are presented in two separate surveys and participants should take the survey that aligns with their cyber operation goals. Each scale is presented in a series of three cyber operational phases: preparation, intrusion, and active breach.

The COSES is not a knowledge assessment, it's a self-efficacy measure to define user confidence in achieving cyber operations success. Users are encouraged to answer honestly regarding their confidence in their abilities as the results can provide important feedback for improving academic and organizational cyber success.

The following set of demographic questions is not part of the Cyber Operations Self-Efficacy Scale. It is included to ensure comprehensive item evaluation during scale development, validity, and reliability testing and will be used to report findings while ensuring individual anonymity.

Mark the applicable box.

1. Your age group.

18-24	25-30	31-40	41-50	51-64	65+
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2. Your gender.

Male	Female	Them/Theirs
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

3. Your current education.

H. S. Degree	Some Undergrad	Undergrad Complete	MA/MS Complete	PhD Complete
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

4. Indicate if are currently in a program or have a degree majoring in IT, Computer Science, Cybersecurity or other similar programs. Mark all applicable boxes.

Undergrad	Masters	PhD
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5. Your industry certifications. Mark all applicable boxes.

None	Cybersecurity	Cyber Offense
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

4. Your current cybersecurity work experience.

< 3 months	4-11 months	1-2 years	3-5 years	6-10 years	11+ years
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Cybersecurity Cyber Operations Self-Efficacy Scale

You are assigned a new role at work, responsible for assessing the attack and defensive strategies of local organizations. To determine your first assignment, you are asked to complete the following questionnaire. You want to provide a good impression, but you recognize that accurately reporting your skills and capabilities will lead to accurate expectations.

These local organizations have a wide range of cybersecurity capabilities. Some are highly sophisticated while others have no cybersecurity program at all. You're asked to indicate your confidence.

Mark the box that represents on a sliding scale of 1 to 10 how confident you are in completing the presented task.

1. I can identify a fake website phishing for information.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2. I can update system or network patches based on vulnerabilities published to the internet.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

3. I can identify vulnerabilities with regards to safeguarding sensitive data practices on computer systems or networks.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

4. I can conduct active scans of network traffic and identify potential threats.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5. I can configure firewall settings to limit inbound system access while allowing appropriate outbound access to the internet.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Despite ongoing efforts, indications indicate that an intrusion is likely to occur. You're asked to indicate your confidence of the following:

6. I can identify suspicious program execution through system process analysis.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

7. I can identify a malicious browser extension redirecting traffic.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

8. I can correct administrative privilege abuses.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

9. I can identify changes to the boot records or BIOS.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

The offensive team has breached the target network and is executing collection and exploitation techniques. You're asked to indicate your confidence of the following:

10. I can identify unusual driver installation activity.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

11. I can determine the source of software exploits used against a network or system. [AB-CS-2]

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Cyber Offense Cyber Operations Self-Efficacy Scale

You are assigned a new role at work, responsible for conducting cyber-attacks against potential threat organizations. To determine your first assignment, you are asked to complete the following questionnaire. You want to provide a good impression, but you recognize that accurately reporting your skills and capabilities will lead to accurate expectations.

These target organizations have a wide range of cybersecurity capabilities. Some are highly sophisticated while others have no cybersecurity program at all. You're asked to indicate your confidence.

Mark the box that represents on a sliding scale of 1 to 7 how confident you are in completing the presented task.

1. I can identify vulnerabilities with regards to safeguarding sensitive data practices on computer systems or networks.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

2. I can create a phishing campaign to actively collect sensitive data from a target organization.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

3. I can use tools and techniques to extract system characteristics from a target system.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Following weeks of preparation, your team has completed their initial preparation phrase. Intrusion is expected to begin soon. You're asked to indicate your confidence of the following:

4. I can execute command scripts to manipulate system processes.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

5. I can evade debugger detection.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

6. I can modify directory permissions to allow access to protected files.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

7. I can use a computer's peripheral devices to capture video or audio and gain access to sensitive information.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

After multiple attempts, you have infiltrated the target computer. As directed, you now are expected to manage command and control operations while executing exploitation. You're asked to indicate your confidence of the following:

8. I can leverage operating system software known vulnerabilities or the kernel to execute malicious code.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

9. I can wipe or corrupt raw disk data on a target system to interrupt availability of system resources.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

10. I can identify direct access read/write attempts.

	1. Can't do this	2	3	4	5. Unsure Either way	6	7.	8.	9.	10. Can absolutely do this
How Confident are you?	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Appendix E

Model Sensitivity Statistical Analysis Results

Node Count MANOVA Test Results

Between-Subjects Factors

	N
node_count 1	500
6	500
11	500
16	500
21	500
26	500
31	500
36	500
41	500
46	500
51	500
56	500
61	500
66	500
71	500
76	500
81	500
86	500
91	500
96	500
101	500

Estimated Marginal Means

1. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.302	.001	.301	.303
SustainabilityIndex	253.956	1.104	251.791	256.120

Multivariate Tests^a

Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.988	431171.908 ^b	2.000	10478.000	<.001
	Wilks' Lambda	.012	431171.908 ^b	2.000	10478.000	<.001
	Hotelling's Trace	82.300	431171.908 ^b	2.000	10478.000	<.001
	Roy's Largest Root	82.300	431171.908 ^b	2.000	10478.000	<.001
node_count	Pillai's Trace	1.421	1284.498	40.000	20958.000	<.001
	Wilks' Lambda	.038	2150.597 ^b	40.000	20956.000	<.001
	Hotelling's Trace	13.101	3431.438	40.000	20954.000	<.001
	Roy's Largest Root	12.114	6346.880 ^c	20.000	10479.000	<.001

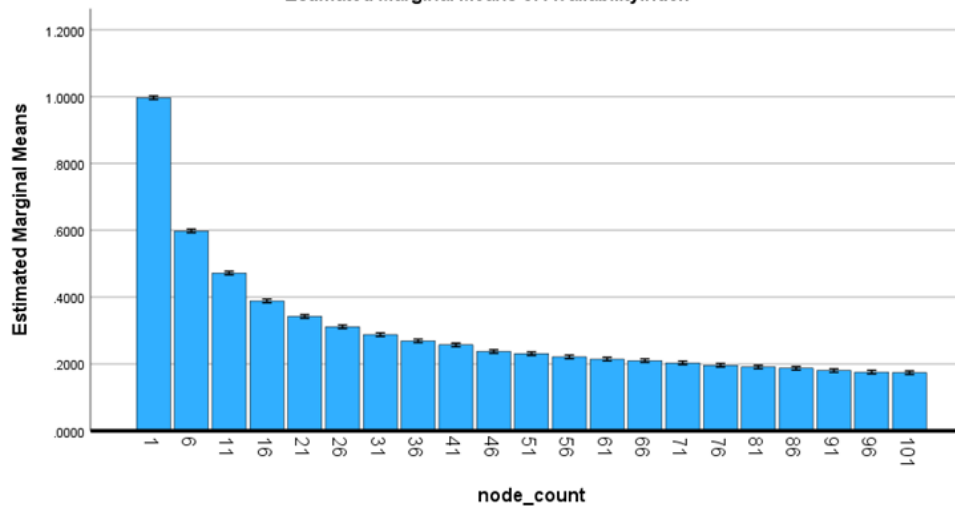
a. Design: Intercept + node_count

b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

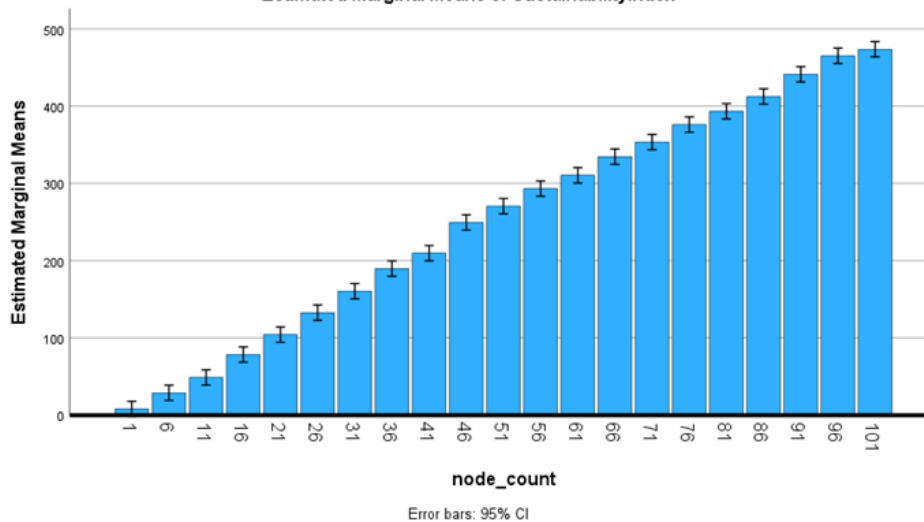
AvailabilityIndex

Estimated Marginal Means of AvailabilityIndex



SustainabilityIndex

Estimated Marginal Means of SustainabilityIndex



Security Strength MANOVA Test Results

Between-Subjects Factors

		N
security_strength	1	500
	6	500
	11	500
	16	500
	21	500
	26	500
	31	500
	36	500
	41	500
	46	500
	51	500
	56	500
	61	500
	66	500
	71	500
	76	500
	81	500
	86	500
	91	500
	96	500
	101	500

Estimated Marginal Means

1. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.396	.001	.395	.398
SustainabilityIndex	71.754	.273	71.219	72.289

Multivariate Tests^a

Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.990	526338.034 ^b	2.000	10478.000	<.001
	Wilks' Lambda	.010	526338.034 ^b	2.000	10478.000	<.001
	Hotelling's Trace	100.465	526338.034 ^b	2.000	10478.000	<.001
	Roy's Largest Root	100.465	526338.034 ^b	2.000	10478.000	<.001
security_strength	Pillai's Trace	.328	102.782	40.000	20958.000	<.001
	Wilks' Lambda	.673	114.501 ^b	40.000	20956.000	<.001
	Hotelling's Trace	.483	126.438	40.000	20954.000	<.001
	Roy's Largest Root	.478	250.564 ^c	20.000	10479.000	<.001

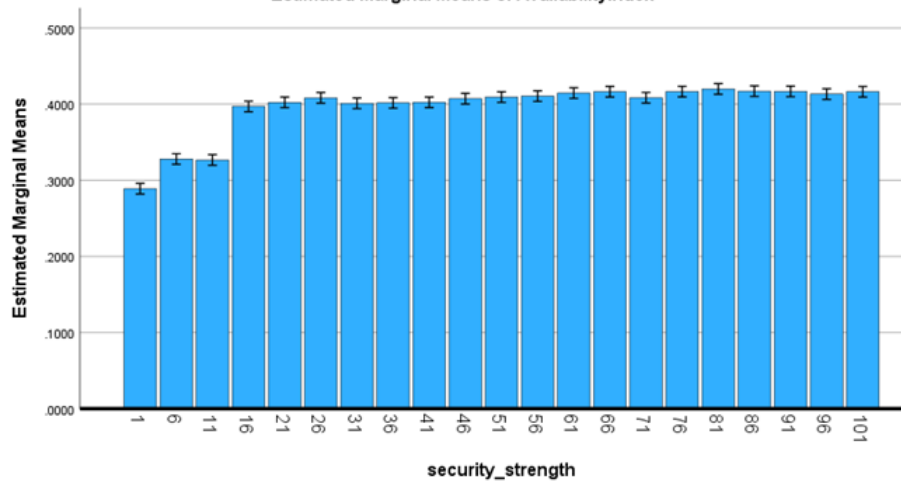
a. Design: Intercept + security_strength

b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

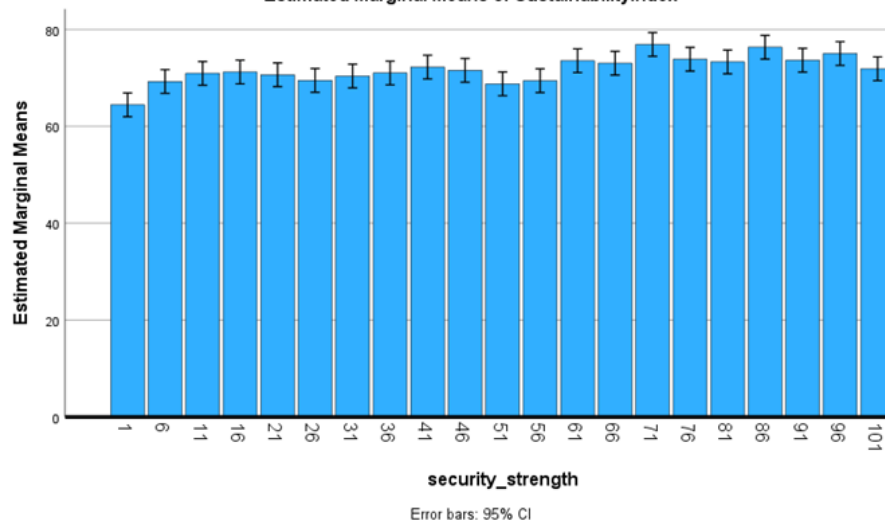
AvailabilityIndex

Estimated Marginal Means of AvailabilityIndex



SustainabilityIndex

Estimated Marginal Means of SustainabilityIndex



Friendly Count MANOVA Test Results

Between-Subjects Factors

	N
friendly_count 1	500
6	500
11	500
16	500
21	500
26	500
31	500
36	500
41	500
46	500
51	500
56	500
61	500
66	500
71	500
76	500
81	500
86	500
91	500
96	500
101	500

Estimated Marginal Means

1. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.859	.000	.858	.859
SustainabilityIndex	935.462	.649	934.189	936.734

Multivariate Tests^a

Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.999	9946272.316 ^b	2.000	10478.000	<.001
	Wilks' Lambda	.001	9946272.316 ^b	2.000	10478.000	<.001
	Hotelling's Trace	1898.506	9946272.316 ^b	2.000	10478.000	<.001
	Roy's Largest Root	1898.506	9946272.316 ^b	2.000	10478.000	<.001
friendly_count	Pillai's Trace	1.466	1439.837	40.000	20958.000	<.001
	Wilks' Lambda	.008	5252.911 ^b	40.000	20956.000	<.001
	Hotelling's Trace	62.879	16469.613	40.000	20954.000	<.001
	Roy's Largest Root	61.948	32457.442 ^c	20.000	10479.000	<.001

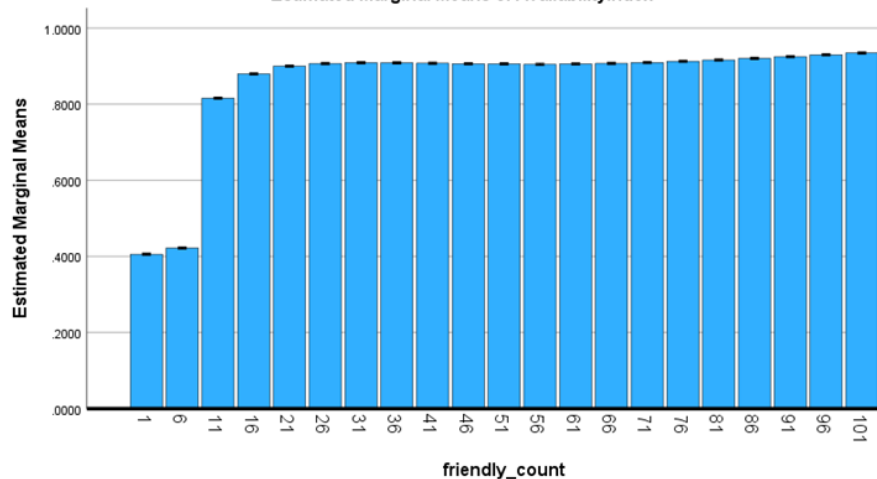
a. Design: Intercept + friendly_count

b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

AvailabilityIndex

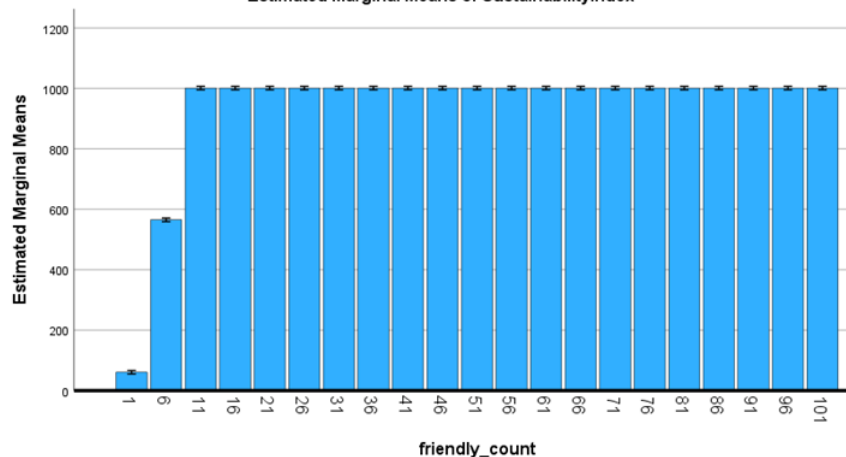
Estimated Marginal Means of AvailabilityIndex



Error bars: 95% CI

SustainabilityIndex

Estimated Marginal Means of SustainabilityIndex



Error bars: 95% CI

Friendly Efficacy MANOVA Test Results

Between-Subjects Factors

	N
friendly_efficacy 1	500
2	500
3	500
4	500
5	500
6	500
7	500
8	500
9	500
10	500

2. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.407	.001	.404	.409
SustainabilityIndex	69.554	.378	68.814	70.295

Multivariate Tests^a

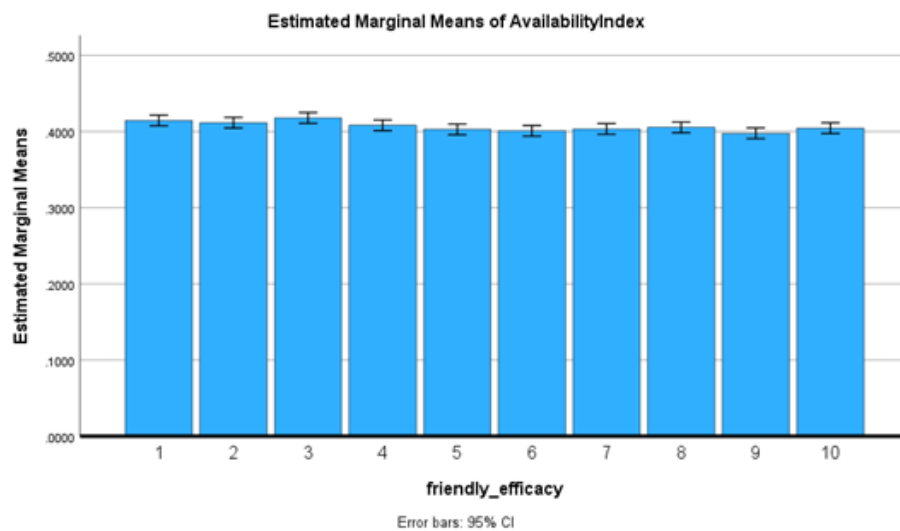
Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.991	276356.103 ^b	2.000	4989.000	<.001
	Wilks' Lambda	.009	276356.103 ^b	2.000	4989.000	<.001
	Hotelling's Trace	110.786	276356.103 ^b	2.000	4989.000	<.001
	Roy's Largest Root	110.786	276356.103 ^b	2.000	4989.000	<.001
friendly_efficacy	Pillai's Trace	.021	5.991	18.000	9980.000	<.001
	Wilks' Lambda	.979	6.013 ^b	18.000	9978.000	<.001
	Hotelling's Trace	.022	6.036	18.000	9976.000	<.001
	Roy's Largest Root	.020	11.174 ^c	9.000	4990.000	<.001

a. Design: Intercept + friendly_efficacy

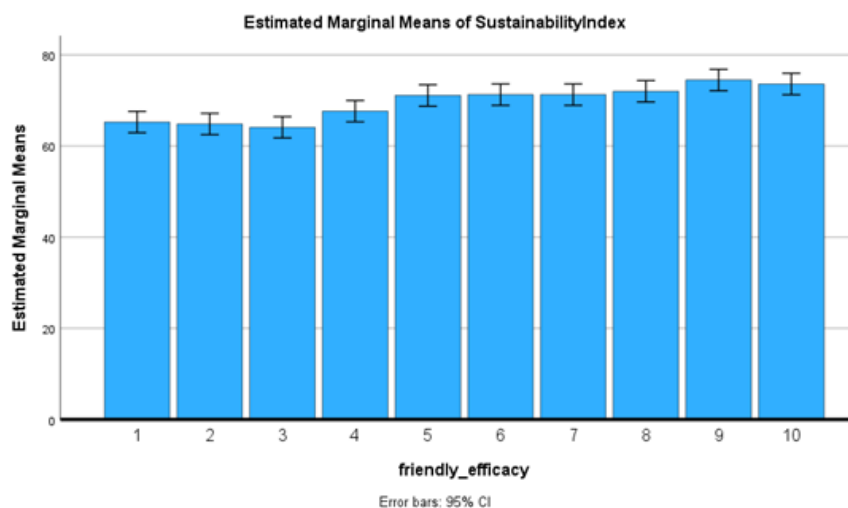
b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

AvailabilityIndex



SustainabilityIndex



Friendly Skills MANOVA Test Results

Between-Subjects Factors

		N
friendly_skills	1	500
	2	500
	3	500
	4	500
	5	500
	6	500
	7	500
	8	500
	9	500
	10	500

Estimated Marginal Means

1. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.413	.001	.411	.415
SustainabilityIndex	66.360	.351	65.671	67.048

Multivariate Tests^a

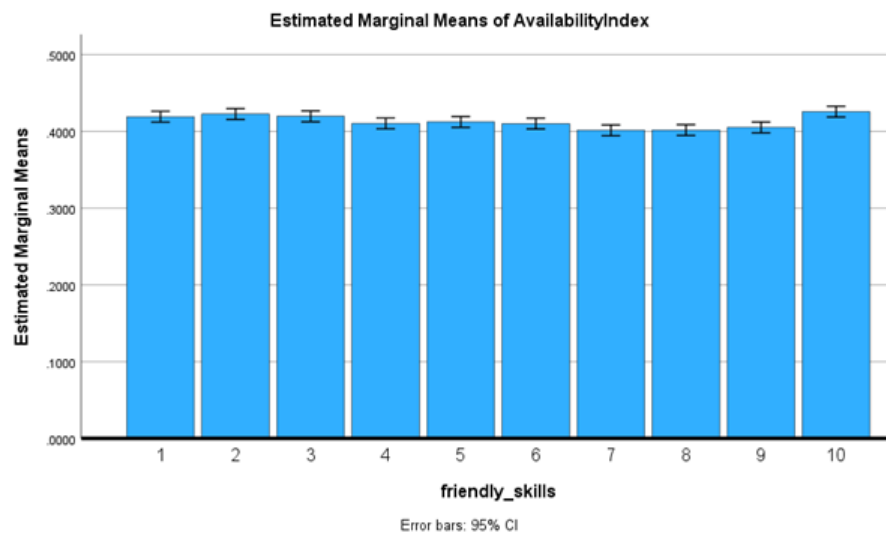
Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.991	276356.103 ^b	2.000	4989.000	<.001
	Wilks' Lambda	.009	276356.103 ^b	2.000	4989.000	<.001
	Hotelling's Trace	110.786	276356.103 ^b	2.000	4989.000	<.001
	Roy's Largest Root	110.786	276356.103 ^b	2.000	4989.000	<.001
friendly_efficacy	Pillai's Trace	.021	5.991	18.000	9980.000	<.001
	Wilks' Lambda	.979	6.013 ^b	18.000	9978.000	<.001
	Hotelling's Trace	.022	6.036	18.000	9976.000	<.001
	Roy's Largest Root	.020	11.174 ^c	9.000	4990.000	<.001

a. Design: Intercept + friendly_efficacy

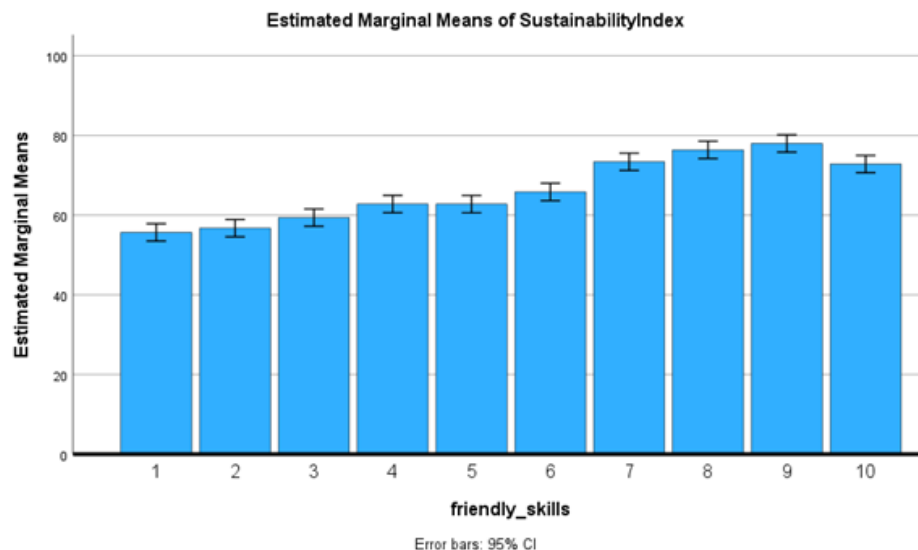
b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

AvailabilityIndex



SustainabilityIndex



Adversary Count MANOVA Test Results

Between-Subjects Factors

adversary_count	N
1	500
6	500
11	500
16	500
21	500
26	500
31	500
36	500
41	500
46	500
51	500
56	500
61	500
66	500
71	500
76	500
81	500
86	500
91	500
96	500
101	500

Estimated Marginal Means

1. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.431	.001	.429	.433
SustainabilityIndex	65.528	.068	65.394	65.662

Multivariate Tests^a

Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.996	1225913.480 ^b	2.000	10478.000	<.001
	Wilks' Lambda	.004	1225913.480 ^b	2.000	10478.000	<.001
	Hotelling's Trace	233.998	1225913.480 ^b	2.000	10478.000	<.001
	Roy's Largest Root	233.998	1225913.480 ^b	2.000	10478.000	<.001
adversary_count	Pillai's Trace	1.029	554.895	40.000	20958.000	<.001
	Wilks' Lambda	.001	19194.520 ^b	40.000	20956.000	<.001
	Hotelling's Trace	1373.967	359876.320	40.000	20954.000	<.001
	Roy's Largest Root	1373.937	719874.159 ^c	20.000	10479.000	<.001

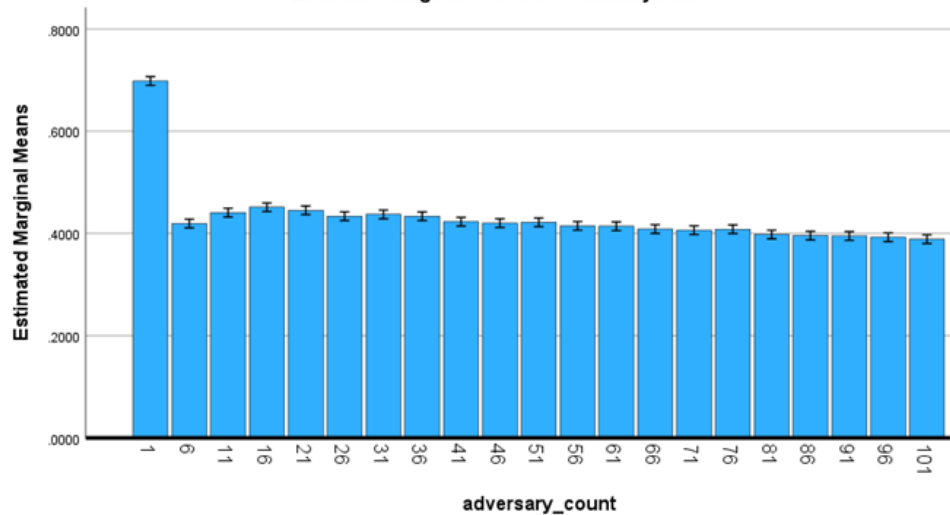
a. Design: Intercept + adversary_count

b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

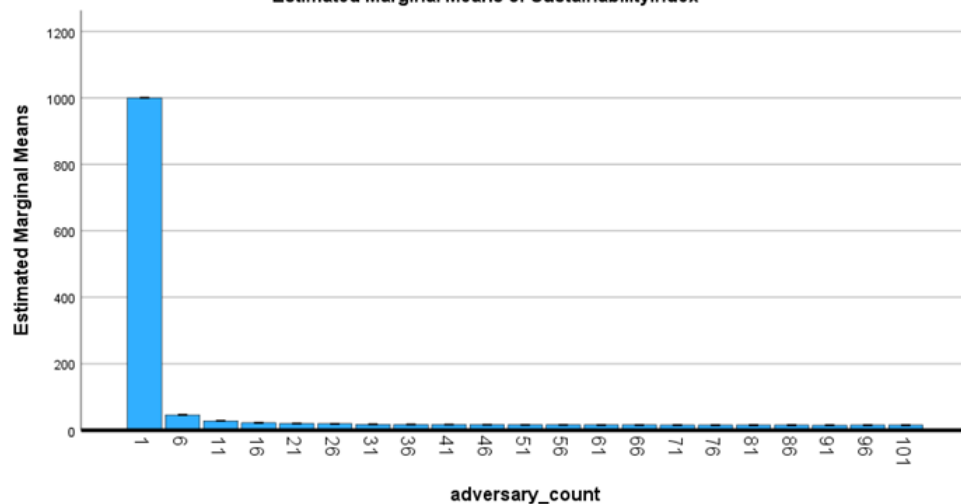
AvailabilityIndex

Estimated Marginal Means of AvailabilityIndex



SustainabilityIndex

Estimated Marginal Means of SustainabilityIndex



Adversary Skills MANOVA Test Results

Between-Subjects Factors

	N
adversary_skills 1	500
2	500
3	500
4	500
5	500
6	500
7	500
8	500
9	500
10	500

Estimated Marginal Means

1. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.480	.001	.478	.482
SustainabilityIndex	228.595	1.130	226.380	230.810

Multivariate Tests^a

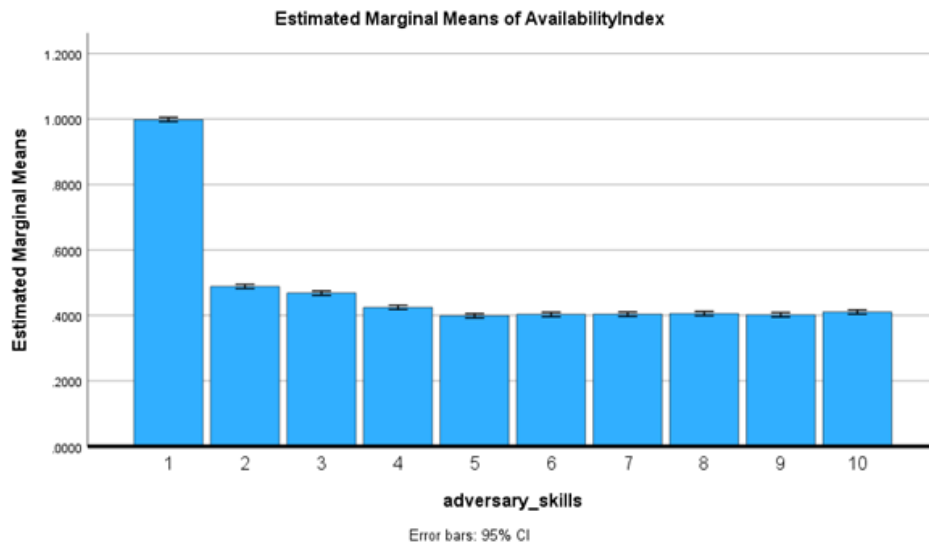
Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.996	1225913.480 ^b	2.000	10478.000	<.001
	Wilks' Lambda	.004	1225913.480 ^b	2.000	10478.000	<.001
	Hotelling's Trace	233.998	1225913.480 ^b	2.000	10478.000	<.001
	Roy's Largest Root	233.998	1225913.480 ^b	2.000	10478.000	<.001
adversary_count	Pillai's Trace	1.029	554.895	40.000	20958.000	<.001
	Wilks' Lambda	.001	19194.520 ^b	40.000	20956.000	<.001
	Hotelling's Trace	1373.967	359876.320	40.000	20954.000	<.001
	Roy's Largest Root	1373.937	719874.159 ^c	20.000	10479.000	<.001

a. Design: Intercept + adversary_count

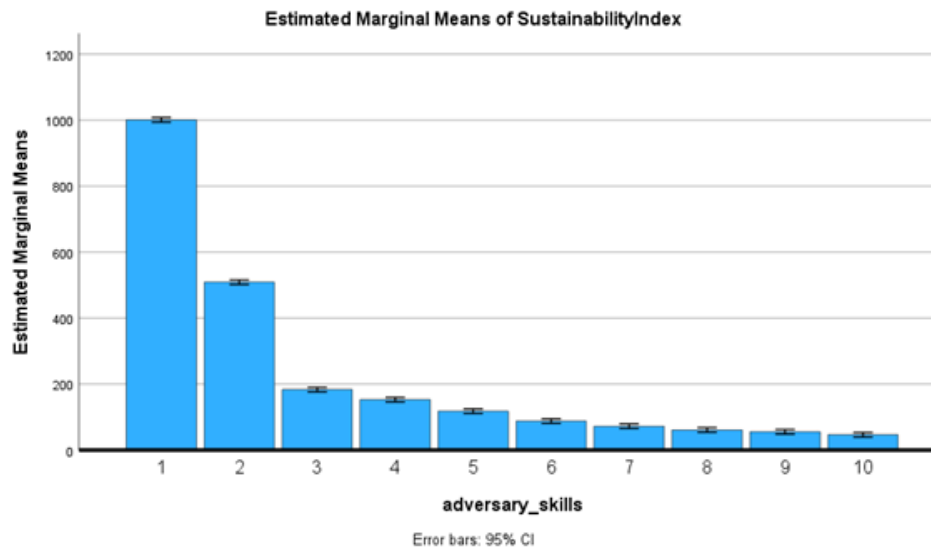
b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

AvailabilityIndex



SustainabilityIndex



Friendly Count Focused MANOVA Test Results

Between-Subjects Factors

		N
friendly_count	1	500
	2	500
	3	500
	4	500
	5	500
	6	500
	7	500
	8	500
	9	500
	10	500
	11	500
	12	500
	13	500
	14	500
	15	500

Estimated Marginal Means

1. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.626	.001	.625	.627
SustainabilityIndex	676.062	1.146	673.815	678.309

Multivariate Tests^a

Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.996	1026922.053 ^b	2.000	7484.000	<.001
	Wilks' Lambda	.004	1026922.053 ^b	2.000	7484.000	<.001
	Hotelling's Trace	274.431	1026922.053 ^b	2.000	7484.000	<.001
	Roy's Largest Root	274.431	1026922.053 ^b	2.000	7484.000	<.001
friendly_count	Pillai's Trace	1.557	1881.176	28.000	14970.000	<.001
	Wilks' Lambda	.009	4972.527 ^b	28.000	14968.000	<.001
	Hotelling's Trace	44.975	12019.478	28.000	14966.000	<.001
	Roy's Largest Root	43.595	23307.652 ^c	14.000	7485.000	<.001

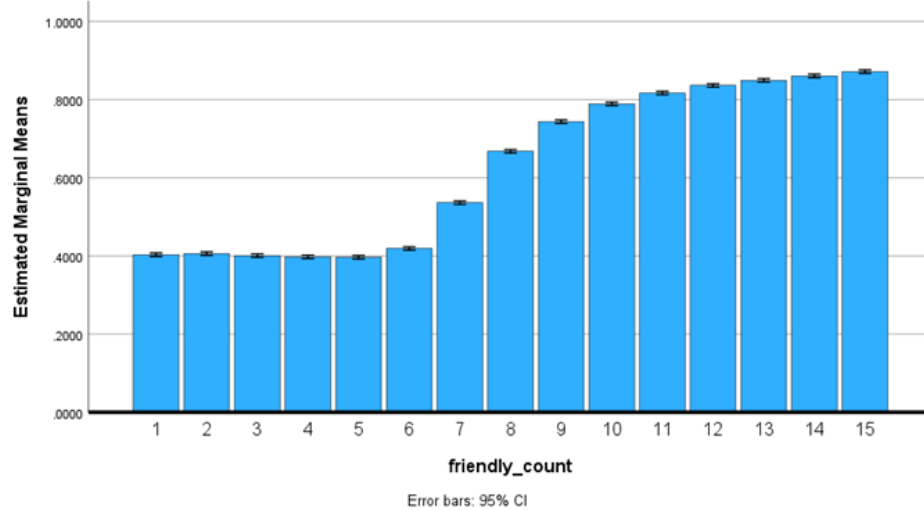
a. Design: Intercept + friendly_count

b. Exact statistic

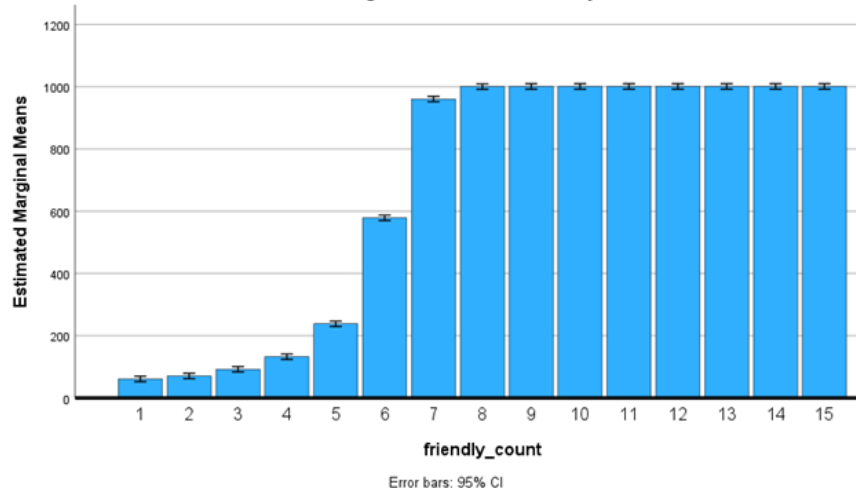
c. The statistic is an upper bound on F that yields a lower bound on the significance level.

AvailabilityIndex

Estimated Marginal Means of AvailabilityIndex



Estimated Marginal Means of SustainabilityIndex



Adversary Count Focused MANOVA Test Results

Estimated Marginal Means

Between-Subjects Factors

	N
adversary_count 1	500
2	500
3	500
4	500
5	500
6	500
7	500
8	500
9	500
10	500
11	500
12	500
13	500
14	500
15	500

1. Grand Mean

Dependent Variable	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
AvailabilityIndex	.446	.001	.444	.448
SustainabilityIndex	118.683	.449	117.803	119.563

Multivariate Tests^a

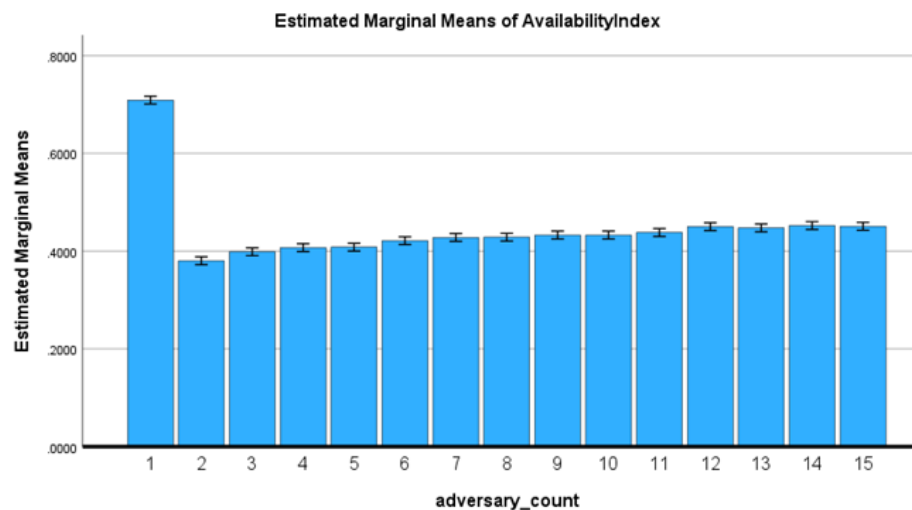
Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.980	186626.425 ^b	2.000	7484.000	<.001
	Wilks' Lambda	.020	186626.425 ^b	2.000	7484.000	<.001
	Hotelling's Trace	49.873	186626.425 ^b	2.000	7484.000	<.001
	Roy's Largest Root	49.873	186626.425 ^b	2.000	7484.000	<.001
adversary_count	Pillai's Trace	1.092	643.371	28.000	14970.000	<.001
	Wilks' Lambda	.018	3448.247 ^b	28.000	14968.000	<.001
	Hotelling's Trace	48.386	12931.253	28.000	14966.000	<.001
	Roy's Largest Root	48.259	25801.580 ^c	14.000	7485.000	<.001

a. Design: Intercept + adversary_count

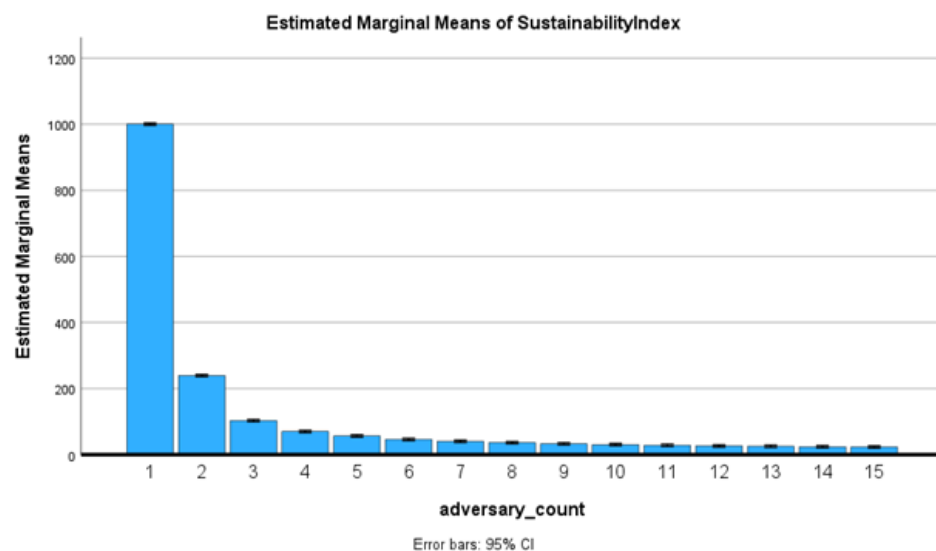
b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

AvailabilityIndex



SustainabilityIndex



GSA Adversary & Friendly MANCOVA Test Results

Between-Subjects Factors

	N
friendly_count	1 900
	2 900
	3 900
	4 900
	5 900
	6 900
	7 900
	8 900
	9 900
	10 900
	11 900
	12 900
	13 900
	14 900
	15 900
adversary_count	1 900
	2 900
	3 900
	4 900
	5 900
	6 900
	7 900
	8 900
	9 900
	10 900
	11 900
	12 900
	13 900
	14 900
	15 900

Multivariate Tests^a

Effect		Value	F	Hypothesis df	Error df	Sig.
Intercept	Pillai's Trace	.915	71389.726 ^b	2.000	13273.000	<.001
	Wilks' Lambda	.085	71389.726 ^b	2.000	13273.000	<.001
	Hotelling's Trace	10.757	71389.726 ^b	2.000	13273.000	<.001
	Roy's Largest Root	10.757	71389.726 ^b	2.000	13273.000	<.001
node_count	Pillai's Trace	.513	6990.205 ^b	2.000	13273.000	<.001
	Wilks' Lambda	.487	6990.205 ^b	2.000	13273.000	<.001
	Hotelling's Trace	1.053	6990.205 ^b	2.000	13273.000	<.001
	Roy's Largest Root	1.053	6990.205 ^b	2.000	13273.000	<.001
friendly_count	Pillai's Trace	.605	411.081	28.000	26548.000	<.001
	Wilks' Lambda	.402	547.963 ^b	28.000	26546.000	<.001
	Hotelling's Trace	1.474	698.617	28.000	26544.000	<.001
	Roy's Largest Root	1.463	1386.986 ^c	14.000	13274.000	<.001
adversary_count	Pillai's Trace	.697	507.582	28.000	26548.000	<.001
	Wilks' Lambda	.308	760.553 ^b	28.000	26546.000	<.001
	Hotelling's Trace	2.231	1057.461	28.000	26544.000	<.001
	Roy's Largest Root	2.223	2107.975 ^c	14.000	13274.000	<.001
friendly_count * adversary_count	Pillai's Trace	.564	26.583	392.000	26548.000	<.001
	Wilks' Lambda	.506	27.447 ^b	392.000	26546.000	<.001
	Hotelling's Trace	.836	28.319	392.000	26544.000	<.001
	Roy's Largest Root	.609	41.249 ^c	196.000	13274.000	<.001

a. Design: Intercept + node_count + friendly_count + adversary_count + friendly_count * adversary_count

b. Exact statistic

c. The statistic is an upper bound on F that yields a lower bound on the significance level.

Tests of Between-Subjects Effects

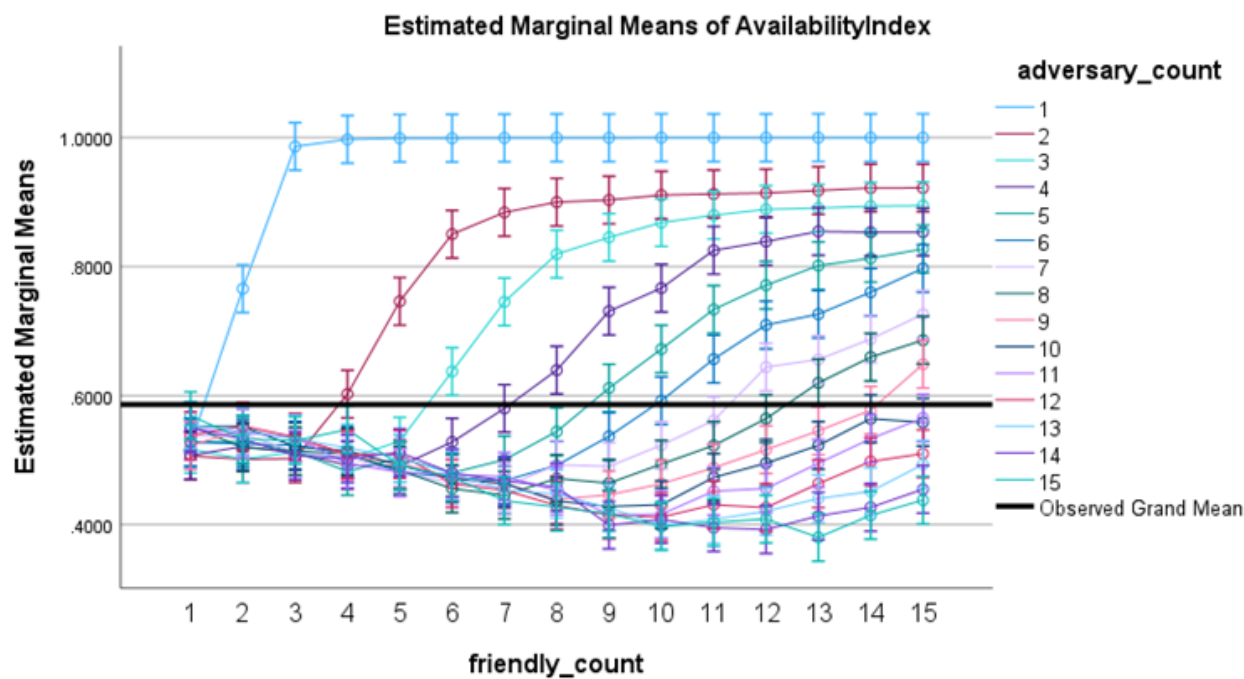
Source	Dependent Variable	Type III Sum of Squares	df	Mean Square	F	Sig.
Corrected Model	AvailabilityIndex	681.725 ^a	225	3.030	142.466	<.001
	SustainabilityIndex	1943892670 ^b	225	8639522.977	149.888	<.001
Intercept	AvailabilityIndex	2567.328	1	2567.328	120716.204	<.001
	SustainabilityIndex	286929515.38	1	286929515.38	4977.973	<.001
node_count	AvailabilityIndex	284.912	1	284.912	13396.631	<.001
	SustainabilityIndex	135344819.32	1	135344819.32	2348.113	<.001
friendly_count	AvailabilityIndex	32.791	14	2.342	110.133	<.001
	SustainabilityIndex	874389181.75	14	62456370.125	1083.563	<.001
adversary_count	AvailabilityIndex	259.658	14	18.547	872.082	<.001
	SustainabilityIndex	639679771.99	14	45691412.285	792.706	<.001
friendly_count * adversary_count	AvailabilityIndex	104.363	196	.532	25.037	<.001
	SustainabilityIndex	294478896.72	196	1502443.351	26.066	<.001
Error	AvailabilityIndex	282.304	13274	.021		
	SustainabilityIndex	765111092.33	13274	57639.829		
Total	AvailabilityIndex	5601.555	13500			
	SustainabilityIndex	5216998801.0	13500			
Corrected Total	AvailabilityIndex	964.029	13499			
	SustainabilityIndex	2709003762.1	13499			

a. R Squared = .707 (Adjusted R Squared = .702)

b. R Squared = .718 (Adjusted R Squared = .713)

GSA Adversary & Friendly MANCOVA Test Results

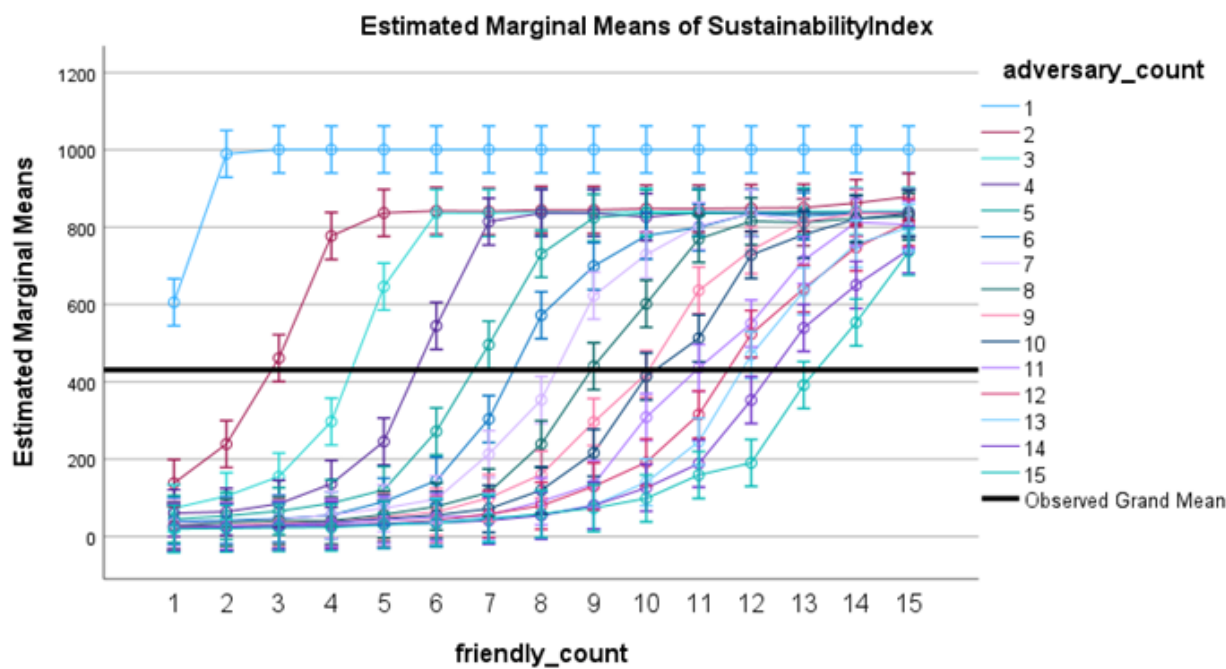
AvailabilityIndex



Covariates appearing in the model are evaluated at the following values: node_count = 13.50

Error bars: 95% CI

SustainabilityIndex



Covariates appearing in the model are evaluated at the following values: node_count = 13.50

Error bars: 95% CI