A Functional All-Hazard Approach to Critical Infrastructure Dependency Analysis

A Dissertation
Presented in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy
with a
Major in Computer Science
in the
College of Graduate Studies
University of Idaho
by
Ryan C. Hruska

Approved by:
Major Professor: Michael A. Haney, PhD
Committee Members: Robert Borrelli, PhD; Ron Fisher, PhD; Constantinos Kolias, PhD
Department Administrator: Terence Soule, PhD

May 2023

Abstract

Infrastructure systems are the backbone of modern societies and are critical for well-functioning communities. Natural and human-induced hazards have the potential to disrupt the services provided by these systems, thus impacting normal community functions. For a community to assess risk from any hazard it must first understand its dependency on the services provided by supporting infrastructure and, in turn, how the infrastructure is not only vulnerable to the hazard, but dependent on other infrastructure systems that may also be vulnerable. However, merely understanding the consequences and impacts of interdependent infrastructure failures on critical community services, let alone prioritizing capital investments to shore up aging, failing, or otherwise vulnerable systems, is a daunting and often unachievable challenge for communities with limited resources.

This dissertation proposes a novel all-hazards analysis (i.e., AHA) methodology and knowledge management framework to enhance our understanding of risks to interconnected infrastructure, systems, and networks. The methodology advances risk analytic capabilities through the integration of concepts from graph theory, knowledge representation, and function-based engineering design. Infrastructure systems are modeled as multilayer networks and their behavior is simulated through the application of scalable function-failure logic designed to enhance risk mitigation guidance, while reducing the need for high-fidelity engineering data. In addition, the AHA Text Analytic System is proposed to enhance the population of the knowledge base through the development and application of infrastructure-specific named entity recognition.

Acknowledgments

A number of people have contributed to the work represented in this dissertation with ideas, insight, and encouragement. Without the support of many other intelligent and talented colleagues, this research would not have been possible. I am very appreciative of the support of my major adviser, committee members, and colleagues.

Dr. Michael Haney, my major professor, has been there to encourage me, not only as a research adviser but as an educator and mentor. I truly appreciate his support and patience during my graduate studies.

I would like to thank my dissertation committee members, Dr. Constantinos Kolias, Dr. Robert Borrelli, and Dr. Ron Fisher, for their support and guidance, including a special thanks to Dr. Fisher; without his mentorship and unwavering support for my continued education and research career, this would not have been possible.

I would like to acknowledge the educational opportunities and financial support of Idaho National Laboratory during my graduate studies at the University of Idaho. I would also like to acknowledge the financial support of the Department of Homeland Security and thank them for seeing value in this research. Finally, I would like to acknowledge my colleagues and teammates at Idaho National Laboratory—Robert Edsall, Cherrie Black, Kent McGillivary, Kenneth Larsen, and Michael Hoover.

Dedication

This dissertation is dedicated to my family; without their love and support, none of this would have been possible. I would like to thank my wife, Leah, whose hours of loving, educating, and entertaining our children, Rylea and Analea, provided me with the opportunity to complete this research, my parents, Carl and Judy, whose unfailing support of their children's curiosity and interests gave me the opportunity to explore the world, and most of all Rylea and Analea, whose unconditional love and excitement for education gave me the motivation to see this through.

Table of Contents

List of Tables

List of Figures

Statement of Contribution

This dissertation is based upon work supported in part by the Idaho National Laboratory (INL) Laboratory Directed Research & Development Project 14-093: All Hazards Critical Infrastructure Knowledge Framework under the Department of Energy's Idaho Operations Office under contract DE-AC07-05ID14517 and Cybersecurity and Infrastructure Security Agency (CISA) under contract No. 70RCSA20K00000033. In fulfilling the requirements of these contracts, the author served as principal investigator, leading both the proposal development and research activities required to design, develop, and test the resulting all-hazards analysis framework (i.e., AHA Framework). The development of the AHA software application was supported by multiple INL staff members including Kent McGillivary, Michal Hoover, Kenneth Larsen, and Mary Klett. In addition, Dr. Robert Edsall served as INL's CISA project manager to ensure stakeholder requirements were met. As a result, the following multiauthor manuscript was published in the *Journal of Critical Infrastructure Policy*. Under the copyright agreement with the Policy Studies Organization, the authors retain ownership of the copyright and may reproduce the article in whole or part.

Manuscript:

R. Hruska, K. McGillivary, and R. Edsall. "A Functional All-Hazard Approach to Critical Infrastructure Dependency Analysis," *Journal of Critical Infrastructure Policy*, vol. 2, no. 2, pp. 103-123, 2021, DOI: 10.18278/jcip.2.2.6.

The authors confirm contribution to the paper as follows: study conception and design, R. Hruska; data collection, R. Hruska; analysis and interpretation of results, R. Hruska and K. McGillivary; and draft manuscript preparation, R. Hruska and R. Edsall. All authors reviewed the results and approved the final version of the manuscript.

Chapter 1: Introduction

Motivation

Infrastructure systems are the backbone of modern societies and are critical for well-functioning communities. Through agriculture, industrial, and technology innovation, the human species has been uniquely able to modify their habitat to effectively increase carrying capacity and allow a high degree of individual specialization among its population. Much of this success can be attributed directly to humankind's ability to design and develop infrastructure systems that optimize the production and transportation of commodities, services, and information for industrial and domestic consumption. The engineering and technology innovations that occurred during and after the *Industrial and Information Revolutions* (IIR) created global and regional supply chains that have overcome significant geographic barriers between centralized commodity production and final consumption. Driven by economies of scale, these infrastructure systems and supply chains are often automated and optimized to allow for continued growth at the minimum cost possible. Hazards such as the COVID-19 pandemic, Hurricane Maria, and recent, debilitating cyberattacks attest that this practice can result in overtaxed infrastructure systems and single points-of-failures that, if disrupted, lead to widespread goods and services shortages that transverse local, state, and national borders. Further, these infrastructure systems have coevolved into highly interconnected network-of-networks which are potentially vulnerable to cascading, escalating, and common cause failures [1, 2].

Since the PCCIP report raised awareness about the need to maintain, protect, and enhance the resilience of these infrastructure systems, many governments (e.g., United States) have developed critical infrastructure and emergency response programs to assess the risk of infrastructure failures, aid in mitigating vulnerabilities, and assistance in recovering from disruption. For example, presidential policy directive (PPD)-8 directed the U.S. Department of Homeland Security (DHS) to establish a national preparedness goal which states: "[a] *secure and resilient nation with the capabilities required across the whole community to prevent, protect against, mitigate, respond to, and recover from the threats and hazards that pose the greatest risk*" [3]. Core to achieving this goal is understanding how infrastructure systems enable critical community and government services and identify potential vulnerabilities that could pose significant risk to their operations [4, 5]. Further, PPD-21 established an additional policy intended to strengthen and maintain secure, functioning, and resilient critical infrastructure and redefined critical infrastructure as "[t]*he physical and cyber systems and assets that are so vital to the United States that their incapacity or destruction would have a debilitating impact on our physical or economic security or public health or safety. The Nation's critical infrastructure provides the essential services that underpin American society*" [6]. In

response, many federal, state, and local government agencies and organizations have developed infrastructure protection and risk mitigation plans, such as the National Infrastructure Protection Plan (NIPP), to address these policies' requirements [7].

These policies, directives, and plans have resulted in the need for advanced analytic techniques to better identify, understand, and analyze infrastructure system criticality, risks, and resilience, including improving our ability to understand their interdependencies [8-11]. Since the PCCIP report was published in 1997, a substantial body of research has been devoted to this task. The increase in situational awareness brought about by this research has enabled resource planners and emergency response organizations to effectively mitigate many vulnerabilities and direct response-and-recovery efforts following a natural or human-induced event. However, events like the February 2021 Texas polar vortex, where a complex series of cascading disruptions to power, natural gas, and water infrastructure left homes and businesses without power and water, crippled supply chains and transportation networks, and sadly was directly or indirectly responsible for over 100 deaths [12]. This and analogous events demonstrate our ability to routinely identify and mitigate cross-sector vulnerabilities from an all-hazards perspective is critical but remains an open and difficult problem.

This is primarily because critical infrastructures display a wide range of spatial, temporal, operational, organizational, and interdependent characteristics which can affect their ability to adapt to changing conditions. The inherent complexity of these systems can introduce subtle interactions and feedback mechanisms that often lead to unintended behavior and consequences during a disruption [13]. Understanding of the vulnerabilities and risks from an all-hazards perspective is further complicated by the fact that in most cases, cyber, physical, and supply domains have been addressed independently of one another. This separation has led to lack of a shared understanding of the threat and hazard landscape. All-hazards refers to the full spectrum of emergencies or disasters, from natural to technology to human-induced disruptions.

From a research and technology point-of-view, there are three major implementation issues that impact the efficacy of currently available vulnerability and risk assessment methods and tools for critical infrastructure from an all-hazards perspective.

- First is the lack of a holistic understanding of the vulnerability, hazard, and threat landscape from an integrated cyber, physical, supply, and interdependency perspective.
- Second is the availability of actuarial data on the characteristics, operational state, and (inter)dependencies of critical infrastructure assets, facilities, and systems.

- Third is the accessibility of usable knowledge discovery and decision support capabilities that results in actionable information across all critical infrastructure sectors for vulnerability, risk, and consequence-driven decision-making.

To effectively address these challenging national needs, a comprehensive knowledge discovery and decision support framework for critical infrastructure vulnerability and risk analysis must be developed. This will lead to enhanced understanding of these critical infrastructure systems' dependencies and interdependencies.

Problem Statement

Due to the complexity of modeling the behavior of interconnected infrastructure systems, there has been an absence of analysis methods, tools, and technologies that can be used to investigate the risks to and resilience of interconnected infrastructure under all-hazard conditions at a national scale. This is partially due to the debate regarding what type of analytic methods are most appropriate to evaluate both risks and resilience when accounting for dependencies within complex network-of-networks environment. In addition, the lack of a consistent knowledge model to collect, store, and analyze critical infrastructure information across domains further complicates the analytic process. As a result, numerous federal and state organizations have developed suboptimal and sometimes duplicative solutions to address risk and resilience knowledge gaps. Moreover, these solutions fail to consider the need to provide a scalable, robust, and repeatable process for developing dependency models of interconnected infrastructure systems and document their spatial and temporal characteristics under all-hazard conditions for risk and resilience analysis. Therefore, an all-hazards analysis knowledge model and analysis capability needs to be developed, which can be used for infrastructure protection, continuity of operations, and by emergency management organizations to aid in whole community and mission resilience assessments from local to national scales.

Research Objectives and Contributions

This dissertation's contribution is a novel all-hazards analysis framework (i.e., the AHA Framework) for critical infrastructure dependency modeling and analysis in defense of the following thesis:

- Functional-basis-informed graphs are ideal for describing and analyzing interconnected infrastructure system behavior under all-hazard conditions.
- Functional-basis-informed graphs provide an ideal structure for modeling function, commodity, and service flows of interconnected systems and facilitate scalable and repeatable assessments of system behaviors suitable for vulnerability, consequence, and risk analysis.

A functional basis for engineered systems is a standardized set of terminology and concepts required to develop meaningful infrastructure models. The proposed framework and supporting research will address the fundamental need for scalable and repeatable assessments capabilities to evaluate risks to and resilience of interconnected critical infrastructure across scales. This framework will provide three primary functions: (1) a standardized knowledge model for the collection, ingestion, and transformations of critical infrastructure dependency information, (2) cross-domain infrastructure dependency model development for all-hazard vulnerability and risk analysis, and (3) geospatially enable knowledge discovery and decision support methods for vulnerability, risk, and consequence analyses.

The main contributions of this dissertation are:

1. Novel application of advanced computer science techniques to enhance infrastructure resilience.
2. Functional basis for engineered infrastructure systems providing a formal language to describe interconnected infrastructure systems.
3. A scalable platform to collect, store, and model interconnected infrastructure systems information.
4. A scalable and robust functional-spatio-temporal approach for interconnected infrastructure behavior analysis.

Objective 1: Develop a Functional Basis for Engineered Infrastructure Systems to Facilitate a Scalable, Robust, and Repeatable Process for Developing Dependency Models of Interconnected Infrastructure Systems.

Objective 1 seeks to develop a robust and adaptable knowledge model based on a functional basis of infrastructure systems to facilitate the collection, storage, and analysis of dependency information suitable for risk and resilience analysis in support of crisis action and strategic risk mitigation activities. Current methods used for the analysis and visualization of dependency information are often challenged by the high volume and variety of data associated with infrastructure systems and supply chains. These issues are further compounded by the dynamic nature of infrastructure networks and the need to correlate data collected from multiple organizations across a region or the Nation.

Hypothesis:

> *Developing a functional basis of engineered infrastructure systems will improve our ability to collect consistent dependency information of infrastructure systems which is suitable for the risk and resilience analysis.*

Objective 2: Develop a Functional-flow Network Modeling Framework to Model the Behavior of Engineered Infrastructure Systems for the Purpose of Risk and Resilience Assessments.

Objective 2 seeks to determine whether functional-flow models are suitable for conducting scalable and repeatable assessments as well as risk and resilience assessments of an engineered infrastructure system and supply chains at different scales and resolutions. A functional-flow models is a graph-based description of a system or supply chain in terms of the elementary functions that are required to achieve its overall function or purpose.

Hypothesis:

> *Functional-flow models are suitable for modeling the behavior of infrastructure systems and supply chains.*

Objective 3: Investigate the Scalability and Robustness of Functional-flow Network Models for Simulating the Behavior of Interconnected Infrastructure Systems.

Objective 3 seeks to determine whether functional-flow models are suitable for conducting scalable and repeatable assessments as well as risk and resilience assessments of an interconnected engineered infrastructure system and supply chains at different scales and resolutions.

Hypothesis:

> *Functional-flow dependency models are suitable for simulating the behavior of interconnected infrastructure systems.*

Objective 4: Develop a Graph-based Knowledge Management System to Enable the Collection, Processing, and Analysis of Structured and Unstructured Infrastructure Data Required to Model Infrastructure Behavior under All Hazards.

Objective 4 seeks to develop a graph-based knowledge management system to automate data collection and processing *pipelines* to increase the efficiency of infrastructure dependency model development. Advances in information extraction and retrieval, such as natural language processing and understanding research, show great potential in optimizing the collection, processing, and synthesis of large volumes of infrastructure data.

Hypothesis:

> *Advances in information extraction and retrieval can increase the efficiency of dependency model creation and improve knowledge management.*

Dissertation Organization

The following chapters discuss how functional-flow dependency models and automated data collection methods can be used effectively to analyze the risk to and resilience of interconnected infrastructure systems. The remainder of the dissertation is organized as follows:

1. "Chapter 2: Background and Literature Review" provides a background and a review of works related to infrastructure dependency modeling and simulation.
2. "Chapter 3: All-Hazards Analysis (AHA) Methodology" develops and evaluates a proposed all-hazards analysis framework and knowledge model for all-hazards analysis of interconnected infrastructure systems in support of infrastructure protection and emergency management.
3. "Chapter 4: AHA Knowledge Management System" presents the AHA knowledge management system for collecting and analyzing dependency information of interconnected infrastructure systems for all-hazards risk and resilience analysis.
4. "Chapter 5: Application of the AHA Methodology to the Colonial Pipeline System" presents the development and analysis of cascaded function-flow models for the Colonial Pipeline System. The resulting model is validated against the 2021 Ransomware attack.
5. "Chapter 6: Preliminary Cyber-Physical Functional-Flow Model Analysis" presents the application of the proposed approach to cyber-physical systems and preliminary results.
6. "Chapter 7: AHA Data Collection and Processing" proposes the AHA Text Analytic System (TAS) to aid in knowledge base population. TAS leverages named entity recognition to extract facility information from unstructured infrastructure system information.
7. Finally, "Chapter 8: Conclusions and Future Work" discusses the conclusions of the research findings, including limitations and potential future work.

Chapter 2: Background and Literature Review

Due to the complexity of infrastructure dependencies and interdependencies, advanced computer science techniques are required to collect and analyze them. This chapter provides an overview of key concepts and work related to dependency analysis and knowledge creation for risk and resilience studies of interconnected infrastructure systems.

Infrastructure System Concepts

The following section provides a description of key concepts related to infrastructure systems.

Infrastructure

Understandably, infrastructure itself is at the core of infrastructure analysis; for the purpose of this dissertation, infrastructure is defined as *engineered systems and/or facilities that enable and enhance a community's ability to meet societal demands by facilitating the production, transport (transmission), and consumption of goods and services*. An engineered system is defined as a system designed or adapted to interact with an anticipated operational environment to achieve one or more intended purposes while complying with applicable constraints [14]. These systems can be composed of multiple interconnected facilities/assets, components, and software that perform specific actions to enable the functions of the system. It is reasonable to assume that engineered systems were deliberately designed and constructed to perform specific functions, thus their functions and purpose can be known.

Infrastructure Dependency

Infrastructure dependencies have been widely reported as a topic in scientific and engineering literature, and there are numerous papers that propose different taxonomies to characterize specific types of dependencies. However, many of these frameworks expand the concept for dependency beyond functional requirements and include dependency classes that are more akin to operational influencers (i.e., policy and regulatory requirements) and hazards (i.e., spatial proximity/geography) [10, 15]. For this reason, the U.S. DHS's *2013* NIPP is used which defines a dependency as *"*[t]*he one-directional reliance of an asset, system, network, or collection thereof—within or across sectors—on an input, interaction, or other requirement from other sources in order to function properly*" [7].

Linking this definition with the definition of infrastructure provides a coherent description of interconnected infrastructure systems that can be leveraged to clearly articulate both their functional (i.e., cyber and physical) requirements and consequences of failure.

Key Resources

Key resources are human and natural resources that are utilized, extracted, or managed to meet societal demands by facilitating the production and consumption of goods and services.

Supply Chain

There are many definitions for a supply chain such as "*the series of linked stages in a supply network along which a particular set of goods or services flows*" [16]. The DHS Supply Chain Resilience Guide defines a supply chain as "*the socio-technical network that identifies, targets, and fulfills demand. It is the process of deciding what, when, and how much should move to where*" [17]. However, for the purpose of this dissertation, a modified version of the definition provide in National Institute of Standards and Technology (NIST) SP 800-161 will be used, and supply chains will be defined as a network of retailer, distributor, transporter, storage, and production facilities that participate in the sale, delivery, and production of a particular good or service [18].

Risk

DHS defines risk as the "*potential for an adverse outcome assessed as a function of threats, vulnerabilities, and consequences associated with an incident, event, or occurrence*" [19]. The concept of risk in infrastructure systems is based on identifying the threat and hazards and the possible consequences and losses associated with their occurrence [20]. The authors contend it is not feasible to fully protect infrastructure systems from all hazards due to a constantly evolving threat landscape; however through careful analysis and planning, mitigations can be put in place to reduce risk to an acceptable level. Significant amounts of research have been conducted on infrastructure risk [21, 22].

Resilience

The concept of resilience has widely been reported in academic research to assess critical infrastructure [8, 20, 23-29]. DHS defines resilience as the "*ability of systems, infrastructures, government, business, communities, and individuals to resist, tolerate, absorb, recover from, prepare for, or adapt to an adverse occurrence that causes harm, destruction, or loss*" [19]. In the context of infrastructure systems, this often refers to the robustness, survivability, and recoverability of a system, asset, or component performance when exposed to a threat or hazard and is generally measured by its quality of service as shown in Figure 2-1. Most current research on resilience is focused on the development of metrics to better inform the risk mitigation and resilience decision-making process.

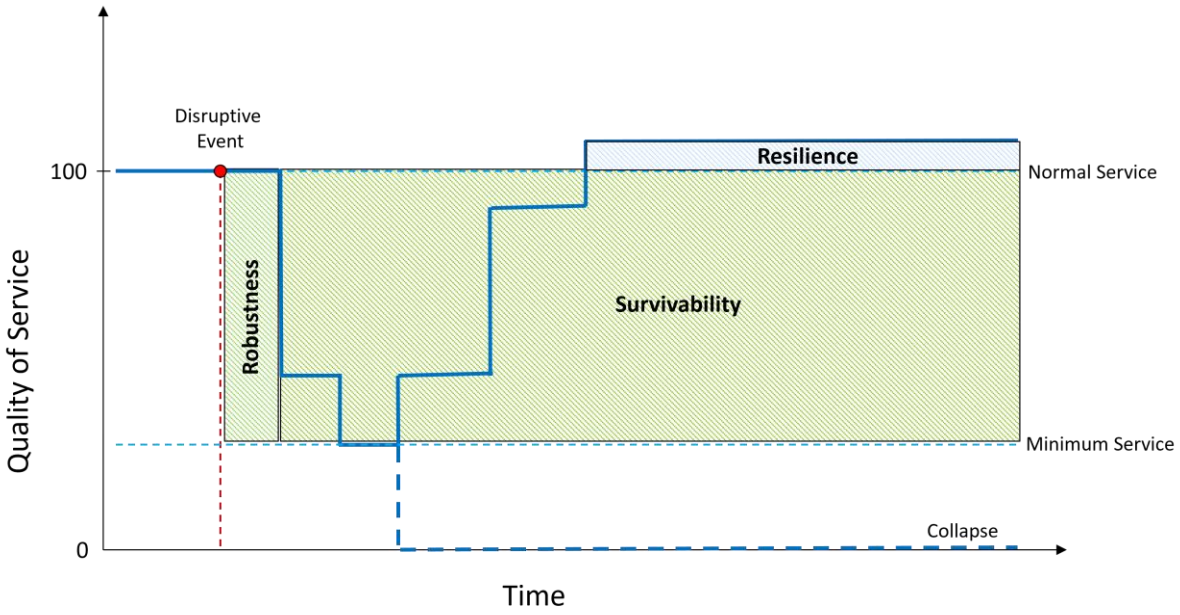

Figure 2-1 Resilience Curve.

All Hazards

All hazards "*means any threat or incident, natural or artificial, that warrants action to protect life, property, the environment, and public health or safety, and to minimize disruptions of government, social, or economic activities. It includes natural disasters, cyber incidents, industrial accidents, pandemics, acts of terrorism, sabotage, and destructive criminal activity targeting critical infrastructure*" [6]. When considering the concept of all hazards, it is important to consider "that warrant action" component of the definition, which helps limit the number of hazards or threats to a specific asset or component that requires attention. For example, landslides are not a direct concern for facilities that are not located in mountainous areas.

Infrastructure Analysis Concepts

Infrastructure systems and the supply chains they enable have evolved to form a dynamic and highly interconnected spatial network-of-networks that require cross-system and time-dependent interactions to function properly. Understanding these interactions has interested researchers from a variety of fields from engineering to social science; interdisciplinary efforts to quantify risk and improve resilience have incorporated modeling of infrastructure systems and simulations of disruptions to prepare or respond to hazard events. To understand and characterize the resulting complex network-of-networks behaviors, numerous analytic approaches have been proposed and findings published in the scientific and engineering literature, including multiple review and survey articles [8, 9, 11, 26, 29, 30]. For this reason, an overview of the more significant literature reviews of dependency

research contributions, such as Satumtira and Dueñas-Osorio [11] and Ouyang [9], which focus on the variety of modeling and simulation approaches to describe and quantify dependencies, risk, cascading impacts, and their impact on overall resilience across time and space, will be provided.

Satumtira and Dueñas-Osorio examined the available recurring themes and dimensions of infrastructure dependency modeling and simulation research in the published literature [31]. In their review, they found the chosen mathematical modeling method provided the strongest distinction between approach and included input-output, agent-based, and network (graph theory) techniques. Other cited dimensions of study include the objective of the modeling effort (e.g., risk and vulnerability analysis, mitigation measures, prediction, and failure propagation awareness), scale of analysis (system of systems) to specific networks and assets, and the targeted discipline (engineering [e.g., to study optimization of resources for reliability], economics [e.g., to study financial risk], and social sciences [e.g., to study decision-making and governance]).

Ouyang's review of infrastructure interdependency research centered on the emerging concept of infrastructure system resilience [9]. While the focus was on the modeling approaches to evaluate resilience, the author provided a classification of interdependency types—not all are physical (based on materials input and output between systems) but can be cyber (information), geographical (based on proximity), logical (policy, regulatory, or market-based), and even more conceptual (quantified by criticality or exclusivity); these distinctions are expanded typologies of [10] and [32]. Models, in their context, include empirical approaches, based on historical accidents or disaster data and expert experience, that reconstruct interdependencies from reports and records of past events such as hurricanes [33] or terror attacks [34]. Similar to [11], the review delineates agent-based and economic approaches and details the growing body of dependency research that leverages network-based approaches, further differentiating between those based on topology of networks [1], [31] and flow-based methods that represent capacities and storage at nodes and along links [35].

In both reviews, discussions of limitations to infrastructure interdependency studies began with the challenge of data access and collection. Empirical approaches to infer interdependencies are limited in scope because they are based on the type of hazard or incident and the incident context [36] and lack a uniform data collection method (including definitions of key concepts like resilience or types of interdependencies). Relevant data for useful modeling can be difficult to access due to confidentiality, business sensitivity, or liability concerns [10], and research innovations include the discovery of novel techniques for handling such sensitivities.

Data acquisition and the availability of accurate validation techniques using limited data is one of the several interdependency research gaps noted recently by Haggag and Ezzeldin while reviewing the interdependence of infrastructure to analyze the resilience of cities [8]. Their text-processing meta-analysis of over 120 publications in the area led to an inductive classification into nine topic areas, including definitions and descriptions of resilience, risk, and critical infrastructure in general, a survey of infrastructure interdependency modeling techniques (similar to the focus of other reviews), and a focused topic area of applying complex network theory, which continues to generate particular interest in the field. Further distinction in this analysis was between physical and functional networks, mirroring the topology vs. flow distinctions in [35]. Along with the challenge of data access and completeness, other gaps in research are presented by the authors, including the incomplete quantification of dependency types, the inability of research to scale entire systems of systems, the lack of linkages between hazards to the performance of systems, and the challenge of incorporating time and the dynamic behavior of systems to respond to shocks and changes.

The following subsections provide additional background on system engineering (SE), geographic information systems, network theory, and discrete event simulation approaches due to their relevance to this research. These concepts were selected because the SE provides domain-specific concepts related to the design and construction of infrastructure, and network theory provides intuitive and natural mathematical constructs to describe, model, and simulate infrastructure behavior.

System Engineering

SE is defined as “a methodical, multi-disciplinary approach for the design, realization, technical management, operations, and retirement of a system” [37]. As such, SE provides a suite of engineering-based design techniques to ensure that a proposed system will achieve the stakeholders’ functional, physical, and operational performance requirements once built [38-40]. The methods and techniques used during the design phase, which is composed of establishing stakeholder expectations, generating technical requirements, preforming logical decomposition, and evaluating design solutions, are of particular interest to this research. Adapting SE design approaches and techniques, such as integrated definition for function modeling (or IDEF) methods, can provide robust and repeatable techniques to reverse-engineer existing infrastructure systems to better understand their as-built operating envelope and consequences of disrupting one or more of a functional requirement’s components. Additional techniques include functional-flow block diagrams (FFBD), data flow diagrams, enhanced FFBD, and behavior diagrams [41].

As one might expect, functional modeling has become a critical task in SE to describe the intent of a system or products. Reference [42] proposed the concept of a functional basis to provide a consistent

approach to describe and model a product's function independent of a specific physical design. The author's primary goal was to provide a tool to aid mechanical engineers in evaluating design decisions earlier in a product life cycle by modeling flows (inputs/outputs) required to enable the products intended functions and sub-functions. The resulting functional basis decomposed flows into three primary classes, energy, materials, and signals, and functions into eight classes and 24 basic functions/operations (e.g., store, supply, and extract). The complete functional basis can be found in [43], and a comparison of other functional basis approaches can in found in [44].

One of the major steps in the SE design process is the functional design review which is intended to evaluate functions and determine what effects their disruption would have on system behavior prior to the more costly physical design phase. An example of tools used during this phase include failure mode and effects analysis, fault tree analysis, and model-based diagnosis; however, the functional-failure identification and propagation (FFIP) framework presented in [45] is the most relevant for this research. The FFIP framework utilizes behavioral simulations to evaluate fault propagation under different event scenarios. The FFIP framework leverages a functional basis to describe component functionality and configuration flow graphs (CFG) to define the system topology. The behavior of the system is modeled by linking the reusable behavior models of the systems components that represent both discrete nominal and faulty modes which are derived from input-output relations and underlying first principles. Transitions between modes are controlled by stated variables and are encoded as state transition diagrams, which the author refers to as function-failure logic (FFL).

Network (Graph) Theory Approaches

Most infrastructure systems are composed of a network of interconnected assets and components that facilitate the production, transport (transmission), and consumption of a good or service. This characteristic makes network-based analysis techniques ideal for evaluating both the structure, function, and behavior of interconnected infrastructure systems. This section provides an overview of the key network theory concepts and recent research related to the analysis of existing real-world interconnected infrastructure systems.

There have been significant works on the concept of network theory including [46-49]. In [48, 49], the authors review the body of research related to multilayer networks and establish they are well suited for the study of interconnected real-world infrastructure systems. Reference [48] defines multilayer networks as networks that "explicitly incorporate multiple channels of connectivity and constitute the natural environment to describe systems interconnected through different categories of connection: each channel (relationship, activity, category) is represented by a layer and the same node or entity may have different kinds of interactions (different set of neighbors in each layer)."

Reference [49] highlights that the rapid expansion of research literatures in this domain has resulted disparate terminology and lack of consensus on mathematical formulations, thus each of these reviews provides detailed mathematical definitions of several variations of multilayer networks.

For infrastructure related studies, [50] provides one of the first known works demonstrating that multigraphs provide a solid mathematical foundation for representing infrastructures and their dependencies. In [51], the authors extend the directed multigraph model to include the concepts of production, consumption, and storage of a dependency type and apply the model to synthetic representations of interconnected power, communication, and natural gas infrastructure systems. Inclusion of these time-dependent variables enabled dynamic analysis of integrated system behavior. Similarly, the authors in [52] advanced the use of multilayer networks models in their study of the New York City power, communication, and transport (train) infrastructure by including multicommodity system capacity. Reference [25] leverages these multilayer network analysis techniques to evaluate the behavior of interdependent infrastructure systems during disruptive events as well as the recovery phase to provide two novel metrics for the quantification of risk and resilience.

Geographic Information Systems

Infrastructure systems and supply chains form spatial networks which are inherently influenced by geography, thus geographic information systems (GIS) provide powerful platforms to collect, store, and visualize information about them and model phenomena that affect their operations. Reference [47] provides a review of spatial networks, which the author defines as a network with nodes located in a space equipped with a metric. Recognizing that infrastructure systems and supply chains inherently form spatial networks, researchers have long used GISs to map, analyze, and visualize infrastructure system information as well as the risks posed to them by artificial and natural hazards [52-55].

System Dynamics and Discrete Event Simulation

System dynamics (SD) and discrete event simulation (DES) models are used to understand system behavior with respect to time and compare their performance under different conditions [56, 57]. SD is a methodology commonly used to capture flows and feedback between components of the model in a causal-loop diagram. SD approaches are generally perceived to be ideal for strategic decision-making support. In contrast, DES, as the name implies, is a simulation technique that is driven by deterministic or stochastic events and used to simulate dynamic system behavior and is considered a better fit to inform tactical decisions. Reference [58] defines DES as "*the modelling of a system as it evolves over time by a representation in which the state variables change instantaneously at separate*

points in time. These points in time are the ones at which an event occurs, where an event is defined as an instantaneous occurrence that may change the state of the system." Both SD and DES approaches also have advantages over high-fidelity physics-based techniques by requiring less detailed engineering information. This makes them useful for evaluating interconnected infrastructure behavior, especially when access to real-time data is not available. For example, [59] leveraged DES to simulate the functional loss and restoration of the Napa water system following the 2014 earthquake. For a more in-depth discussion on the difference of the two approaches, refer to [56, 57].

Knowledge Representation, Generation, and Curation

Knowledge Graphs

Understanding interconnected infrastructure system behavior across scales and under all hazards requires integrating and synthetizing disparate concepts and information across many engineering, modeling, and scientific domains. This requirement has challenged researchers' ability to develop scalable and robust methods to study their structure and behavior; however, recent research on knowledge graphs has demonstrated researchers ability to overcome many of the challenges [60]. Through an analysis of related work, Ehrlinger and Wöß [61] define a knowledge graph, as a graph that "*acquires and integrates information into an ontology and applies a reasoner to derive new knowledge.*" As such, knowledge graphs represent a formal understanding of a domain or topic and can be more clearly defined as "*a structured representation of facts, consisting of entities, relationships, and semantic descriptions. Entities can be real-world objects and abstract concepts, relationships represent the relation between entities, and semantic descriptions of entities, and their relationships contain types and properties with a well-defined meaning* [62]." This is supported by Gruber's [63] description of an ontology as formally represented knowledge that is based on a specification of a conceptualization or a "*abstract, simplified view of the world that we wish to represent for some purpose.*" Gruber goes on to propose the following definition of an ontology as "*the names of entities in the universe of discourse (e.g., classes, relations, functions, and other objects) with human readable text describing what the names mean and formal axioms that constrain the interpretation and well-formed used of these terms.*" Similarly, [42, 43] present a functional basis for engineering design which they describe as a "*formal function representation*" that consists of "*a standardized set of function-related terminology*" which is intended to provide a common design language for functional modeling in engineering. Thus, a functional basis can be considered a specific type of ontology, and the two terms may be used to describe the same concept.

Further, Bryant et al. [64] argue that based on the ontologies in Robert G. Chenhall's *Nomenclature for Museum Cataloging*, logical naming systems should provide three levels of relationships based on the function of the object:

1. A controlled list of major categories, which are limited and easily remembered functional classes.
2. A controlled list of classification terms, which are subdivisions of the major categories.
3. An open vocabulary of object names used to identify individual artifacts.

The authors contend the use of a component function as a central and unifying concept provides the ideal structure for engineered system design and alignment through the "*theory of knowledge capture and representation and the theory of design*" [64]. This is supported by [65], which describes and defines the concept of domain-specific knowledge graphs as "*an explicit conceptualization to a high-level subject-matter domain and its specific subdomains represented in terms of semantically interrelated entities and relations.*" These properties make the concept of a domain-specific knowledge graph ideal for constructing a knowledge base for conducting risk and resilience analysis of existing infrastructure systems based on the idea that existing infrastructure systems were designed and built to provide specific functions. Additionally, the authors present a taxonomy for knowledge graph construction shown in Figure 2-2 below.

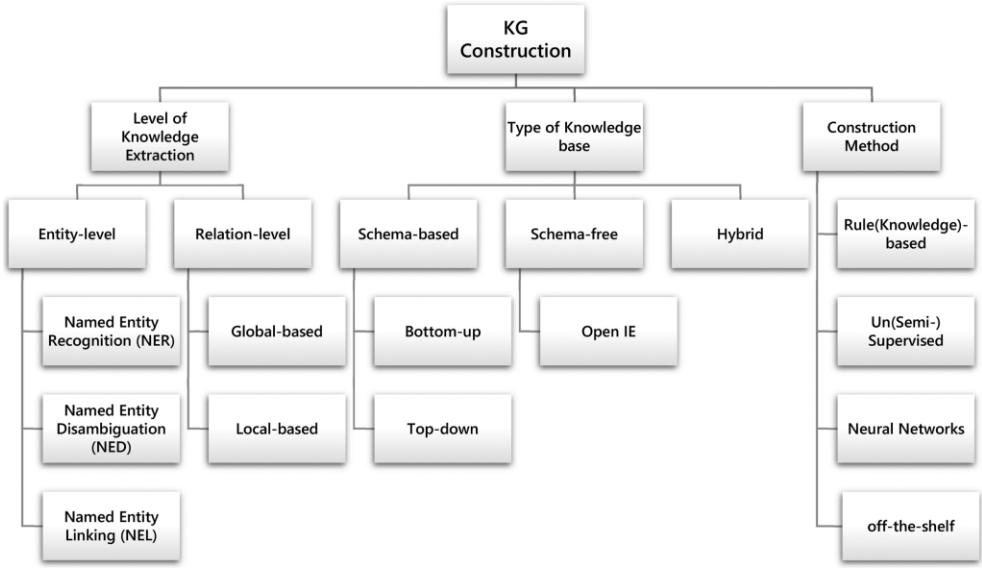

Figure 2-2 A Taxonomy for KG Construction [65].

Web Content Mining

Web content mining is a field of research that focuses on developing methods for the automated discovery of information and knowledge from unstructured and semi-structured Web data [66-71]. It

leverages methods and techniques from the fields of information retrieval (IR), information extraction (IE), machine learning, natural language processing (NLP), and data mining. Web content mining is also closely related to web structure (link) and web usage mining. Web content mining is challenging due to the diversity and quality of content, which is often noisy and dynamic. A brief overview of IR and IE is provided in this section.

Historically, information retrieval research focused on the automated retrieval of relevant documents from structured resources such as databases; however more recently, IR research has expanded to include retrieval of Web documents and resources by indexing, categorizing, and classifying unstructured text [66-71] and had been recently defined by Manning et al. as the processes of "*finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers)* [69]". IE can be considered a subdomain of IR and is based on NLP with the intent to derive structured information for unstructured text [68, 70]. In general, IE identifies concepts related to a specific domain while ignoring irrelevant information.

Natural Language Processing

NLP is a field of research that seeks to develop computational methods to automatically analyze the human language through deep understanding [72-75]. NLP research encompasses both written and spoken language processing, with three primary levels of processing: syntactic, semantic, and pragmatic. Early research in the field focused on developing syntactic approaches. These approaches centered on detecting and tagging the syntactic structure of written language, such a part of speech and sentence detection [75]. More recent research has focused the extraction of semantic content to better understand context through techniques such as the word-sense disambiguation process [76]. Pragmatic research attempts determine meaning from the context of the text to infer hidden meaning, which is critical for deep understanding [73, 74].

Historically, the primary approach used to develop natural language systems was statistical NLP [74, 77]. The statistical NLP approach is generally preferred because it can leverage machine-learning techniques to facilitate automated learning, and it reduces the need to maintain complex rule sets [77]. This approach is strongly rooted in probability, Bayesian, and information theories and had been shown to be very robust when tested with a large volume of text [78]. Since most statistical NLP techniques are considered supervised techniques, providing a "large enough" training set can become challenging. More recently, advances in machine-learning research, specifically the application of artificial neural networks (ANN), have reported great progress in domain-specific NLP tasks [68, 73]; however as [79] concludes, many of these studies are highly engineered, thus lack board applicability

and only report the best case results. Furthermore, the authors highlight the need for greater research on pretraining methodologies to overcome the variability of available domain-specific corpora.

Text Corpus

As stated above, corpus development is one of the most important and essential tasks for NLP applications. As defined by Sinclair, a corpus is "*a collection of pieces of language text in electronic form, selected according to external criteria to represent, as far as possible, a language or language variety as a source of data for linguistic research.* [80]" Further, Sinclair proposes ten guiding principles that should be considered when developing a corpus for linguistics research. These principles are shown in Figure 2-3. An excellent example of a corpus that generally adheres to these principles is the standard sample of Present-day English, commonly referred to as the Brown Corpus [81]. It was intended to capture a diverse range American English usage to facilitate comparative analysis. The Brown Corpus was published in 1964 and contains over 1 million words from 500 sources divided into hierarchical categories. It was later revised in 1979. The 1979 revision included "tagged" annotation content, making it suitable for supervised machine-learning applications, and it has been widely used for document categorization research.

1. The contents of a corpus should be selected based on their communicative function.
2. The corpus should be as representative as possible of the language from which it is chosen.
3. Only those components of corpora which have been designed to be independently contrastive should be contrasted.
4. Criteria for determining the structure of a corpus should be small in number.
5. Annotation should be stored separately from the plain text and merged when required in applications.
6. Samples of language for a corpus should wherever possible consist of entire documents.
7. The design and composition of a corpus should be documented fully.
8. The corpus should be representative and balanced with respect to domain.
9. Any control of subject matter in a corpus should be imposed by the use of external, and not internal, criteria.

Figure 2-3 Summarization of Sinclair's Guiding Principles for Corpus Development.

In corpus development, markup or annotation is one of the most important tasks needed to make a corpus useful for linguistic IE. Examples of commonly used annotations include parts of speech,

phrase structure (syntactic parsing), and named entities. Parts of speech are the most widely used annotation and critical for many other natural language process tasks [82]. It involves tagging every token in a corpus with the proper part of speech (i.e., <noun>computer</noun>). Other word-level features include case, punctuation, digit, morphology, and function. This is compared to phrase structure parsing which is conducted at the sentence level and annotates noun and verb phrase structures.

Documents Categorization

Document categorization is an essential task when organizing and processing large collections of documents for IR and extraction applications. Document categorization methods seek to bin documents into topics and typically utilize machine-learning techniques such as support vector machines (SVM), hidden Markov models (HMM), and convolutional neural networks (CNN) [83, 84]. There are two general approaches to document categorization which either use unsupervised or supervised methods. Unsupervised methods are typically referred to as topic modeling and attempt to derive classes by structuring the content as a bag of words. Supervised methods attempt to assign input documents to predefine classes based on priori knowledge obtained from an annotated training set. Typically, the most effective supervised learning approach has utilized a hierarchical classification scheme to mimic relationship of concepts within a domain [83-85].

Name Entity Recognition and Classification

Name entity recognition and classification (NERC) is a subdomain of NLP and is commonly used for text-based IE, retrieval, and mining applications, such as web content mining and knowledge base population (KBP) [86-89]. The first and most common application of NERC is the extraction of proper names relating to persons, locations, and organizations [90-92]. However, NERC techniques have also been widely applied in numerous domains to extract domain-specific terms [93]. Figure 2-4 shows an example of marked up text utilizing the Message Understand Conference (MUC) 7 annotation classes [94].

The Ebola outbreak in **<LOCATION>**West Africa**</LOCATION>** has sickened at least 9,936 people since **<DATE>**March**</DATE>**, killing at least 4,877 of them as of **<DATE>**Oct. 22**</DATE>** -- making it the worst outbreak of the virus in history, according

Figure 2-4 Named Entity Example.

Numerous conferences and workshops have been devoted to addressing the challenges of NERC, including the MUC, Automated Content Extraction program, and Conference on Natural Language

Learning (CoNLL) [91]. This has resulted in many diverse approaches and strategies. The simplest approach to name entity recognition (NER) is the use of a dictionary or keyword list of all known entities of interest that is used to iteratively search each resource for entities contained within a gazette or dictionary. However, this approach is challenged by the difficulties of creating and maintaining comprehensive lists of all possible entities, as well as resolving ambiguities [95]. To overcome this limitation, methods for automatically labeling textual features as named entities have been developed using rule-based approaches and machine-learning algorithms, such as maximum entropy and continuous random fields [96-98]. More recently, ANN and other deep-learning methods have been applied to the NER task, such as the work by Chen and Nichols [99] that applied a bidirectional long-short term memory (LSTM) unit and CNN to achieve superior NER performance over traditional methods. These results are consistent with a surveys of deep-learning approaches conducted by [79, 100]; however as [79] concludes, these results come with significant data requirements. It is also important to note that Schmitt et al. [101] found contrary results when directly comparing five publicly available NER software libraries against two well-known test corpora, where the CNN base solution was the least performant.

Currently, there are several publicly available NER models that can be utilized within open-source NLP machine-learning toolkits, such the Apache Software Foundation's OpenNLP Name Finder, Stanford NER, SpaCy, and Flair [101, 102]. The OpenNLP Name Finder algorithm is based on the maximum entropy (MAXENT) algorithm which is a statistical technique that maintains as much uncertainty as possible based on a set of constraints to classify without any prior assumptions about the probability distribution. For a comprehensive review of MAXENT as it applies to NLP, review Berger et al. [103]. The Stanford NER is a probabilistic-named entity classifier based on conditional random fields (CRF) [104]. It incorporates non-local structures using Gibbs sampling, a Markov chain Monte Carlo algorithm, and simulated annealing to produce long distance dependency models often found in natural languages and has been demonstrated to be effective for named entity extraction. These models are most often trained to recognize named entities defined on the MUC-7 classes; including people, locations, date, and organizations [94]. As noted above, [101] made a direct comparison of widely available NER solutions, and they found the Stanford NER outperformed the other solutions test.

Summary

In this chapter, a review of selected works that are relevant to the analysis of interconnected infrastructure systems and knowledge base population were presented. A few observations from this literature review include:

- Real-world infrastructure systems can be modeled as networks (graphs)
  - (Physical) engineered systems are not random
    - Physical infrastructure form static topologies
      - Physical infrastructure cannot change in real time and is not adaptive in an operational context
        - The state of components can be dynamic, allowing for adaptive operational topologies.
  - Different types of infrastructure form different topologies depending on the intended purpose/function of the system, such as
    - Transmission, distribution, or gathering
    - Production, transmission. consumption
- Recent advancement of knowledge graphs makes them ideal for capturing knowledge about real-world infrastructures
- NLP techniques can enhance knowledge base populations.

The research presented in the following chapters joins this body of literature in addressing the challenges of analyzing interconnected infrastructure systems, utilizing graph representation techniques to model nodes and links in systems (and between systems), and proposing novel techniques to create defensible and verifiable infrastructure dependency data with a multi-source approach. The framework differentiates types and quantities of dependencies, models generic types of infrastructure to enable rapid development of dependency information for high-level analysis, and builds capabilities to enhance situational awareness and assess resilience.

Chapter 3: All-Hazards Analysis (AHA) Methodology

In this chapter, the all-hazards analysis (AHA) methodology, an analytic framework developed to evaluate critical infrastructure dependencies and therefore identify potential vulnerabilities and consequences of system disruption, is introduced. AHA methodology's objective is to provide a scalable, robust, and repeatable process for developing and analyzing functional dependency models of interconnected infrastructure systems and document their spatial and temporal characteristics under all-hazard conditions. The AHA methodology is influenced by and attempts to synthesize the concepts presented in [42, 44, 45, 50-52, 55] and is intended to be dynamic and adaptive to enable analysis of emerging threat and hazards events.

Since the primary goal of this research is to evaluate the use of functional-basis-informed graphs for the purpose of modeling and simulating the behavior of interconnected infrastructure systems under all-hazard conditions for vulnerability, consequence, and risk analysis; the AHA methodology is influenced by and aligned to a simulation project life cycle, such as the one present by Robinson [105] as shown in shown in Figure 3-1.

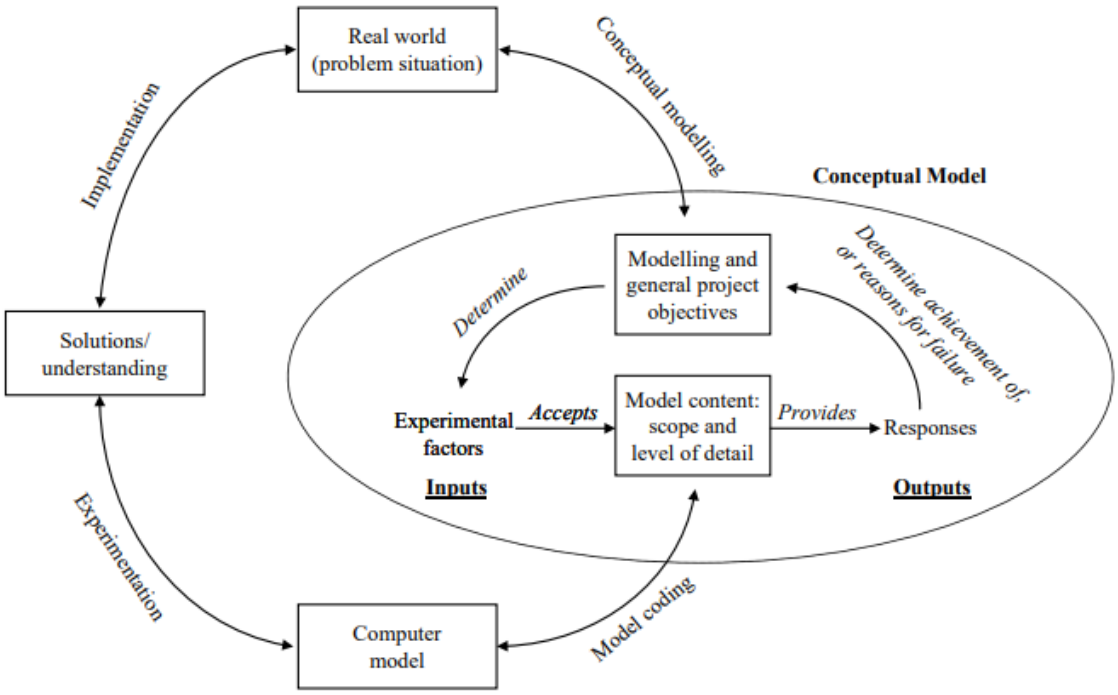

Figure 3-1 The Conceptual Model in the Simulation Project Life Cycle [105].

As Robinson acknowledges, the verification and validation activities are not explicitly described in the figure above but should be carried out in parallel with each of the four key process: conceptual

modeling, model coding, experimentation, and implementation. The following sections provide a review of the research thesis and objectives, as well as a detailed description of the proposed AHA methodology.

Research Thesis and Objectives

- **Thesis:** Functional-basis-informed graphs are ideal for describing and analyzing interconnected infrastructure system behavior under all-hazard conditions. Functional-basis-informed graphs provide an optimal structure for modeling function, commodity, and service flows of interconnected systems and facilitate scalable and repeatable assessments of system behaviors suitable for vulnerability, consequence, and risk analysis.
- **Objective 1**: Develop a functional basis for engineered infrastructure systems to facilitate a scalable, robust, and repeatable process for developing dependency models of interconnected infrastructure systems.
- **Objective 2**: Develop a functional-flow network modeling framework to model the behavior of engineered infrastructure systems for the purpose of risk and resilience assessments.
- **Objective 3**: Assess the ability of functional-flow network models to simulate the behavior of interconnected infrastructure systems, including their scalability and robustness.
- **Objective 4**: Develop a graph-based knowledge management system to enable the collection, processing, and analysis of structured and unstructured infrastructure data required to model infrastructure behavior under all hazards.

Terminology

One of major characteristics of a knowledge graph is a clear, standardized, and interlinked set of terminology used to describe the domain interest. Table 3-1 provides a list of terms that are core to the AHA knowledge representation and are defined here for clarity. Where possible, every attempt was made to align definitions from [42, 43, 64] and Sector-Specific Agency reports.

Table 3-1 Knowledge Graph Terminology.

Terminology	Definition	Source
Functional Basis	A design language consisting of a set of functions and a set of dependencies that are used to enable a system function.	Adapted from [30]
System Function	The primary input/output relationship of an infrastructure system, having the purpose of performing an overall task, typically stated in verb-object form.	Adapted from [30]
Function	A description of an operation to be performed by an asset or device, expressed as the active verb of the sub-function.	Adapted from [30]
Dependency Type (Flow)	A commodity, service, or datum that are exchanged between facilities/sub-facilities with respect to time. Expressed as the object of the function, a flow is the recipient of the function's operation.	Adapted from [30]
Facility/Assets	A structure or facility that has value and supports the provisioning of a commodity or service.	Adapted from the DHS Lexicon
Sub-Facility/ Components	An independently deployable device that exposes its functionality through a set of services accessed via well-defined interfaces and has value and support the provisioning of a good or service.	Adapted from the DHS Lexicon
Dependency Profile	A description of a facility or sub-facility in terms of dependencies that are required to achieve its overall function or purpose.	—
Dependency Model	A graph-based description of a system or supply chain in terms of the elementary functions that are required to achieve its overall function or purpose.	Adapted from [30]
Specific Property	An attribute or parameter that describes or characterizes a facility or dependency relationship.	—

AHA Real-World Problem Situation

Federal, state, and local risk and emergency management organizations, as well as infrastructure owners require information on infrastructure systems and their dependencies to inform their risk and recovery decision-making processes. For example, the Cybersecurity and Infrastructure Security Agency (CISA) has the goal "to reduce risks, and strengthen resilience of, America's critical infrastructure," which they intend to achieve by [106]:

- Expanding visibility of risks to infrastructure, systems, and networks
- Advancing CISA's risk analytic capabilities and methodologies
- Enhancing CISA's security and risk mitigation guidance and impact
- Building greater stakeholder capacity in infrastructure and network security and resilience
- Increasing CISA's ability to respond to threats and incidents.

Similarly, the chair of the Joints Chiefs of Staff (CJCS) has recently published the Joint Risk Analysis Methodology (JRAM) to establish a common risk lexicon to promote consistency across the across the Department of Defense (DoD) complex to better inform decisions and action on whether to accept, avoid, mitigate, or transfer risks related DoD operations [107]. The joint risk framework is shown in Figure 3-2 below.

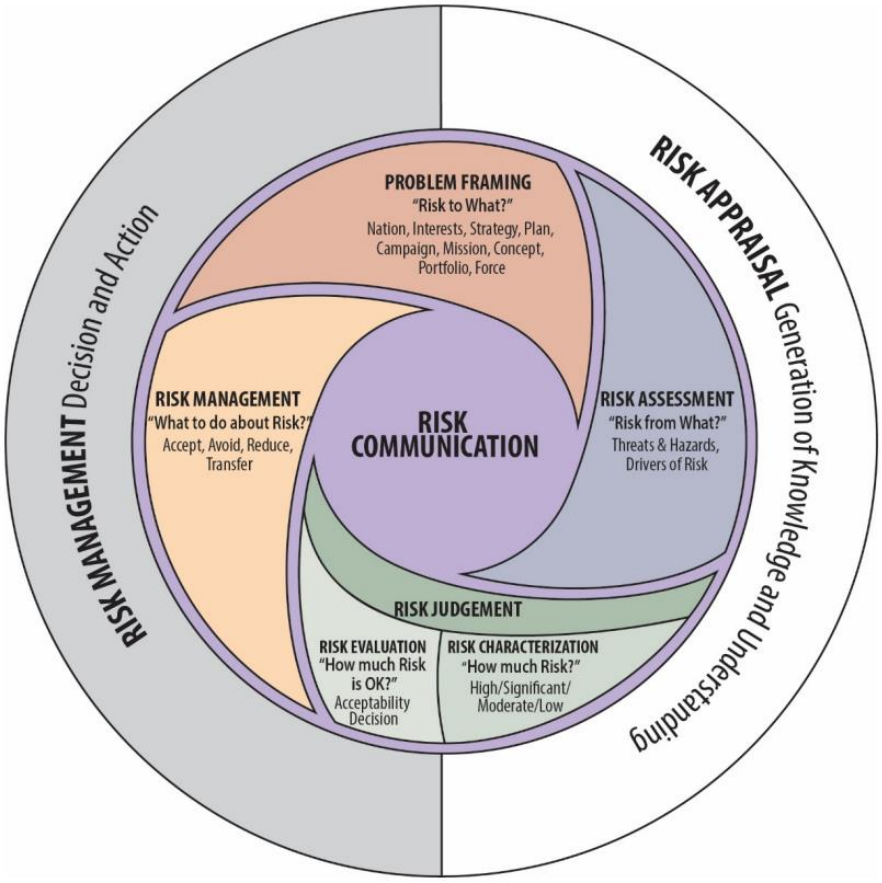

Figure 3-2 CJCS Joint Risk Framework.

For federal, state, local, territorial, tribal, and private sector organizations to realize these objectives, they need to be able to address several critical questions at each stage of the risk management and emergency response continuum. Figure 3-3 provides a breakdown of the major activities across the risk management and recovery continuum, and Table 3-2 provides a set of core questions by each stage derived from [4, 5, 17, 107-110].

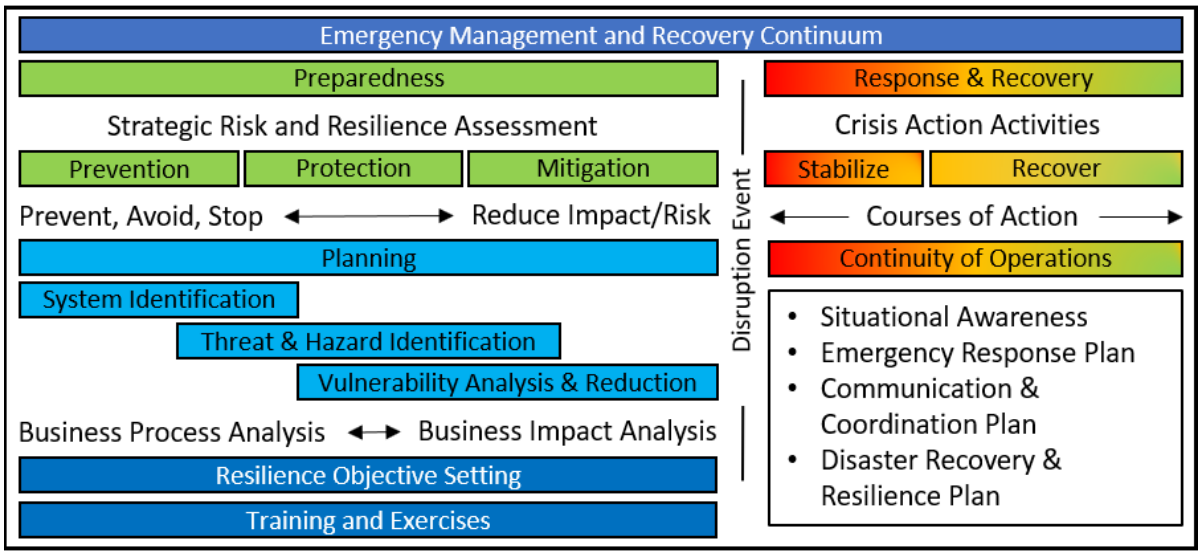

Figure 3-3 Risk and Resilience Management Continuum.

Table 3-2 Core Infrastructure System Risk and Resilience Questions.

Core Infrastructure System Risk and Resilience Questions by Stage	
Stage	Question and Context
Strategic Risk and Resilience	Are there national, regional, or local risks associated with the disruption of an infrastructure system, and is there a potential for cascading or escalating failures? Requires knowledge of consequences, threats, and vulnerabilities in the context of the system's operational environment, dependencies, and system geography.
Strategic Risk and Resilience	What are the potential threats and hazards that could impact the operation of an infrastructure system? Requires knowledge of operational environment requirements, dependencies, system geography, and threat/hazard potential.
Strategic Risk and Resilience/ Mitigation-COOP	What are the potential mitigation options that can reduce risks and enhance resilience of an infrastructure system? Requires knowledge of operational environment requirements, dependencies, system geography, threat/hazard potential, and consequences of disruption.

Table 3-2 Continued.

Core Infrastructure System Risk and Resilience Questions by Stage	
Stage	Question and Context
Mitigation-COOP	What mitigation options are most effective in reducing national, regional, or local risks from potential cascading or escalating failures and enhancing overall community resilience? Requires knowledge of operational environment requirements, dependencies, system geography, threat/hazard potential, consequences of disruption, and mitigation options.
Mitigation-COOP	What activities must occur to effectively respond to and recover from a disruptive event? What training is required to enable an optimal response?
Crisis Action	If an infrastructure failure occurs, what are the national, regional, and local impacts? What is the significance of the failure?
Crisis Action	How long until impacts of an infrastructure failure are realized? What mitigations are in place to buffer the event?
Crisis Action	Is there a potential for cascading or escalating failures?
Crisis Action	What activities are required to stabilize or recover service from a disrupted infrastructure system?

AHA Knowledge Graph

The AHA knowledge base leverages a knowledge graph paradigm to construct, represent, and provide interlinked and semantically rich infrastructure information for the purpose of conducting risk and resilience assessments of interconnected infrastructure systems. In total, the AHA knowledge graph contains seven dimensions which are a system/facility/asset taxonomy, sub-facility/component taxonomy, dependency-type (e.g., good and services) taxonomy, hazard-type taxonomy, functions, dependency profiles, owner/operator, and knowledge artifacts (i.e., sources). The knowledge graph was derived from functional decompositions of infrastructure systems and leverages a modified version of the functional basis for the engineering design proposed by [43] for the purpose of fault identification and propagation modeling [45]. Following [62], a knowledge graph is defined as

$$G = (\mathcal{E}, \mathcal{R}, \mathcal{F})$$

Where $\mathcal{E}$, $\mathcal{R}$, and $\mathcal{F}$ are sets of entities, relations, and facts. The following provides a description of the computational subgraph of the knowledge graph used for modeling and simulation of interconnected infrastructure systems.

The AHA computational knowledge graph is modeled as a directed multidimensional network or multigraph and, in its simple form, can be represented as [48]

$$G = (V, E, D)$$

Where *V* is a set of infrastructure nodes, *D* is a set of labels representing dependency types, and *E* is a set of labeled directed edges representing dependencies such as

$$E = \{(u, v, d); u, v \in V, d \in D\}.$$

Since the primary goal of the AHA methodology is to support the simulation of interconnected infrastructures behavior, this formulation needs to be extended to include both time-dependent behavior and strength of dependency. The objective is to determine the systems state under different disruption scenarios at discrete events that result in component state transitions. To address this requirement, the concept of a time-marked or temporal graphs as described in [49, 111] is leveraged, defined as

$$G = (V_c, E_d, D, C, t, x)$$

Where *V* is a set of nodes, *E* is a set of labeled directed edges representing dependencies, *C* is a set of system states, *D* is a set of labels representing dependency types, *t* is a time event, and $\mathcal{X}: V_c \longrightarrow C$ is a function that indicates node state. In this context, a node in *G* represents a computational element, and each edge represents the flow of a strength-based dependency type. The strength of dependency of each edge is modeled as weights shown below.

$$E = \{(u, v, d, c); u, v \in V, d \in D, c \in C\}$$

Where *c* is an integer number representing the strength of the dependency relationship between nodes $u, v \in V$ and labeled with both $d \in D$ and $c \in C$.

AHA Methodology

The AHA methodology is designed to facilitate developing dependency models of interconnected infrastructure systems to simulate the effects of disruptions on intra- and inter-system operations in support of risk and resilience decision-making. The methodology is divided into three processes shown in Figure 3-4: knowledge model development, dependency model development, and system

behavior simulation. These processes are aligned to the conceptual modeling, model coding, and experimentation processes described by [105]. The following subsection provides more detailed descriptions of each process.

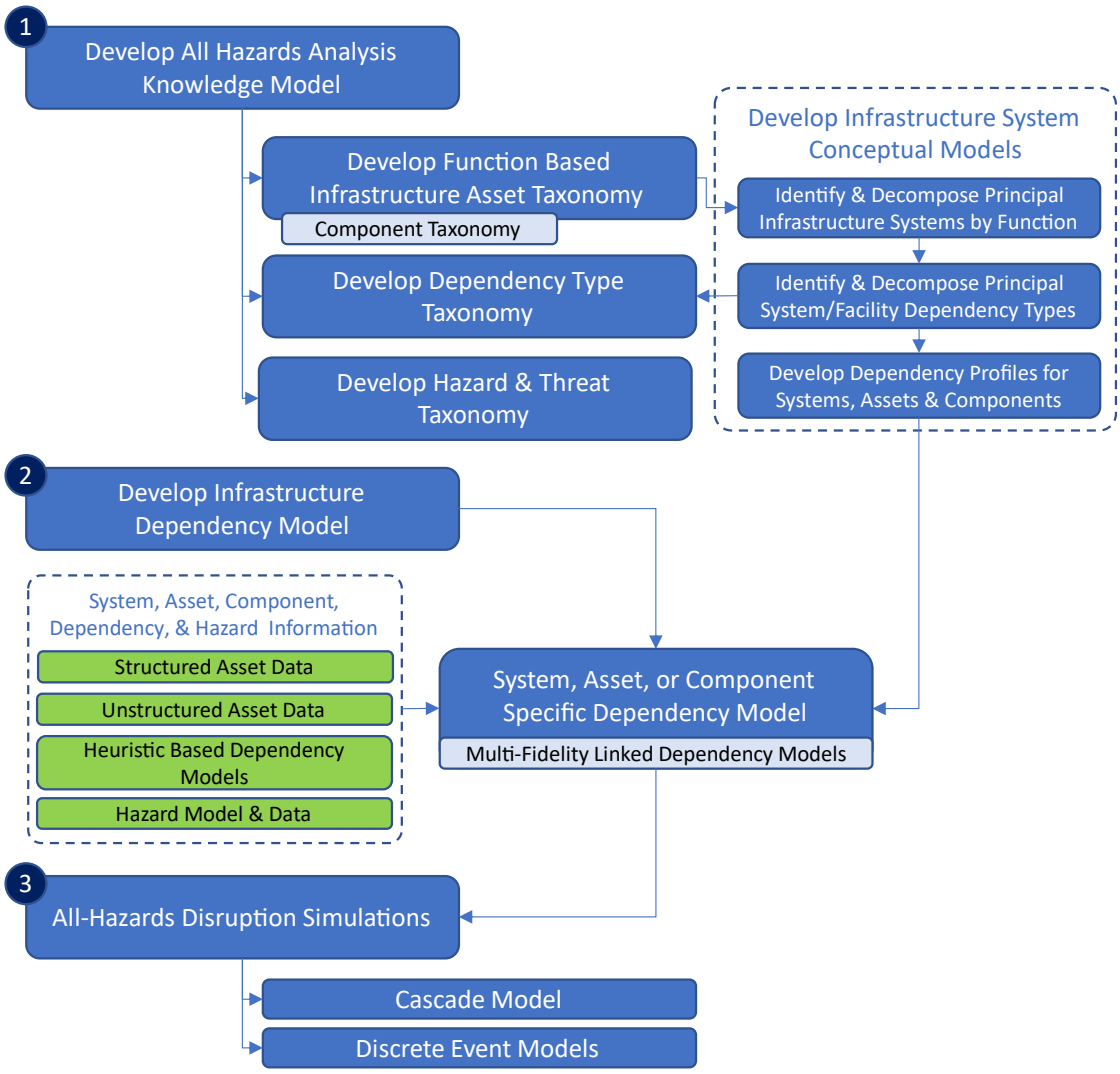

Figure 3-4 AHA Methodology.

AHA Knowledge Model Development Process

The primary objective of the knowledge model development process is constructing conceptual dependency models of general infrastructure systems, facilities, and components based on the infrastructure system's primary functions. The outcomes of this process are a function-based infrastructure asset/component taxonomy and a dependency-type taxonomy. When combined, these taxonomies result in a functional basis for engineered infrastructure systems. In addition, a hazard and threat taxonomy were constructed to achieve the overall purpose of the AHA methodology.

AHA Knowledge Model

The initial stage of the process is developing a function-based system/asset taxonomy which enumerates a standard set of functions and flows by asset types. This process is based on the concept of functional decomposition and leverages a modified version of the functional basis approach first proposed by Stone and Wood [30] which was later refined by Hertz et al. [31]. This results in a taxonomy of function-based asset models which are referred to as dependency profiles. Breaking down infrastructure systems in this manner provides a scalable systematic and precise mechanism to collect and communicate domain-specific SE knowledge [42, 44].

AHA Facility and Flow Knowledge Model

Structurally, the AHA function-based asset taxonomy is developed around the primary purpose or task (function) of a system and includes two primary structures, facility/sub-facility types (asset/components) and dependency types (flows). The system taxonomy is created through an iterative process that incorporates the following steps: (1) facility-type enumeration, (2) dependency-type enumeration, and (3) dependency profile generation which is described in greater detail below.

AHA Asset and Component Taxonomies

The facility-type enumeration step provides the ability to enumerate the type of facilities associated with a particular infrastructure system and is broken into two major categories: (1) facilities/assets and (2) sub-facilities/components. This step results in an ordered list or taxonomy of facility types that is used to enable the function of an infrastructure system. The taxonomy is modeled as an acyclic-directed graph where the nodes represent system, facility/assets, and component class objects, and the edges encode the subclass relationship as shown in Figure 3-5. This restricts the ability of facility or component type form being both parent and child node. The hierarchical approach allows for the inheritance of properties and aggregation of functions eliminating the need create entries for all potential functional combinations [55]. A facility type represents a major infrastructure type (e.g., data center), and a sub-facility represents internal components and devices which are used to facilitate operations of internal or external systems and facilities (e.g., uninterruptable power supply). Each type can be assigned specific properties which enable additional capabilities such as advanced modeling and simulation. These properties describe important characteristics about the facility type, such as storage capacity or generation capability.

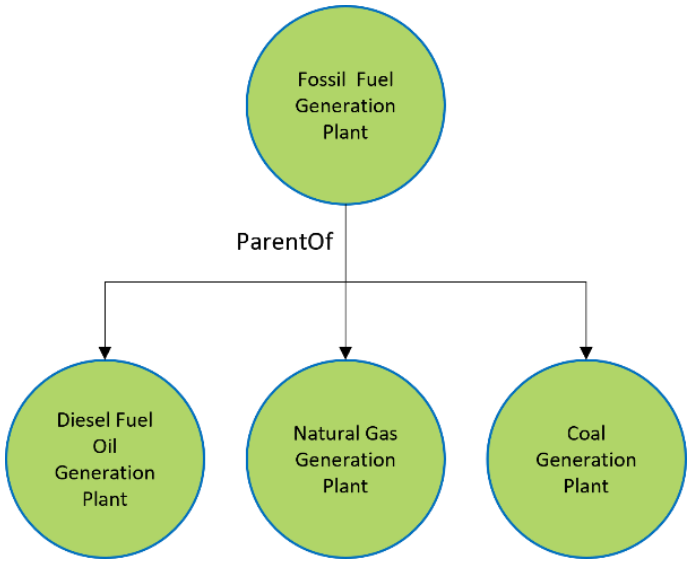

Figure 3-5 Facility/Asset Model.

AHA Dependency (Flow) Taxonomy

The dependency-type enumeration step provides the ability to enumerate the commodities, services, and data types that are required, transported, or produced by an infrastructure system, facility, or asset. Dependency types are broken down into three distinct categories: general, network, and transportable. A general dependency type represents a commodity or service that can be directly mapped between two facility types. A network dependency type represents a service network that can transport a transportable dependency type. For example, freight rail transport dependency type is a network and thus can be utilized to transport other commodities, such as agricultural products. Each dependency type can be assigned specific properties to enable additional capabilities such as advanced modeling and simulation.

AHA Dependency Profile Model

Dependency profiles are modeled after CFGs described in [112], which facilitate the creation of generic facility conceptual models that describe the range of inputs and outputs required by a particular system, facility/asset, or component/device type, and the relationship is described in a verb-object form (e.g., requires or provides electricity). Definition of function classes are provided in Table 3-3. Dependency profiles represent black boxes of operational flows of commodities and services, and as implemented, dependency relationships are passed up the taxonomic tree to facilities comprehensive sector and system-level profile development. Further, with each dependency relationship, a categorical measure of criticality is assigned based on the following DoD protection failure criticality levels [113] as shown in Table 3-4, which used to model impacts functional degradation utilizing FFL described in [45]. It is important to note these are guides and can be altered

when developing facility-specific dependency relationships between actual facilities. An example dependency profile is shown in Figure 3-6.

Table 3-3 Function Class Definitions.

Class	Graph Property	Definition	Source
Require (Import)	Dependent On	To bring in a flow (material, energy, signal) from outside the system/asset/component boundary. Example: a natural gas generation plant imports dry natural gas into the facility.	Adapted from [30].
Provide (Export)	Provider of	To send a flow (material, energy, signal) outside the system/asset/component boundary. Example: a natural gas compressor station exports dry natural gas into a natural gas transmission pipeline.	Adapted from [30].
Provide	Provider of: Source	To produce a flow (material, energy, signal) for the purpose of exporting. Example: a natural gas processing plants produces dry natural gas.	—
Provide	Provider of: Store	To accumulate a flow for later use. Example: a refined fuel terminal stores refined fuels.	Adapted from [30].
Provide	Provider of: Transport	To move or convey a flow from one system/asset/component to another system/asset/component.	Adapted from [30].

Table 3-4 AHA Criticality Levels.

Name	Criticality Level	Definition
Critical/Facility Down	4	Production down or major malfunction resulting in an inoperative condition. Operators are unable to reasonably perform their normal functions. Consumers without service. The specific functionality is mission critical to the system, and the situation is considered an emergency.
Significant Impact	3	Critical loss of functionality or performance resulting in abnormal operation. Operators are unable to perform their normal functions. Major feature/product failure; inconvenient workaround or no workaround exists. The facility is usable but severely limited. Consumers with limited or impaired service.
Moderate/Minor Impact	2	Moderate loss of functionality or performance resulting in abnormal operations. Operators impacted in their normal functions. Minor feature/product failure; convenient workaround exists/minor performance degradation/not impacting production.
Low/No Impact	1	Minor loss of functionality, product feature requests, how-to questions. The issue consists of "how-to" questions including issues related to one or multiple modules and integration, installation, and configuration inquiries, enhancement requests, or documentation questions.

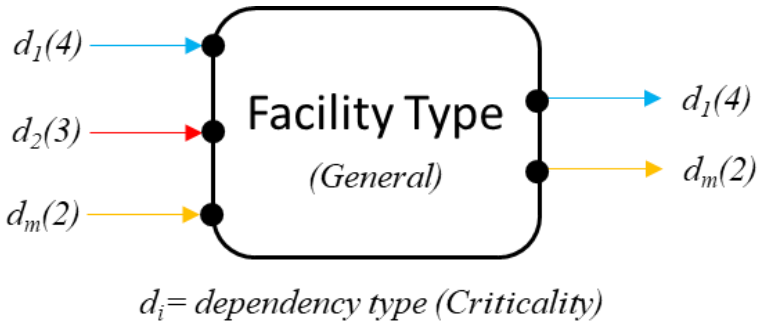

Figure 3-6 Generic Dependency Profile.

Finally, each output relationship is assigned a functional type, which includes source, storage, and transport (default). Source type indicates a facility or device type can produce a commodity, service,

or datum. Storage type indicates a facility or device type can store a commodity, service, or datum. Transport type indicates a facility or device type simply passes or diverts a commodity, service, or datum.

When carefully applied, this stage results in a comprehensive, scalable, and non-redundant functional basis of an infrastructure system, and when combined with coherent definitions, it provides a universal assessment language and conceptual model. Further, this process can be used to create conceptual models or generic, integrated CFGs of infrastructure systems; for example, a generalized version of the electric system is shown in Figure 3-7.

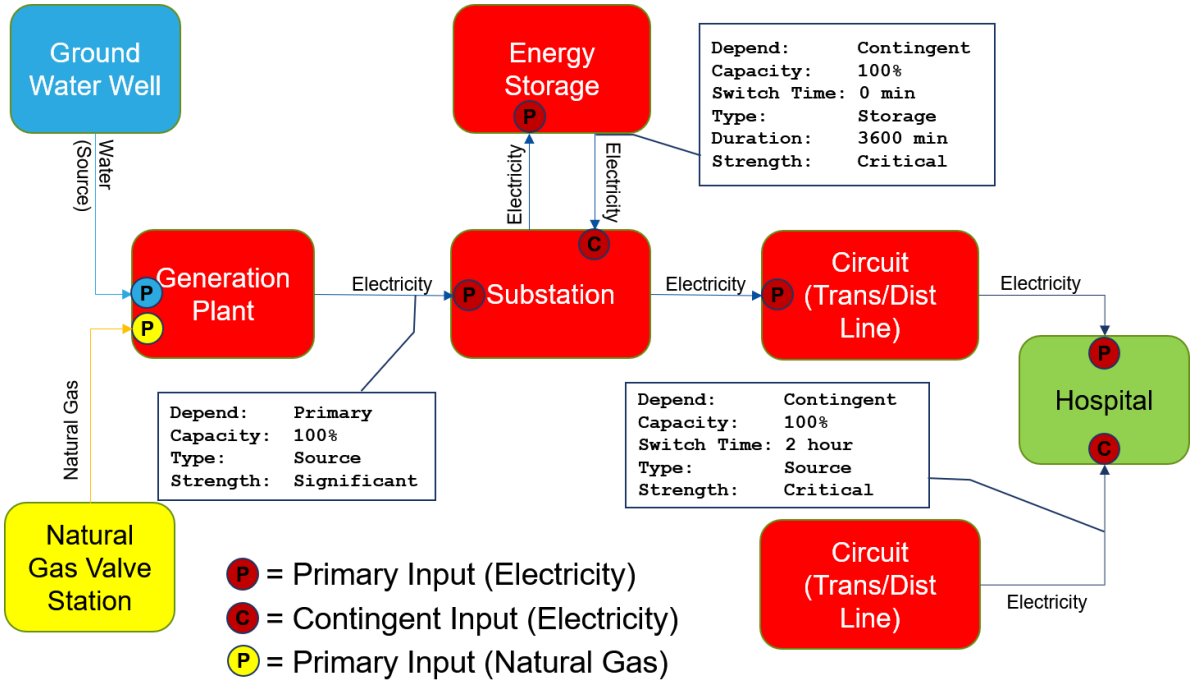

Figure 3-7 Generalized Electric System Conceptual Model.

AHA Dependency Profile Component Models & Templates

Dependency profile component models represent internal systems that support a facility's function and are generally composed of multiple components. For example, a back-up power model could be developed to simulate the transition from commercial power to generator power. If applicable, a model can be converted to a template to facilitate reuse of the component across similar facilities or facility types. An example of component model template is shown in Figure 3-8.

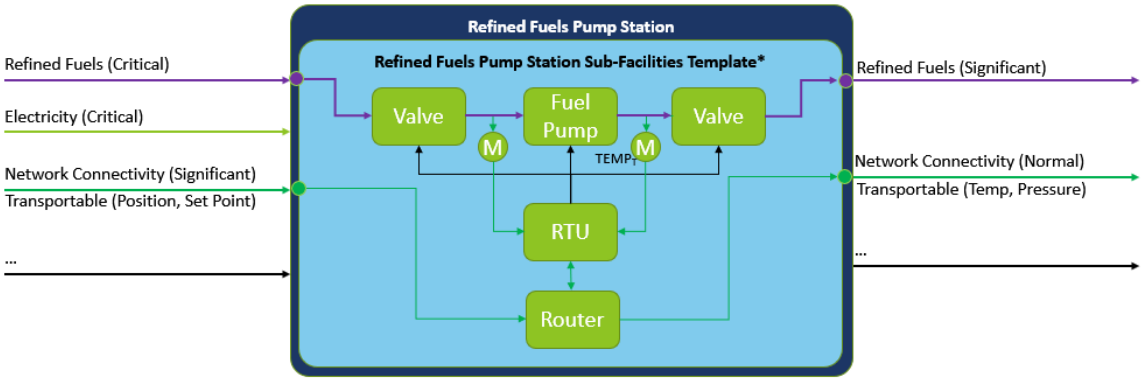

Figure 3-8 Notional Refined Fuels Pump Station Component Model.

Threat and Hazard Model

The threat and hazard model describes the taxonomic structure of the threat and hazard domain relevant to critical infrastructure, where the resulting categories can be assigned to specific systems, facilities/assets, and components/devices. In addition, each specific threat and hazard type is assigned a default risk level. Risk levels were derived from NIST's *Guide for Conducting Risk Assessments* and are based on the likelihood of the threat or hazard event resulting in adverse impacts (see Table 3-5) [114]. Actual facility and component risk levels can be adjusted during the risk analysis phase. The taxonomic structure consists of three primary categories: cyber, human-induced, and natural hazards. The following subsections provide additional information on each of the categories.

Table 3-5 Likelihood of Threat and Hazard Impact Levels [114].

Qualitative Values	Semi-Quantitative Values		Description
Very High	96–100	10	If the threat event is initiated or occurs, it is **almost certain** to have adverse impacts
High	80–95	8	If the threat event is initiated or occurs, it is **high likely** to have adverse impacts
Moderate	21–79	5	If the threat event is initiated or occurs, it is **somewhat likely** to have adverse impacts
Low	5–20	2	If the threat event is initiated or occurs, it is **unlikely** to have adverse impacts
Very Low	0–4	0	If the threat event is initiated or occurs, it is **highly unlikely** to have adverse impacts

Cyber Hazard Model

The AHA cyber hazard model leverages MITRE's adversarial tactics, techniques, and common knowledge (ATT&CK) and ATT&CK-industrial control systems (ICS) models to define the cyber hazard types [115-117]. The ATT&CK framework is a knowledge base of enumerated cyber adversary behaviors and builds upon the concepts like Lockheed Martin's intrusion kill chain [118]. Its design is intended to aid network defenders in developing behavior analytics and assessing defense gaps. This is accomplished by the decomposition of actual observed adversarial behaviors into tactics and techniques. Tactics describe the attacker's objectives, and techniques describe how the attacker achieves them. This structure aligns directly to the attack phase paradigm to organize the techniques that may be used to compromise cyber system components or services enabling business and operations functions.

Artificial Hazard Model

The AHA artificial hazard model leverages Federal Emergency Management Agency's *Threat and Hazard Identification and Risk Assessment and Stakeholder Preparedness Review Guide* to provide an initial model for artificial threat and hazard enumeration [5]. However, as noted above, the cyber threat and hazards are modeled separately.

Natural Hazard Model

The AHA natural hazard model leverages the comprehensive review of natural hazards conducted by Gill and Malamud to provide the base enumeration of 21 different natural hazards which were categorized into six distinct hazard groups [119]. Their taxonomy was extended and refined to incorporate additional hazards enumerated by the International Panel on Climate Change [120, 121].

AHA Infrastructure Dependency Knowledge Base

The second stage of the AHA methodology is developing system- and facility-specific dependency models, which are modeled as a directed multidimensional network [48] leveraging a modified version of the approach described by Svendsen and Wolthusen [51]. A graph data model was selected as the primary method to represent the AHA knowledge base because interconnected infrastructure systems can be intuitively represented as graphs. Further, graphs provide ideal structures to represent relationships between entities such as an organization's ownership of an infrastructure system.

This stage is a three-step process that requires the loading of system facilities/components, which represent the vertices of the graph, assignment of dependency links between system facilities, and assignment of intersystem dependency links. This includes the initialization of storage and link

parameters. A comprehensive mapping will trace the commodity, service, or datum type from the time it enters until it is converted or exits the system or facility.

Facility/Asset Loading

The first step in developing a system-specific dependency model is the loading of system facilities and components. During the loading process, individual facilities will receive their taxonomic assignment, and they will inherit the respective dependency profile. In addition to facility type, the following elements of information can also be included during the loading process: name (required), alias, owner, operator, address, zip, state, county, country, confidence, latitude (required), and longitude (required). Confidence category assignments are assigned to each facility based on the underlying source information and include the values shown in Table 3-6. Depending on available information, specific properties and profile exceptions can also be defined. For example, the production capacity of a petroleum refinery or the IP address of a server could be included.

Table 3-6 Knowledge Model Confidence Levels.

Confidence Type	Level	Definition
Vetted	4	Infrastructure owner/operator has recently confirmed information.
High	3	Information has been confirmed by infrastructure owner/operator in the past or derived from recently published and openly available owner/operator or derived for recent provided regulatory data.
Moderate	2	Information has been published in the past by owner/operator or in regulatory data or derived from recently published third party sources. Some heuristics.
Low	1	Outdated information or heuristics.

Dependency Model Generation

The second step in this stage is the assignment of dependency relationships between facilities based on their respective profile. Profiles enforce how a facility functions and defines its dependency interfaces resulting in a consistent and defensible model. Creating a dependency relationship between facilities will require that one of the facilities can provide a commodity or service, while the other requires the same commodity for operations. As implemented, the AHA application automatically enforces these rules and presents potential facility relationships in an ordered list by distance. Configurable dependency relationship parameters are defined in Table 3-7.

Table 3-7 Standard Dependency Relationship Parameters.

Dependency Parameter	Definition
Strength/Criticality Level	As defined above and can be inherited from the profile or overridden based on facility-specific information.
Confidence	As defined above.
Precent Commodity	Defines the degree to which a specific dependency relationship can provide the entire required amount for normal operation.
Contingency Type	Categorizes a dependency relationship as either primary or contingent. Dependency relationships are considered primary by default. Contingent relationships are alternate sources that are used in the event a primary dependency source is disrupted. A contingent relationship requires a time-to-switch variable to be set.
Storage Duration	Number of minutes a dependency relationship can be maintained after initial disruption if defines as a storage type dependency.

The third and final step of regional dependency model creation is the aggregation of distinct system dependency model into a single cohesive representation. The result is a functional dependency model of interconnected infrastructure systems, with documented spatial and temporal characteristics directly related to system operations. The resulting model enables direct simulation of system behavior based on steady-state design parameters.

AHA Knowledge Base Metamodel

The AHA knowledge base metamodel serves as a supporting capability to increase confidence in the analytic and simulation outcomes of the AHA methodology. The metamodel leverages the Office of the Director of National Intelligence to Intelligence Community Directive 206, “Sourcing Requirements for Disseminated Analytic Products,” as a guide to implementation and seeks to capture the sourcing information for the asset, component, and dependency information contained within the AHA knowledge management system [122]. Source information captured includes the title, description, publisher (owner/author of data), source (URL [universal resource locators] or repository), data of information access, date of information, sector tags, analyst-defined tags, base document/information product, and refined information product. This information is used to enhance the credibility and transparency of the dependency model and simulation analysis outcomes for risk and resilience decision-making processes.

As described in Table 3-6, source information is utilized to establish confidence levels of the model elements, and parameters are recorded in the AHA knowledge graph. Confidence levels are determined by the pedigree of the source documents or heuristics used to identify the infrastructure or its assigned dependency relationships. If source information is explicitly identified in an artifact, it is assigned a confidence value based on the date of the information. If the source information is established through heuristics, it receives a low confidence rating.

All-Hazard Disruption Simulation Techniques

The AHA methodology and knowledge management system were designed to provide a flexible set of simulation capabilities to inform the risk and recovery decision-making processes of federal, state, and local risk and emergency management organizations, as well as infrastructure owners based on their best available data. The approach seeks to provide an integrated simulation platform that draws from a common topology model of interconnect infrastructure systems to generate executable simulation models for the desired system or region of interest required to inform the decision-making process. To accomplish this objective, the modeling principles outlined by [111, 123] act as guidelines to our approach and include:

- Keeping the models as simple as possible to meet simulation objectives (Occam's razor)
- Promoting ease of analysis, limiting computational complexity where possible
- Allowing for versatility and extendibility
- Promoting ease of result interpretation.

The following subsections describe the initial set of simulation approaches that have been fully implemented within the framework thus far.

Simple Cascade Simulation

The simple cascade simulation is a qualitative approach to evaluating the cascading impacts of infrastructure disruptions and is intended to enable a range of influence for the analytic results best suited for crisis-action situational awareness and initial strategic risk and resilience assessments. The technique integrates and simplifies the approaches described by [45, 124] to reduce the model parameter requirements while still providing sufficient information to begin addressing many of the core questions outlined in Table 3-2 above. The approach implements the concept of FFL described in [45] and the strength of dependency described in [124] without the consideration of time, contingent dependency relationships, or component-level mitigations. In this manner, the asset state transitions are instantaneous and are determined entirely by the strength of the dependency value encoded on the output dependency relationships from the initial disruption. Accordingly, the system

states that are described by a disruption simulation can be used to answer questions about the range and degree of impact. The pseudo code of the algorithm is shown in Figure 3-9 below, and the verification model used to test the proposed model logic is presented in Figure 3-11.

Algorithm 1 Simple Cascade

Input:	User selected graph ***G***	
	User selected disrupted node ***V***	
Output:	Impact graph ***I*** such that ***I*** is a subset of ***G***	
Variables:	DP^i	Set of all O-D pairs related to impacted node
	d_i	O-D pair in the set DP^i
	O^S	Origin state as defined by state values
	D^S	Destination state as defined by state values
	d_i^{CT}	Dependency type continuity category
	d_i^{CL}	Strength of dependency as defined by state values
State Values:	5: Disrupted, 4: Critical, 3: Significant, 2: Moderate, 1: Low, 0: Steady State	
Contingency Type (CT):	Primary, Contingent	

Simple Cascade Function-Failure Logic (***G, O***)

1: **Get** DP^i **(*V*)**
2: **For Each (*d*)** in DP^i
3: Get O^S (***O***), ***CT*** (d_i), ***CL***(d_i), D^S (***D***)
4: If ***CT*** = Primary
5: If D^S < Min (O^S, ***CL***)
6: Set D^S = Min (O^S, ***CL***)
7: Simple-Cascade FFL (D)
8: **Return *I***

Figure 3-9 Algorithm 1: Simple-Cascade ().

Time-Dependent Cascade Event Simulation

The time-dependent cascade event simulation is an event-driven semiquantitative approach that augments the range-of-influence simulations with time to better understand failure propagation both within and between systems, facilities, and components. The technique incorporates concepts from both systems dynamics and DES to form a hybrid solution to provide estimates of time-to-impact based on a predefined user scenario. Time is addressed with storage duration and time-to-switch parameters of contingent relationships which buffer state transitions of nodes in the initial graph depending on the offset from the initiating event. In addition, the approach enforces the requirement

of the initial graph to contain a source or temporary source node for all dependency types included. The pseudo code of the algorithm is shown in Figure 3-10 below, and the verification model used to test the proposed model logic is presented in Figure 3-11.

Algorithm 2 General Time-Dependent Cascade

Input:	User selected graph $\boldsymbol{G}$	
	Set of user defined events $\boldsymbol{EL = \{EV_1(V, ET, t), ..., EV_i(V, ET, t)\}}$	
Output:	Set of impact graphs $\boldsymbol{I_i}$ such that $\boldsymbol{I_i}$ is a subset of $\boldsymbol{G}$ at time t	
	Master Scenario Event List (***MSEL***)	
Variables:	$\boldsymbol{t}$	Simulation Time
	$\boldsymbol{DP^i}$	Set of all O-D pairs related to impacted node
	$\boldsymbol{d_i}$	O-D pair in the set $\boldsymbol{DP^i}$
	$\boldsymbol{O^S}$	Origin state as defined by state values
	$\boldsymbol{D^S}$	Destination state as defined by state values
	$\boldsymbol{d_i^{DT}}$	Dependency type
	$\boldsymbol{d_i^{CL}}$	Strength of dependency as defined by state values
	$\boldsymbol{d_i^{CT(TTS)}}$	Dependency type continuity category
	$\boldsymbol{d_i^{RL}}$	Contingency Restoration Level
	$\boldsymbol{d_i^{ST(SD)}}$	Dependency type source category
	TTS	Time-to-switch in minutes
	SD	Storage Duration in minutes
State Values(*S*):	7: Disrupted, 6: Storage Depleted, 5: Discharging, 4: Critical, 3: Significant, 2: Moderate, 1: Low, 0: Steady State	
Criticality Levels (CL)	4: Critical, 3: Significant, 2: Moderate, 1: Low, 0: Steady State	
Contingency Type (*CT*):	Primary, Contingent (TTS)	
Source Type (*ST*):	1: Source, 2: Transmission, 3: Storage (SD)	
Event Type (*ET*):	Disruption, Restoration	

Time-Dependent Cascade Function-Failure Logic (***G, S***)

1: **Validate Source Nodes Exist (*G*)**
2: **Generate MSEL (*S*)**
3: **For** ***EV*** in ***S (Min (t), Max (t))***
4: **if** ***ET*** **= Disruption**
5: **Disrupt (V, 7)**
6: **if** ***ET*** **= Restoration**

Algorithm 2 General Time-Dependent Cascade Continued

```
                    Restore (V, 0)
          Clean Events
Disrupt (V, S)
          Set O^S = S
          For Each (d_i) in DP^i
                    If CT = Primary, Set D^S = Max (D^S, CL)
                    If D → DT & D^(ST = 3)
                       If DT source unreachable
                          Add Storage Depletion Event (D^S, SD)
                    If CT = Contingent & Primary d_i disrupted
                          Add Restore Event (D^S, RL, TTS)
                    Disrupt (D, DS)
Restore (V, S)
          Set O^S = max(←CL)
          For Each (d_i) in DP^i
                    If CT = Primary, Set D^S = Max (D^S, CL)
                    If D → DT & D^(ST = 3)
                       If DT source unreachable
                          Add Storage Depletion Event (D^S, SD)
                    If CT = Contingent & Primary d_i disrupted
                          Add Restore Event (D^S, RL, TTS)
                    Disrupt (D, DS)
```

Figure 3-10. Algorithm 2: Time Dependent-Cascade ().

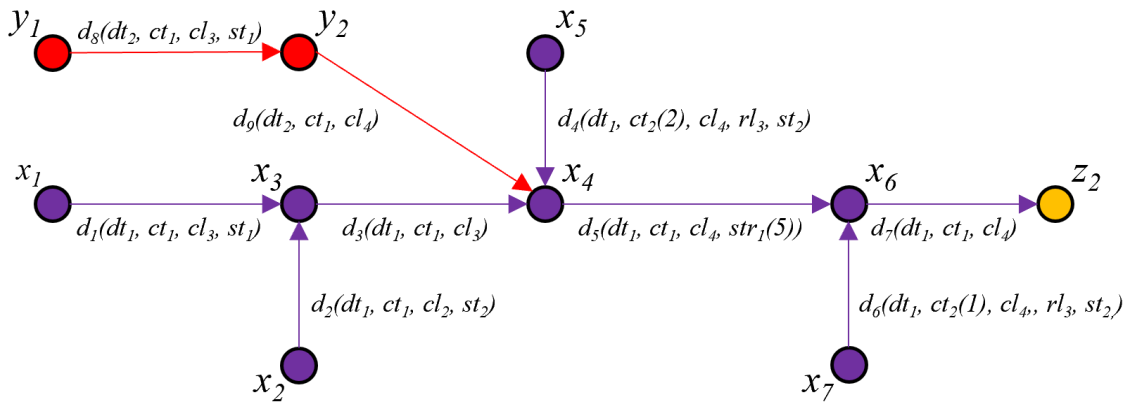

Figure 3-11 Verification and Validation Graph.

Table 3-8 Node Event and State Table.

	t=0	*t=1*	*t=2*	*t=3*	*t=4*	*t=5*	*t=6*	*t=7*	*t=10*
x_1	(0, 0)		D(2, 5)				R(6, 0)		
x_2	(0, 0)	D(1, 5)							R(10, 0)
x_3	(0, 0)	(1, 2)	(2, 4)				(6, 2)		(10, 0)
x_4	(0, 0)	(1, 2)	$SD_{1,t=(t+5)}$ $C_{1t=(t+2)}$	(3, 4)	R_1(4, 3)	(5, 3)	(6, 2)	SD(7, 4)	(10, 0)
x_5	(0, 0)								
x_6	(0, 0)	(1, 2)	(2, 3)	$C_{1,t=(t+1)}$	R_2(4, 3)		(6, 2)		(10, 0)
x_7	(0, 0)								
y_1	(0, 0)								
y_2	(0, 0)			D(3, 5)		R_3(4,0)			
z_1	(0, 0)	(1, 2)	(2, 3)	(3,4)	(4, 3)		(6, 2)		(10, 0)

Chapter 4: AHA Knowledge Management System

In this chapter, the AHA knowledge management system (KMS), designed around the concept of a dynamic and function-based infrastructure system data model which can be represented as a multilayer network, is described. The process-centered ontology-driven approach of the AHA-KMS represents information about entities in the form of nodes (e.g., infrastructure facilities and organization), links (e.g., dependency relationships), and specific properties/attributes (e.g., labels) which describe characteristics of an entity or relationship as a knowledge graph. The flexible knowledge graph structure provides the ability to incorporate multiple existing infrastructure schemas, such as the DHS infrastructure taxonomy [125] or generate new custom schemas. The dynamic framework also allows the base knowledge model to be modified to enable the rapid capture of new infrastructure types, dependency types, and properties as well as information related to organizational and mission support. The dynamic nature of the knowledge model is critical to address risk and resilience decision processes with respect to the evolving infrastructure and threat and hazard landscapes. The following sections of this chapter describe the overall architecture and major elements of the AHA-KMS.

AHA Framework Architecture

Figure 4-1 illustrates the overall system design of the AHA-KMS which include six distinct dimensions that are designed to enhance knowledge capture, analysis, and visualization of infrastructure systems information. Each of the components will be described in greater detail in the following sections of this chapter.

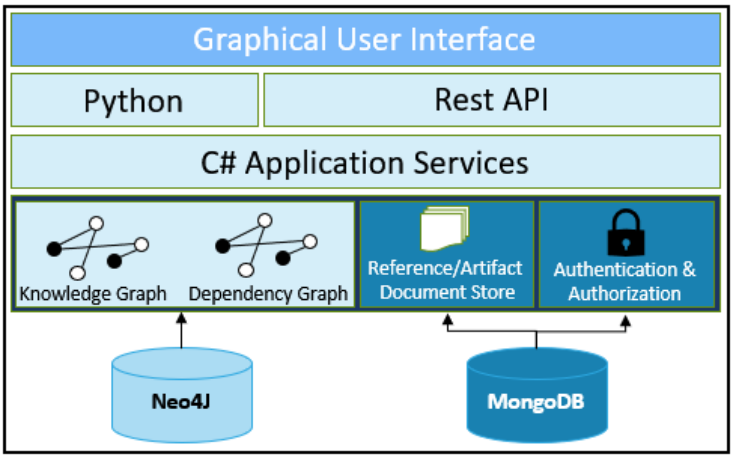

Figure 4-1 AHA Architecture Diagram.

To facilitate efficient IR and analysis of the multilayer network knowledge structure of the AHA-KMS, it leverages both a Neo4j graph store and MongoDB document store database technology.

All-Hazards Ontology and Knowledge Graph Module

The all-hazards ontology and knowledge graph module is the core element of the AHA-KMS and was designed to enable the dynamic development of the AHA knowledge graph structure. Thus, the module supports the design and development of the asset, component, dependency type, and threat/hazard taxonomies, as well as dependency profiles as described in the "AHA Knowledge Graph" section above. A subgraph of the AHA knowledge graph from the Neo4j database is shown in Figure 4-2. Currently, the knowledge base contains 330 unique infrastructure types and 288 unique dependency types. These have resulted in the development of 330 dependency profiles that describe the general functional requirements and outputs of infrastructure assets and component types.

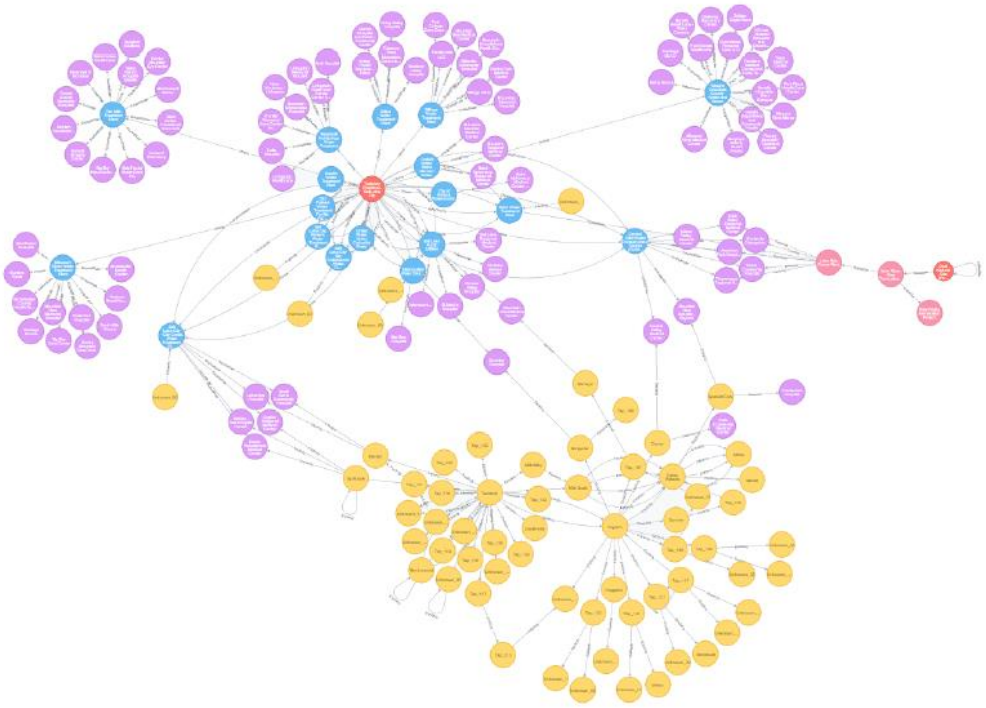

Figure 4-2 Subgraph of the AHA Knowledge Graph.

The development of the knowledge graph is facilitated by the GUI shown in Figure 4-3. Part 1 (see first blue circle below) provides a selectable tree view representation of the facility/asset, sub-facility/component, dependency (flow), and threat and hazard knowledge object models depending on which menu item is selected (see part 2). Part 3, shown in its collapsed form, contains the asset type name, description, and source information used to generate the asset type. In addition, the three-dot icon to the right of the name is clickable and will expose additional editing and visualization capabilities which will be described in more detail below. Part 4 is the specific properties edit and visualization window, clicking on the blue plus sign (+) on the right side will allow new specific properties to be associated with the facility type. Part 5 is the dependency-type association edit and visualization window, clicking on the blue plus sign (+) on the right side will allow an analyst to associate additional dependency types defined in the dependency-type taxonomy with the profile. For dependent-on dependency-type associations, a general dependency strength must be defined. For provider-of associations, the general strength and source type (i.e., source, transmission, and storage) are defined. If the asset type selected has additional children facility types defined, their profile dependencies will be shown in part 6; however, they are not editable on the parent's profile page. Additional parts not shown in Figure 4-3 include the threat and hazard association and the sub-facility/component template editing-and-visualization window.

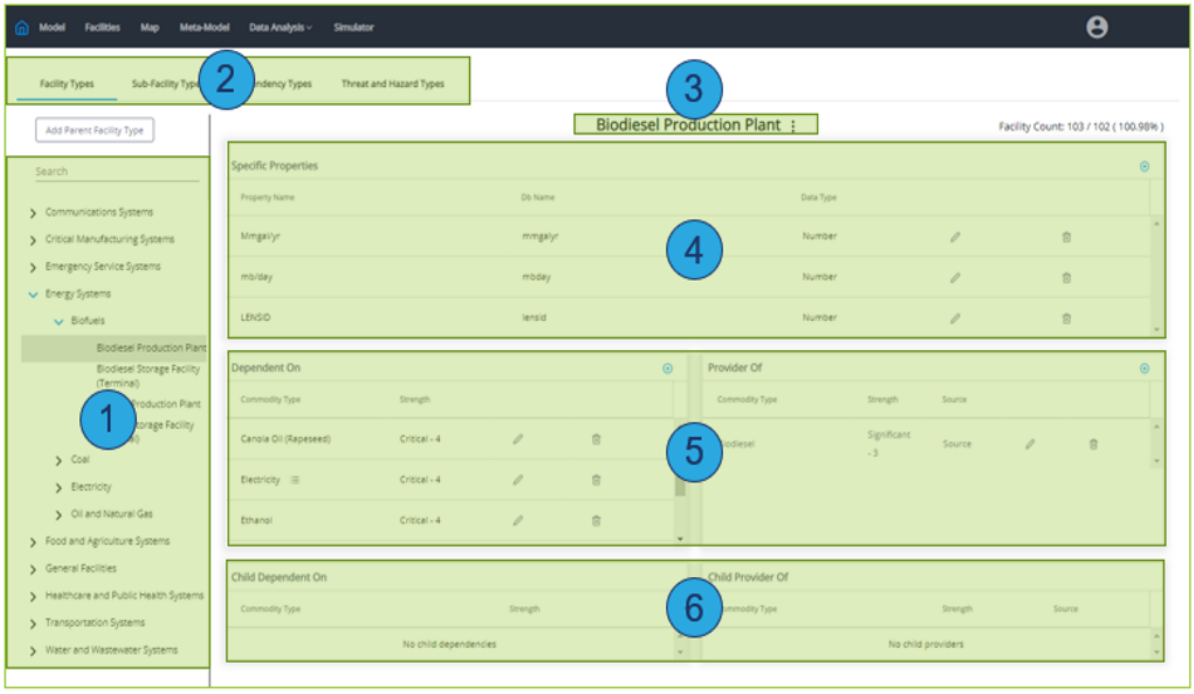

Figure 4-3 Knowledge Graph Graphical User Interface.

Knowledge Base Population

The information and knowledge generation module was designed to enable the loading and translation of infrastructure information from multiple sources. The core capabilities of this module are intended to enhance loading of both structured and unstructured information artifacts. The "AHA Methodology" section describes the methodology used as a guide for the population of the AHA knowledge base. Currently, there are over 1.3 million systems, assets, and components contained in the knowledge base with approximately 1 million unique dependencies.

The development of the knowledge graph is facilitated through multiple GUIs that support both manual and bulk loading of infrastructure assets and dependencies. Data checks are conducted during the load process to ensure that the required elements of information are present, and a source has been identified to support the data element as described in the "AHA Infrastructure Dependency Knowledge Base" section. Figure 4-4 shows the manual entry form for loading asset-level data.

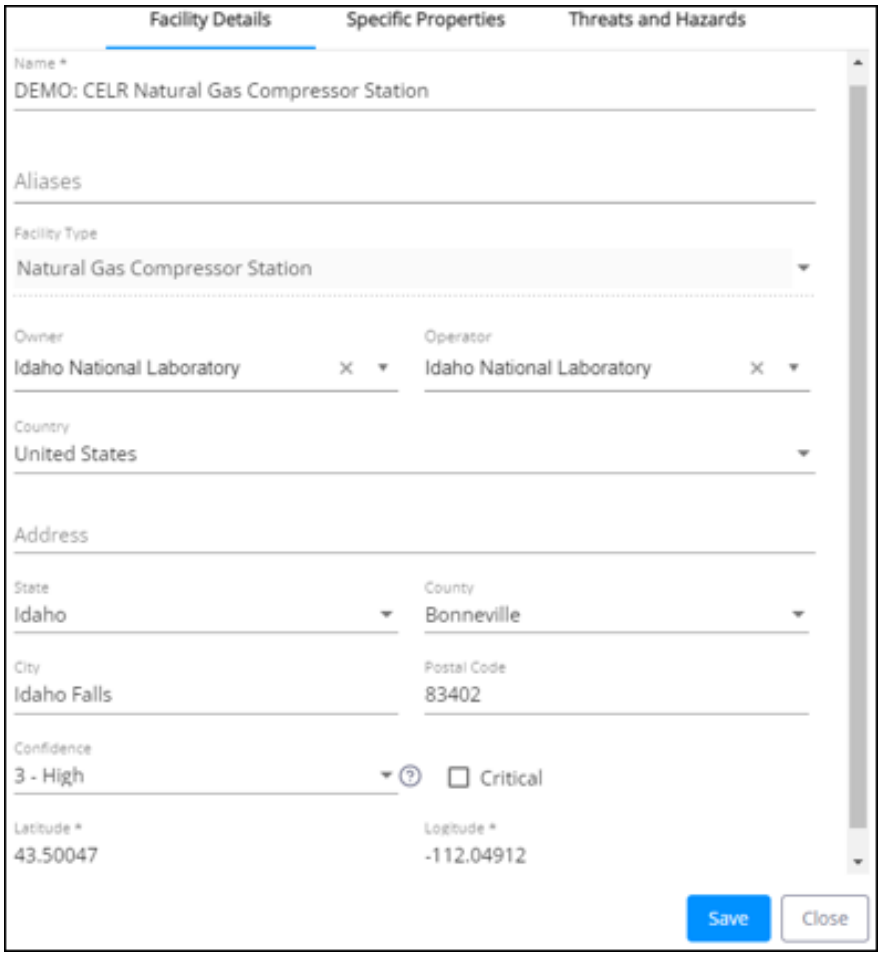

Figure 4-4 Manual Facility Asset Entry Screen.

Infrastructure System and Dependency Models

The infrastructure system and dependency models are constructed from the system, asset, and component information stored within the knowledge base as multilayer networks of interconnected infrastructure to enable simulation and analysis activities. “Chapter 3: All-Hazards Analysis (AHA) Methodology” describes the AHA infrastructure system and dependency knowledge model development process. Dependency models are dynamically generated from the knowledge base through analyst-defined queries and can be stored for later use or shared between users of the system. In addition, generated models are utilized for the time-dependent cascade simulation project files. Figure 4-5 depicts a simple three-node dependency model generated from an owner query; the nodes represent facilities, and the lines represent dependency between facilities. The pink-highlighted node indicates the selected facility. It should be noted that by examining the dependency graph at the bottom of the GUI, an additional first order dependency for the selected infrastructure has been

defined in the knowledge base. To add this entity to the selected model, the user would increment the upstream dependency parameter to one (1) and resubmit the query.

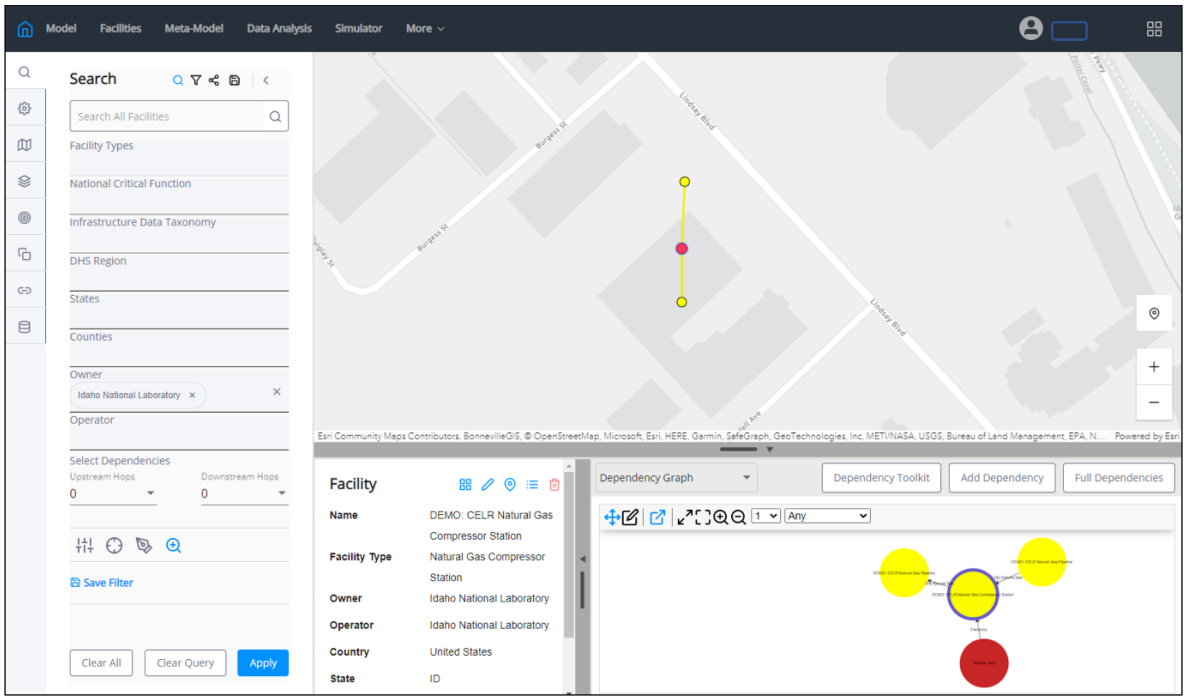

Figure 4-5 Example of an AHA Infrastructure Dependency Model via Map GUI.

Simulation and Analysis

The simulation and analysis capability was designed to simulate the behavior of interconnected infrastructure systems in support of all-hazard risk and resilience assessments to include the identification of systematically important critical infrastructure (SICI). This capability consists of qualitative system behavior/cascade simulation and a semiquantitative time-dependent cascade simulation depicting state transitions at discrete time events. The "AHA Infrastructure Dependency Knowledge Base" section describes in detail the AHA simulation and analysis approach.

The simulation capabilities transform the facility-type multilayer representation into facility-state multilayer graph representation. The simple cascade simulation can be run directly from the map GUI as shown in Figure 4-6 and Figure 4-7. Figure 4-6represents steady-state operations, and Figure 4-7 represents the post-disruption state. In this case, the substation powering a natural gas compressor station has been disabled (black node), and the cascading effects are shown in red, indicating critical impacts at the compressor station and the downstream natural gas pipeline. The FFL is described in "Chapter 3: All-Hazards Analysis (AHA) Methodology." Simulation results are reported in data tables to support development of additional analytic products and decision-making requirements.

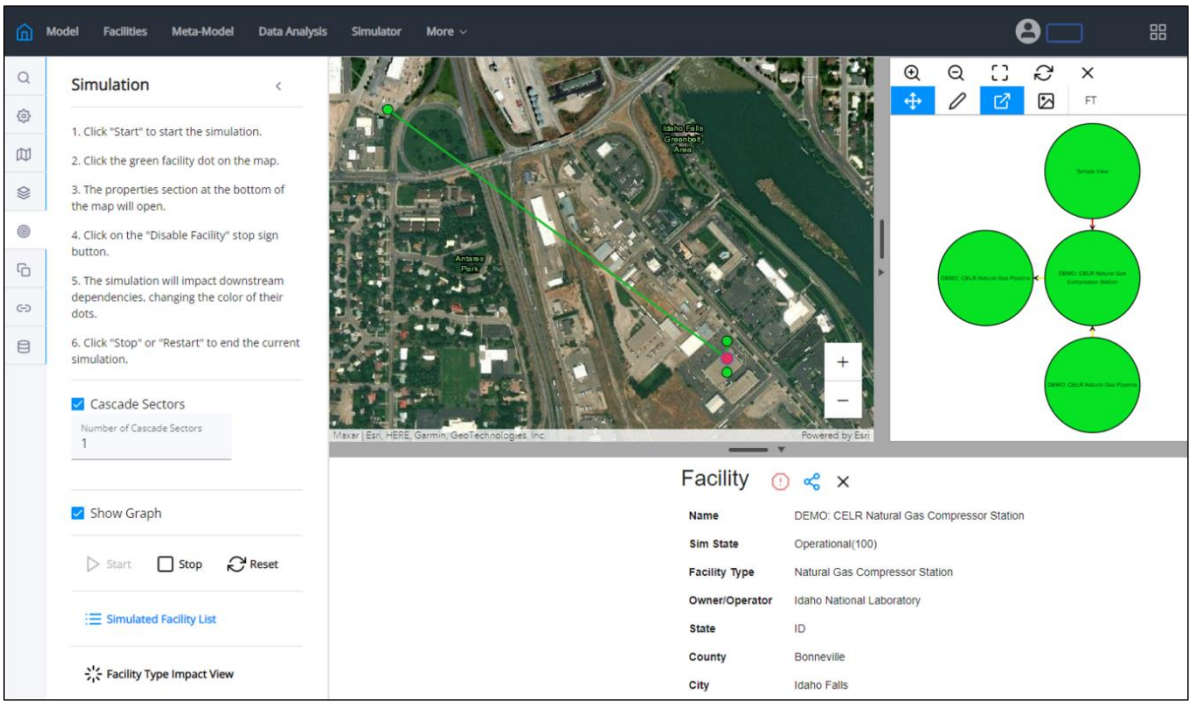

Figure 4-6 Simple Simulation GUI: Prior to Disruption.

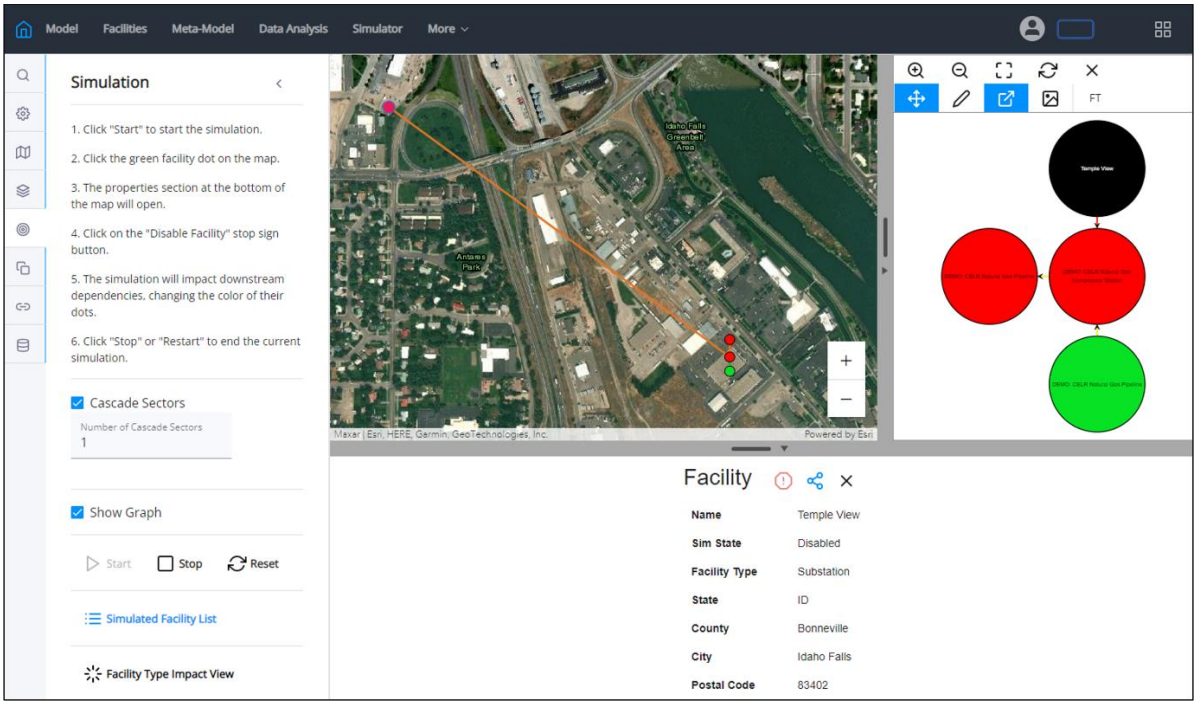

Figure 4-7 Simple Simulation GUI - Post Disruption.

The time-dependent cascade simulations are run from a dedicated GUI due to the configuration required for each project developed. Step 1 for running a time-dependent cascade is to select and validate a simulation graph, which includes ensuring all dependency types have a source, or a

temporary source assigned as described in “Chapter 3: All-Hazards Analysis (AHA) Methodology.” After validation, user-defined events are created. A minimum of one disruption event is required as shown in Figure 4-8. In this example, two substations are disable, one at 15 minutes and another at 18 minutes. It is also possible to select facilities based on their assigned hazard risk levels or by a map-based spatial query. If desired, restore points may be set through the same interface.

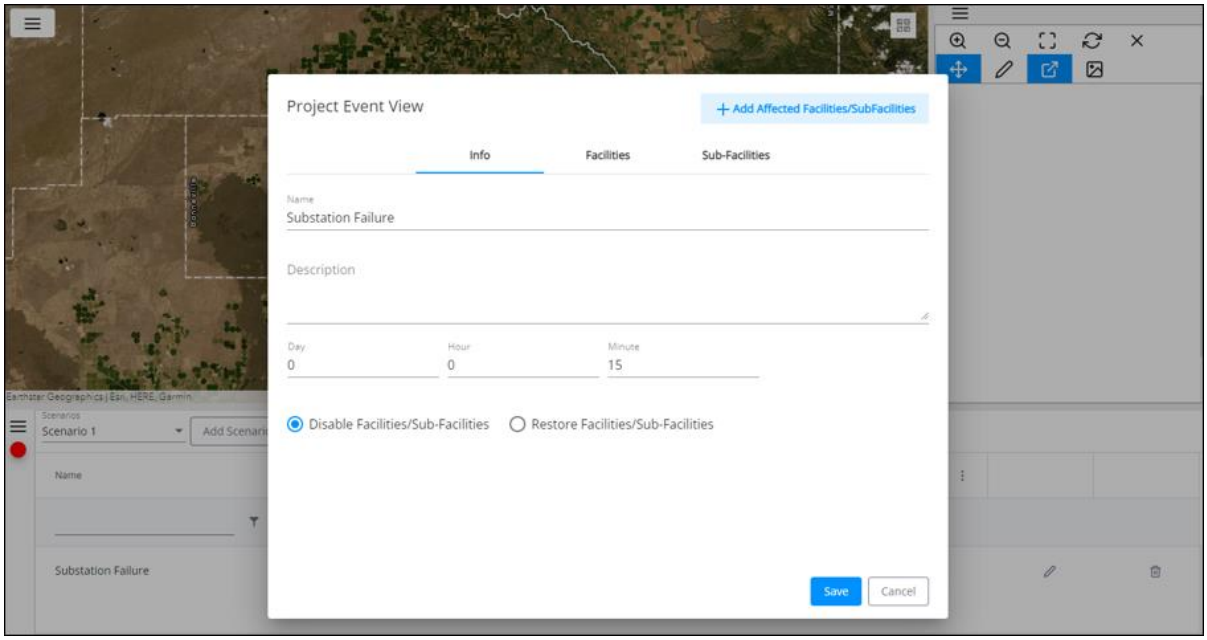

Figure 4-8 Time-dependent Cascade Simulation Configuration.

Step 2 is creating the master event list, which is derived from a union of the user-defined events with the asset and component storage and time-to-switch parameters. This is transparent to the user and results in the creation of the simulation timeline which can be played by the users; in this case, five events are generated from the disruption of the substations. The GUI is shown in Figure 4-9.

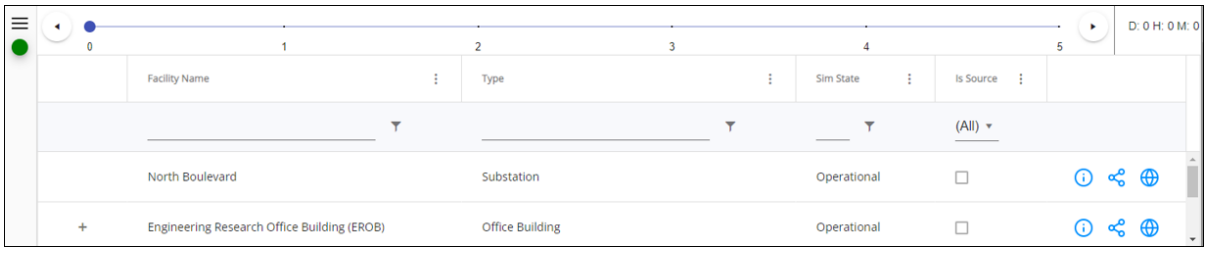

Figure 4-9 Time-dependent Simulation Timeline.

Step 3 of the simulation is to play the events. The example results of these events are shown in Figure 4-10. At simulation start (minute 0), all steady-state assets are operational, which is indicated by green and includes the three substations, automatic transfer switch, and high-performance computing servers. The contingent components are indicated by gray, which include the UPS and back-up

generator. At minute 15, the first user-defined disruption event is fired, which disables the first substation and is followed by the second substation disruption at minute 18. When commercial power is lost, the first contingent event is triggered, which results in the UPS becoming active. This is indicated by the blue color. At minute 33, the second contingent event is triggered, and the generator becomes operational. At day 1 and 33 minutes, the generator goes offline, and UPS is reactivated. Finally, the UPS loses charge, and the entire system in deenergized.

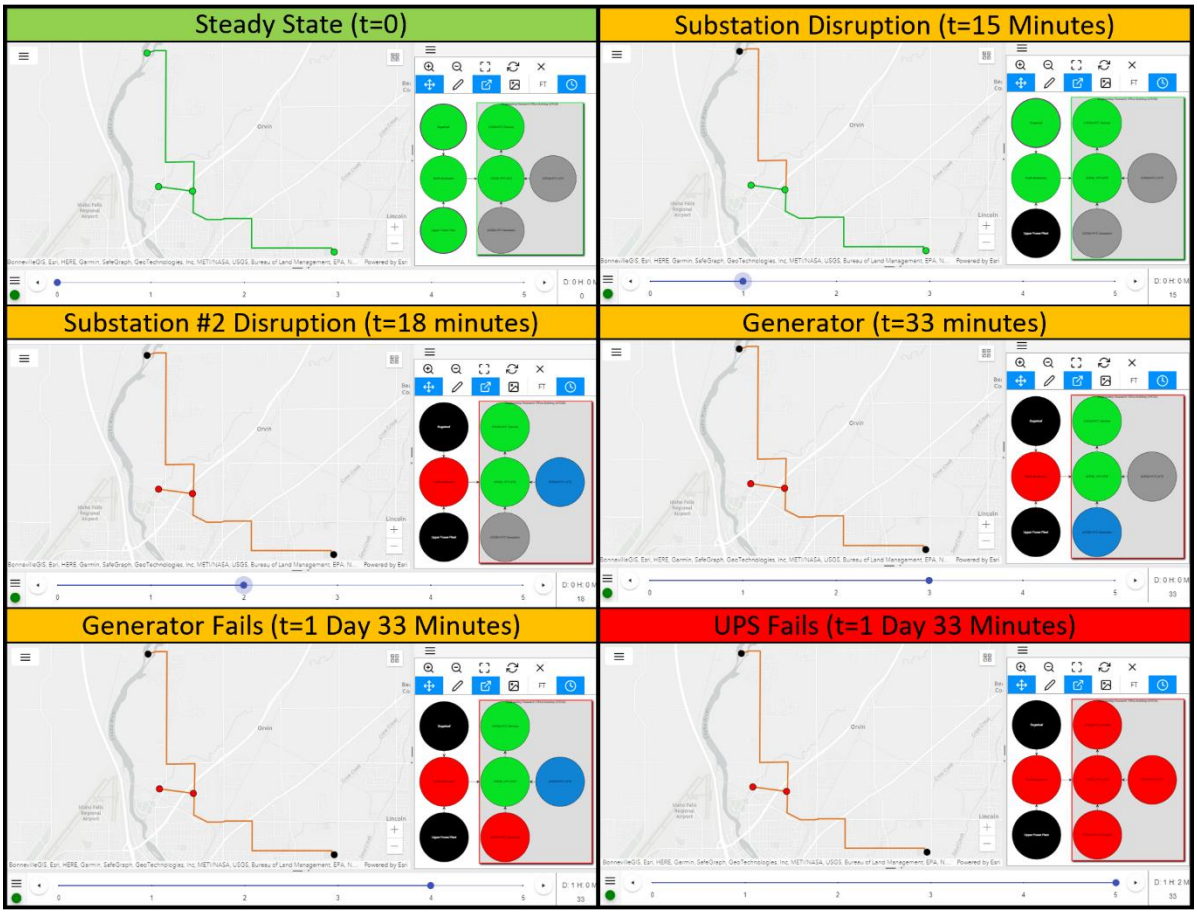

Figure 4-10 Time-dependent Cascade Simulation Results.

Metamodel

The metamodel contains the source documents and their metadata for systems, assets, and components entered into the AHA-KMS. This information includes the name, description, author, source, information access date, date of information, sector assignments, and user-defined tags. If applicable, the source documents or artifacts can be uploaded, including derivative products. The metamodel upload screen is shown in Figure 4-11.

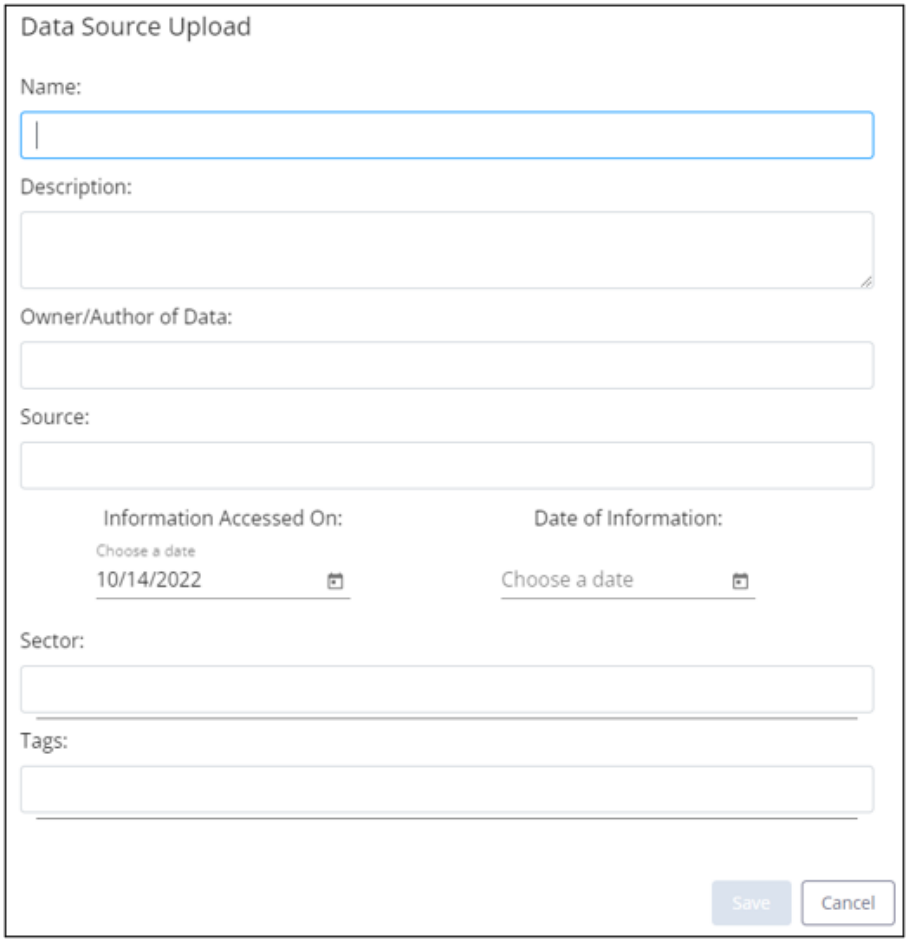

Figure 4-11 Metamodel Upload GUI.

User & Data Management Module

The user and data management module was designed to provide access to administration functions of the AHA-KMS. The primary functions of this module include: (1) user and role management, (2) access and change logging, and (3) online training material. The user and data management functions of the AHA-KMS are outside this research's scope and will not be covered.

Chapter 5: Application of the AHA Methodology to the Colonial Pipeline System

Refined fuel products are critical input commodities for most, if not all, modern industrial and community functions, such as electricity generation, transportation, manufacturing, and residential heating. To supply these commodities in sufficient quantities, pipeline systems are essential infrastructure for most aspects of the petroleum supply chain including production (gathering), transportation, and distribution. Thus, private and public infrastructure operators with dependence on petroleum products, as well as emergency management organizations, must have a sufficient understanding of general pipeline operations and how specific pipeline systems support their needs to ensure continuity of operations. In this chapter, the AHA methodology is used to develop a conceptual model of a refined fuel pipeline system, including first order dependencies, and apply it to the Colonial Pipeline (CPL) for the purpose of informing risk and resilience decision-making. The resulting model is used to simulate the 2021 Colonial Pipeline ransomware cyberattack and explore the potential impacts to airport operations.

Colonial Pipeline Ransomware Attack

On May 7th, the Colonial Pipeline Company reported they curtailed operations of their 2.5 million-barrel-per-day refined product pipeline due to a ransomware cyberattack [21]. The Colonial Pipeline is one of the primary sources of refined fuels for the East Coast of the Unites States with a capacity 3.5 times greater than its primary competitor, the Kinder Morgan Product (SE) Pipeline. So, the pipeline company's media release left fuel service providers and companies scrambling to secure alternate sources of fuel and emergency response and government organizations trying to understand and mitigate the potential impacts of the disruption. This included looking at alternative sources of transportation for petroleum products such as rail, marine, and truck systems. However, the carrying capacity of these systems severely limit their usefulness and would have required almost 3,600 rail cars or 12,500 tanker trucks to match the Colonial Pipeline volume. Although the event did not result in sustained widespread fuel shortages across the East Coast, mostly due to the existing terminal storage supplies, some areas did experience significant price inflation and shortages at retail locations. Retail shortages have been mostly attributed to panic buying by retail consumers.

This event provides an ideal empirical use case to demonstrate the utility of the AHA Framework as a scalable approach to understanding the operation of interdependent critical infrastructure systems and the potential consequence of their disruptions. The use case walks through the functional

decomposition of a refined fuel system and the creation of a functional dependency model for the Colonial Pipeline, including its primary first order dependencies.

Refined Fuel Systems Functional Asset Taxonomy Creation

The Colonial Pipeline is the largest and possibly most complex refined product pipeline system in the United States, and its continuous operation requires a diverse set of input and output commodities and services to perform it primary function of providing refined petroleum products. According to the Energy Information Agency, the primary uses of refined fuels are for transportation, heating, and power generation. Thus, the primary function of a refined petroleum product pipeline system can be expressed as: provide and transport fuels for transportation, heating, and power generation.

As described in "Chapter 3: All-Hazards Analysis (AHA) Methodology," the next step in the process is the decomposition of a typical refined fuel product pipeline system into its major component facility types. The decomposition process resulted in identifying four primary types including a refined fuel product pipeline, refined fuel product pump station, refined fuel product valve station, and refined fuel product terminal [22]. It is important to note it might be possible to decompose each of these facility types further. For example, a refined product valve station could be decomposed into a metering station, a block valve, and pipeline inspection gauge terminal stations; however, this additional layer of decomposition is not required for this use case. This step also included defining specific properties for each facility type that would be necessary to model the high-level behavior of the system, such as storage capacity for refined product terminals. In addition, to pipeline-specific facility types, it is also important to consider refined fuel production (e.g., refinery) and consumer facility types (e.g., airports).

With major asset facility types identified, the dependency link decomposition step looked to enumerate the functional requirements for each type. Table 5-1 provides the list of the commodities and services, including their general purpose, which were identified for this use case. Table 5-2 provides a description of the primary function of each facility type, including additional facility types required to demonstrate the cross-sector capability.

Table 5-1 Refined Fuel Pipeline Systems Dependencies.

Dependency Types		Function
Refined Fuels	—	Used for transportation, heating, and power generation.
	Aviation Gasoline	Used for engine fuel in light aircraft.
	Diesel	Used for engine fuel in heavy-duty trucks, trains, heavy equipment, and back-up generators.
	Fuel Oils	Used for space heating and electric power generation.
	Gasoline	Used for engine fuel in passenger cars and light trucks.
	Jet Fuel	Used for engine fuel in jet aircraft.
Electricity	—	Used to power equipment and devices.
Network Connectivity	—	Provides communication paths for enterprise systems, operational technologies, and other communication-enabled devices.

Table 5-2 Refined Fuel Pipeline System Functional Basis.

Facility Types		Function	Verb-Object Form
Refined Product Pipeline System	—	Collection of physical facilities designed to transport refined fuels between locations	**Provides** Refined Fuels **Requires** Electricity
	Refined Product Pipeline	Pipeline designed to transport refined fuels	**Provides** Refined Fuels
	Refined Product Pump Station	Facility designed to pump refined fuels through a pipeline	**Provides** Refined Fuels **Requires** Electricity
	Refined Product Storage Terminal	Facility designed to store refined fuels	**Provides** Refined Fuels **Stores** Refined Fuels

Table 5-2 Continued.

Facility Types		Function	Verb-Object Form
	Refined Product Valve Station	Facility designed to control and measure refined fuel flows within a pipeline system	**Provides** Refined Fuels
Petroleum Refinery	—	Facility designed to produce refined fuels	**Produces** Refined Fuels **Requires** Electricity **Requires** Crude Oil
Substation	—	Facility designed to distribute electric energy	**Provides** Electricity
Airport	—	Facility designed to facility air transportation services	**Requires** Refined Fuels

With a representative example of facility types required to produce, transport, and store refined fuels identified, basic dependency profiles were generated for each type. Figure 5-1 below presents the profile for a refined product pump station. In this example, refined fuels and electricity dependencies are considered critical dependencies for pump station, where the disruption of their flow would result in an inoperative condition. In general, network connectivity is considered a significant input dependency for a pump station because its disruption would most likely only result in a reduced quality of service due to potential workarounds for its intended functions. As an output dependency, network connectivity from a pump station is generally considered to have low criticality on network operation because as an endpoint, its disruption would have minimal impacts on overall network operations. It is important to note; however, as a network dependency type, functions or signals transported by the network connectivity dependency type could have varying levels of impacts to system components.

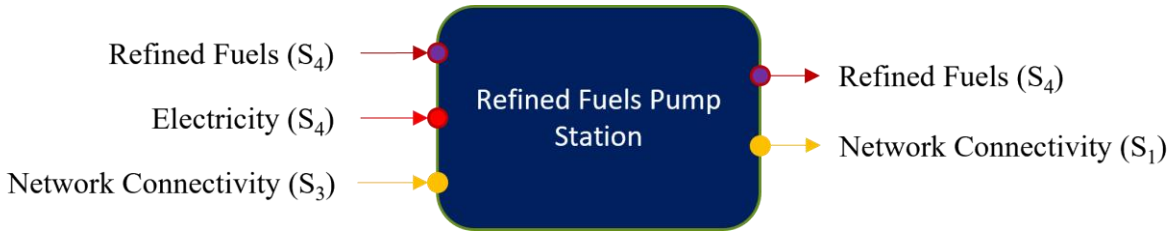

Figure 5-1 Refined Fuels Pump Station.

The final step in creating the functional basis for a refined product pipeline system is creating a conceptual model. Figure 5-2 below, provides a simplified model developed for the Colonial Pipeline use case; however by design, this model could be used for other refined fuel delivery systems. The model seeks to describe the major functions, facilities/assets (sources, storage, and capacity), flow (primary/contingent, capacity), and high-level behaviors of a general system.

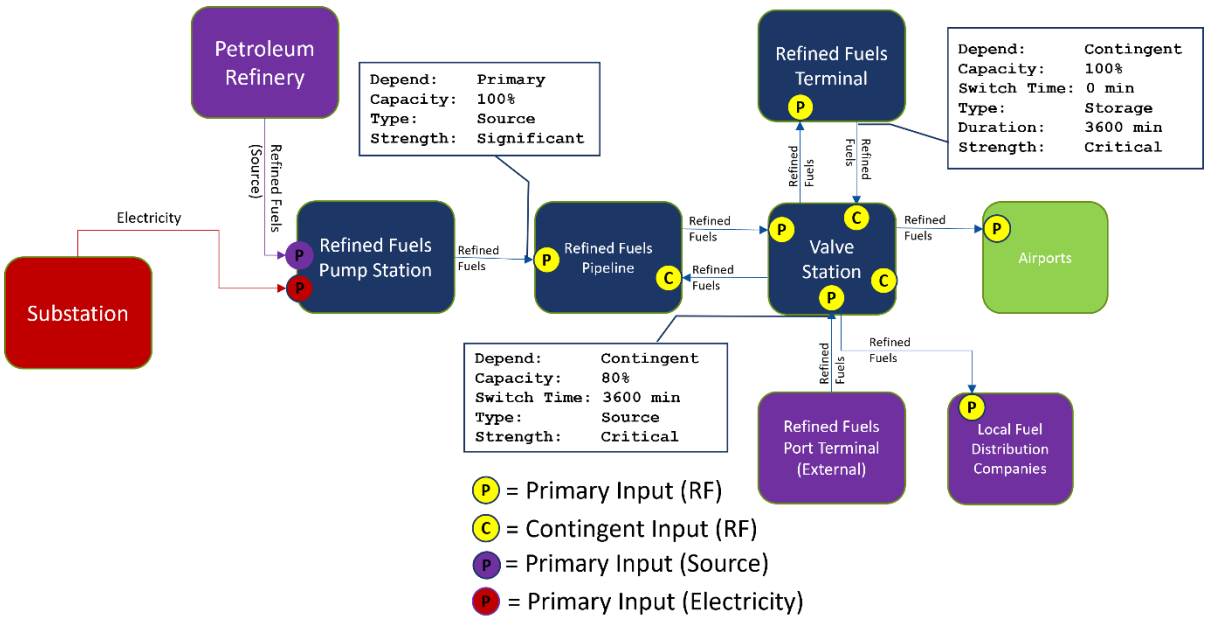

Figure 5-2 Refined Fuel System Conceptual Model.

The resulting model serves two primary purposes: one is to act as a guide for developing system-specific models, and the other is to provide a general mechanism to answers high-level questions regarding the operation of a refined product pipeline system and the potential consequences of a disruption. For example, the question "what functions does a refined fuel pipeline system enable?" can be directly answered by evaluating a functional basis of a typical refined system as shown below (Figure 5-3). This capability is valuable during crisis actions where system-specific information may not be readily available, Further, the information can be combined with a high-level system to provide greater situational awareness. The system state matrix shown in Figure 5-4 was used to validate the

typical dependency relationship and strength of the conceptual model components shown in Figure 5-2. For example, the table should be interpreted as follows: if a petroleum refinery is directly providing refined fuels to a refined product pump station, the pump station's dependency on refined fuels is considered critical and would cause the pump station to become inoperable without an alternate source of refined fuels. Finally, the conceptual model for the refined fuel product pipeline was entered into the AHA-KMS as described in "Chapter 4: AHA Knowledge Management System."

Table 5-3 Refined Product Pipeline Conceptual Model Question and Answer.

QUESTION: What functions does a refined fuel system enable?		
Facility Type	Facility Function	Dependency-Type Function
Refined Fuel System	Provide Refined Fuels	Used for transportation, heating, and power generation
ANSWER: Refined fuel systems provide refined fuels for transportation, heating, and power generation.		

Table 5-4 Refined Product Pipeline System State Matrix.

General Strength of Dependency							
Producer/Transport/Storage Node	Dependency	Consumer Node					
		Petroleum Refinery	Refined Fuel Pump Station	Refined Fuel Pipeline	Refined Fuel Terminal	Substation	Airport
Petroleum Refinery (P)	Refined Fuels	NA	4	4	3	NA	NA
Refined Fuel Pump Station (T)	Refined Fuels	NA	4	4	3	NA	3
Refined Fuel Pipeline (T)	Refined Fuels	NA	4	4	3	NA	3
Refined Fuel Valve Station (T)	Refined Fuels	NA	4	4	3	NA	3
Refined Fuel Terminal (S)	Refined Fuels	NA	4	4	NA	NA	4
Substation (T)	Electricity	4	4	NA	4	4	4

Colonial Pipeline Systems Functional Model Creation

The second phase of the AHA methodology is developing a system-specific model from the general conceptual model; in this case, the focus was on developing a function-flow model of the Colonial Pipeline system which is shown in Figure 5-3. The model incorporated known Colonial Pipeline facilities, petroleum refineries (i.e., refined fuel product production), petroleum storage terminals (i.e., refined fuel storage facilities), substations (i.e., electrical power for facilities), and airports (i.e., consumers of refined fuels). It is important to note the model's facilities, dependency links, and parameters were identified from multiple sources, including GIS data layers, pipeline reports, news articles, and estimates from subject-matter experts; however, they have not been validated with the Colonial Pipeline Company.

Figure 5-3 Colonial Pipeline Dependency Model.

Overall, the dependency model consists of 807 nodes with 346 nodes representing Colonial Pipeline facilities. The remaining 461 nodes represent other connected pipeline assets, refineries, product terminals, and airports. A count of the facilities by their degrees of separations from the Colonial Pipeline system is provided in Table 5-5.

Table 5-5 Colonial Pipeline Dependency Model Facility Count.

Colonial Pipeline Dependency Model Facility Count							
Facility Type	**Upstream**				**Downstream**		
	3rd	2nd	1st	**CPL**	1st	2nd	3rd
Petroleum Refinery	0	7	0	**0**	0	0	0
Refined Product Pipeline	2	1	3	**174**	0	10	3
Refined Product Valve Station	0	0	1	**75**	2	1	3
Petroleum Pump Station	6	2	0	**81**	5	0	7
Petroleum Product Storage (Terminal)	2	2	13	**16**	157	6	0
Substation	6	7	70	**0**	0	0	0
Airport	0	0	0	**0**	1	7	0
Total Facility Count	**16**	**19**	**87**	**346**	**165**	**24**	**13**

Initial validation of the system's dependency model was conducted utilizing the simple cascade simulation to ensure the modeled flows were sufficiently accurate to support range-of-influence analysis for risk and resilience assessments. The approach used was a supervised iterative n-1 removal of system nodes. Figure 5-4 illustrates the results of a refined product injection site and a main line disruption, where downstream pipeline facilities are only moderately impacted by the injection site disruption (shown in yellow) as opposed to the main line disruption resulting in both critical (shown in red) and significant impacts (shown in orange) to facility operations. The transition from critical impacts to significant impacts is due to the presence of a bulk storage terminal buffering the loss of the main inputs.

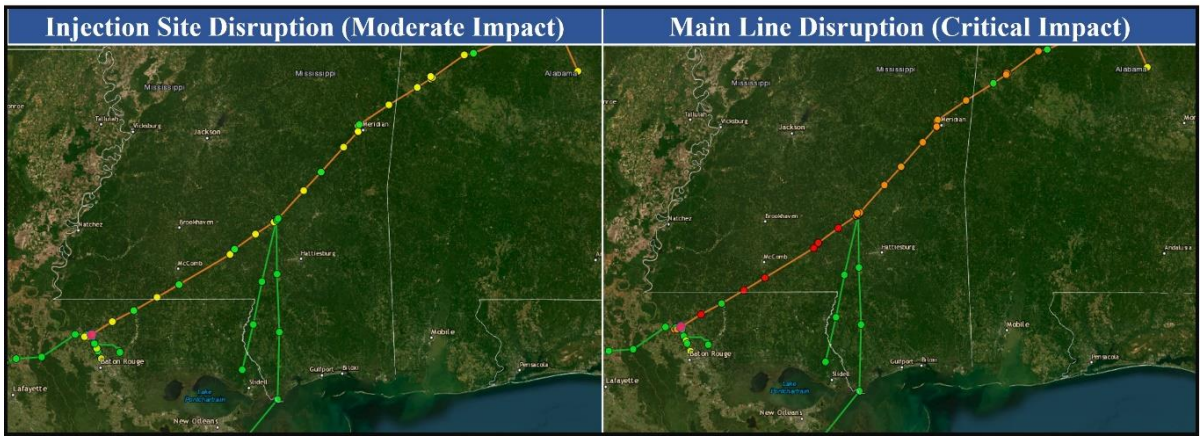

Figure 5-4 Colonial Pipeline Cascade Simulation Validation Example.

After initial validation of the flows and FFL, the model's facility nodes were enriched with contingency, system storage, and system recovery information, based on their respective dependency

profile. For the Colonial Pipeline, model storage duration was assigned to each refined product storage terminal based on subject-matter expert input on the average time a typical terminal could supply a product before it would be depleted. Considerations included location (e.g., rural vs. urban) and terminal capacity if known, values ranged from 1 to 7 days. In addition, open-source information was leveraged to assign storage durations for many terminals supporting airport operations. This resulted in a cross-sector time-sequenced functional-flow model (shown in Figure 5-5) intended to support system behavior simulations to assess potential impacts of facility-level disruptions. The resulting model of the Colonial Pipeline was validated against the 2021 ransomware attack events.

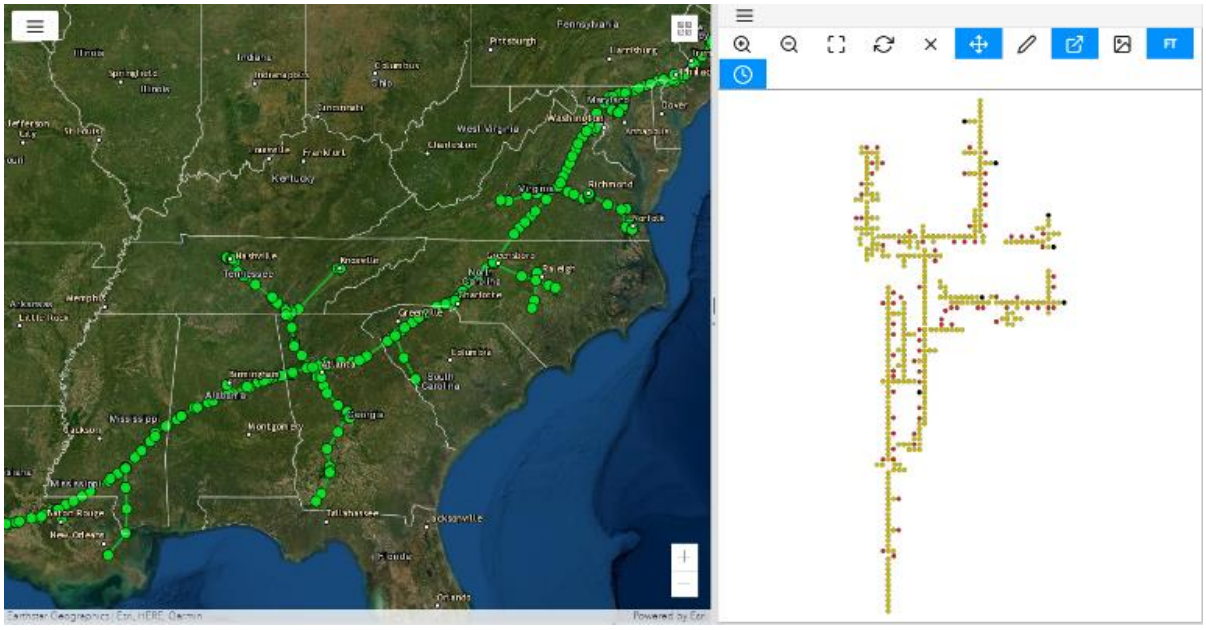

Figure 5-5 Colonial Pipeline Generalized Discrete Event Model.

2021 Colonial Pipeline DarkSide Ransomware Attack Scenario

On April 29th, 2021, the DarkSide Cybercrime Group's ransomware-as-a-service was thought to have been used to gain access to and compromise the business systems of the Colonial Pipeline Company. As a result of the compromise, the company was forced to preemptively shut down pipeline operations on May 7th to investigate the integrity of the systems operational technology (OT) control environment [126]. As a consequence, fuel transportation from Gulf Coast refineries to terminals along the entirety of the 5,500-mile system was curtailed, effectively sequestering products already in route and at the system's multiple breakout terminals. Table 5-6 provides a timeline of events as reported by the Colonial Pipeline Company.

The Colonial Pipeline dependency model was simulated under four scenarios. Scenarios NR1 and NR2 simulated system behavior without restoration activities, where NR1 consisted of only Colonial

Pipeline assets and their known downstream assets (up to three orders), and NR2 included Kinder Morgan SE product pipeline assets and additional upstream assets. Scenarios R1 and R2 simulated a set of plausible restorations activities taken to recover the pipeline system after the curtailment, again with and without the SE product pipeline assets.

Table 5-6 Colonial Pipeline Ransomware Attack Recovery Timeline.

Colonial Pipeline Ransomware Attack Recovery Timeline	
Date	**Status**
Friday, May 7, Time Unknown	System Operation Curtailed
Saturday, May 8	System Operation Curtailed
Sunday, May 9	Mainlines (Lines 1, 2, 3 and 4) remain offline.
	Lateral restoration underway, with some smaller lateral lines between terminals and delivery points are now operational.
Monday, May 10, 12:25 p.m.	Mainlines (Lines 1, 2, 3 and 4) remain offline.
	Lateral restoration underway, with some smaller lateral lines between terminals and delivery points are now operational.
	Federal Government issues a temporary hours of service exemption for motor carriers and drivers transporting refined products across Colonial's footprint
Monday, May 10, 7:50 p.m.	Line 4, which runs from Greensboro, NC, to Woodbine, MD is operating under manual control for a limited period of time while existing inventory is available.
Tuesday, May 11, 5:15 p.m.	Colonial has delivered approximately 967,000 barrels (~41 million gallons) to various delivery points along our system. This includes delivery into the following markets: Atlanta, GA, Belton and Spartanburg, SC, Charlotte and Greensboro, NC, Baltimore, MD, and Woodbury and Linden NJ
Wednesday, May 12, 5:10 p.m.	Colonial Pipeline initiated the restart of pipeline operations today at approximately 5 p.m. ET.
Thursday, May 13, 9 a.m.	By mid-day today, we project that each market we service will be receiving product from our system. The green segments on this map are operational, meaning product delivery has commenced. Blue lines will be operational later today (See Figure).
Thursday, May 13, 4:40 p.m.	Colonial Pipeline has continued to make substantial progress in safely restarting our pipeline system. We can now report that we have restarted our entire pipeline system and that product delivery has commenced to all markets we serve.
Monday, May 17, 5:25 p.m.	We can now report that we are transporting refined products (gasoline, diesel and jet fuel) at normal levels and are fully operational.

To replicate the initial shutdown for all scenarios, the 346 Colonial Pipeline assets were disable at simulation time of hour 1, which was correlated to 8 a.m. eastern standard time. For the restoration

scenarios, disruption and restoration events were generated through analysis of the Colonial Pipeline Company's media releases outlined in Table 5-7 and engineering judgment. Restoration events as described above were offset based on the differential between real and simulation time. In many cases, details about the timing of specific lateral restoration were not reports, so subject-matter expert input was used in conjunction with Colonial Pipeline Company's media statements to generate a plausible restoration scenario. The following general rules were considered when determining lateral restoration:

1. Laterals needed to be connected directly to bulk storage facilities
2. Markets not serviced by the Kinder Morgan pipeline were prioritized
3. Larger urban markets were prioritized.

The primary scenario event lists (MSEL) for both scenarios are shown in Table 5-7 and Table 5-8.

Table 5-7 No Restoration Primary Scenario Event List (NR1 & NR2).

Date	Time	Event	Simulation Time	Notes
May 7th	7:00	Simulation Start	0:0:00:00	
May 7th	8:00	Curtailed Operations	0:1:00:00	

Table 5-8 Restoration Primary Scenario Event List (R1 & R2).

Date	Time	Event	Simulation Time	Notes
May 7th	8:00	Curtailed Operations	0:1:00:00	
May 9th	8:00	Belton Restoration	2:1:00:00	
May 9th	11:00	Linden Restoration	2:4:00:00	
May 9th	14:00	Woodbury Restoration	2:7:00:00	
May 9th	17:00	Spartanburg Restoration	2:10:00:00	
May 9th	17:10	Media Release	2:10:10:00	Small Laterals w/Terminals

Table 5-8 Continued.

Date	Time	Event	Simulation Time	Notes
May 9th	20:00	Charlotte Restoration	2:13:00:00	
May 9th	23:00	Atlanta Market Nashville	2:16:00:00	
May 10th	2:00	Atlanta Market Bainbridge	2:19:00:00	
May 10th	5:00	Atlanta Market Knoxville	2:22:00:00	
May 10th	8:00	Athens Market Restoration	3:1:00:00	
May 10th	11:00	Greensboro Market Restorations	3:4:00:00	
May 10th	14:00	Mitchell Market Restoration	3:7:00:00	
May 10th	19:00	Greensboro-Woodbury Manual Operation	3:12:00:00	
May 10th	19:50	Media Release	3:12:50:00	Manual Restart
May 10th	22:00	Port Arthur Delivery Restoration	3:15:00:00	
May 11th	1:00	Port Neches Delivery Restoration	3:18:00:00	
May 11th	5:00	Beaumont Delivery Restoration	3:22:00:00	
May 11th	8:00	TEPPCO Restoration	4:1:00:00	
May 11th	12:00	Collins Restoration	4:5:00:00	
May 11th	15:00	Pasadena Injection Restoration	4:8:00:00	
May 11th	17:15	Media Release	4:10:15:00	Delivery from Refineries
May 12th	16:00	Collins To Epes Mainline Restoration	5:9:00:00	
May 12th	17:10	Media Release	5:10:10:00	Restart Initiated
May 12th	19:00	Houston to Hebert	5:11:00:00	

Table 5-8 Continued.

Date	Time	Event	Simulation Time	Notes
May 12th	20:00	Epes To Pelham	5:12:00:00	
May 12th	22:00	Hebert to Lake Charles	5:14:00:00	
May 12th	23:00	Pelham to Atlanta	5:15:00:00	
May 12th	0:00	Lake Charles to Baton Rouge	5:17:00:00	
May 13th	1:00	Baton Rouge to Collins	5:18:00:00	
May 13th	3:00	Woodbury to Liden	5:20:00:00	
May 13th	4:00	Atlanta to Belton	5:21:00:00	
May 13th	5:00	Belton to Charlotte	5:22:00:00	
May 13th	6:00	Charlotte to Greensboro	5:23:00:00	
May 13th	9:00	Media Release	6:2:00:00	Restart Progress
May 13th	9:00	Raleigh Lateral Restoration	6:2:00:00	
May 13th	11:00	Pelham Market Restoration	6:4:00:00	
May 13th	13:00	Baltimore Lateral Restoration	6:6:00:00	
May 13th	16:40	Media Release	6:9:40:00	System Restart Complete

Scenario Results

The summary results for the four use cases are presented below with a focus on airports and refined product storage terminal status. These facility types represent the downstream interconnection points between the refined fuel pipeline systems and the consumers of refined fuels.

No Restoration Baseline Scenario Results

In the baseline scenarios NR1 & NR2, there is a single initiating event, which is the preemptive shutdown of all the Colonial Pipeline assets at *t=D0:H1*. From this event, seven or eight subsequent

events were autogenerated based on the time-to-switch and storage durations parameters set for each asset participating in the respective simulation scenario.

NR1 - No Restoration without SE Products Pipeline

The NR1 scenario consists of 633 total assets, which include 346 Colonial Pipeline facilities. The remaining 287 assets represent both upstream and downstream facilities. For the NR1 scenario, the system-generated state changes occur at: D3:H1 (2), D4:H1 (3), D5:H1 (4), D6:H1 (5), D7:H1 (6), and D11:H1 (7); which are driven entirely by the storage duration parameter of the refined fuel terminals. Figure 5-6a presents the count of airport by state for each timestep, and Figure 5-6b presents the count of refined product terminals by state for each timestep based on the failure-function-logic simulation.

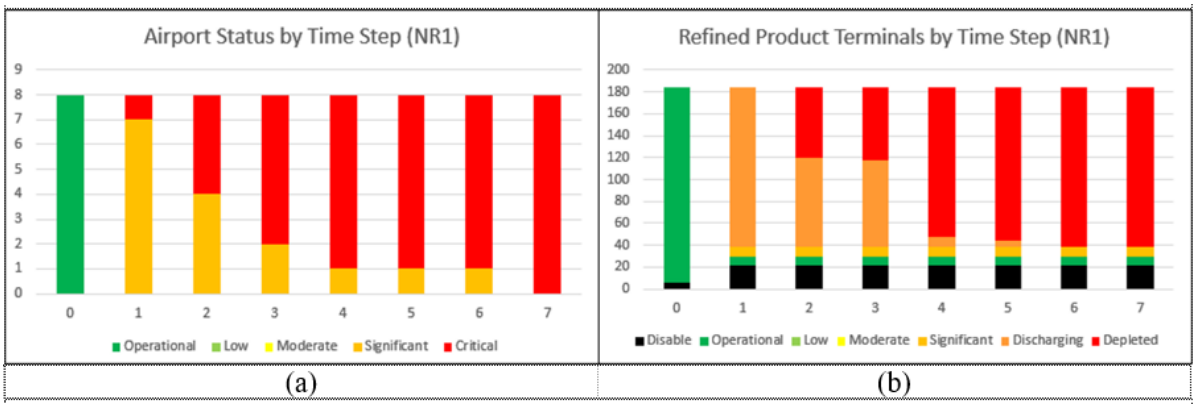

Figure 5-6 NR1 Airport (a) and Terminal (b) States by Timestep.

As expected, the storage capacity of the connected fuel terminals buffered initial impacts of the pipeline curtailment; however, leveraging the criticality measures, airports and terminal operators are understood to have transitioned into a state of operations where they would be unable to perform their normal functions and services. As defined, this would result in major feature/product failure (e.g., spot markets drying up), inconvenient workarounds (e.g., utilizing rail/trunk systems), and limited or impaired consumer services. Without major behavioral changes, the simulation suggests that the region would begin to experience significant system wide shortages in 3–4 days.

NR2 - No Restoration with SE Products Pipeline

The NR2 scenario consists of 842 assets. The NR2 scenario incorporates the Kinder Morgan Southeast product pipeline into the simulation model. The SE product pipeline supplies refined products to many of the same airports, product terminals, and market area that the Colonial Pipeline services. The system-generated state changes occur at D1:H1 (1), D2:H1 (2), D3:H1 (3), D4:H1 (4), D5:H1 (5), D6:H1 (6), D7:H1 (7), and D11:H1 (8). Figure 5-7a presents the count of airports by state

for each timestep, and Figure 5-7b presents the count of refined product terminals by state for each timestep based on the failure-function-logic simulation.

\

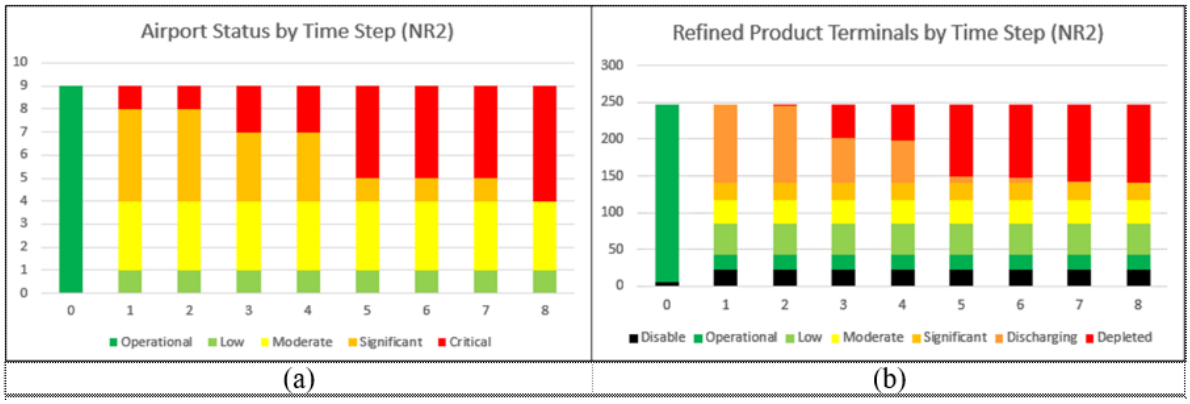

Figure 5-7 NR2 Airport (a) and Terminal (b) States by Timestep.

The scenario provides insight into the alternative supplies and identifies airports and markets that would have at least a portion of their steady-state supplies. In this case, the Kinder Morgan pipeline could help at least four major international airports remain at least partially operational and keep many of the major market areas completely drying up.

Restoration Scenario Results

In the restoration scenarios R1 & R2, there is a single initiating event, which is the preemptive shutdown of all the Colonial Pipeline assets at *t=D0:H1*; however, 33 restoration events were also added to the scenario's MSEL as shown in Table 5-8. As a result of these 34 events, an additional eight and ten autogenerated events were added to the MSEL based on the time-to-switch and storage durations parameters set for each asset participating in the respective simulation scenarios.

R1 - Restoration without SE Products Pipeline

For the R1 scenario, the estimated restoration activities greatly reduced the number of airports and terminals that were critically impacted by the event. In this case only, the Piedmont Triad International and Thurgood Marshall Baltimore-Washington International Airports were predicted to have extended fuel shortages with Washington-Dulles International Airport having a potentially brief disruption seen at timestep 10 as shown in Figure 5-8. At timestep 35 (D6:H4), the final airport is returned to operational status.

For the refined product terminals, 146 of terminals transition to a discharging state immediately following the pipeline curtailment. Over the next eight timesteps, as segments of the Colonial

Pipeline were restored, many terminals were able to receive fuels stored in the pipeline breakout tank farms. Then, at t=10 (D3:H1), terminals that remained disconnected with estimated storage capacities of 3 days transitioned to a critical or depleted state. In this case, a maximum of 53 terminals went dry, which occurred at T=20 (D5:H1).

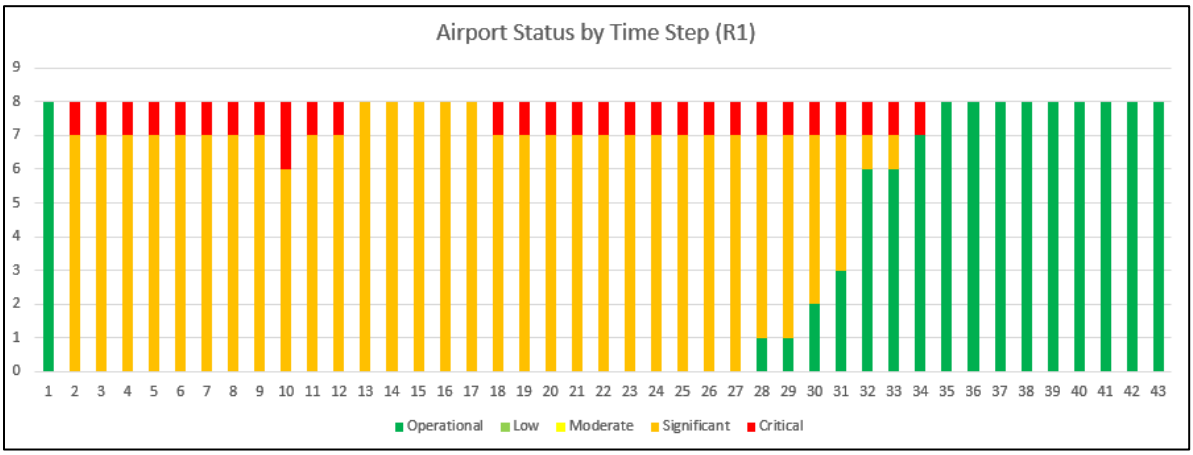

Figure 5-8 R1 Airport Status by Timestep.

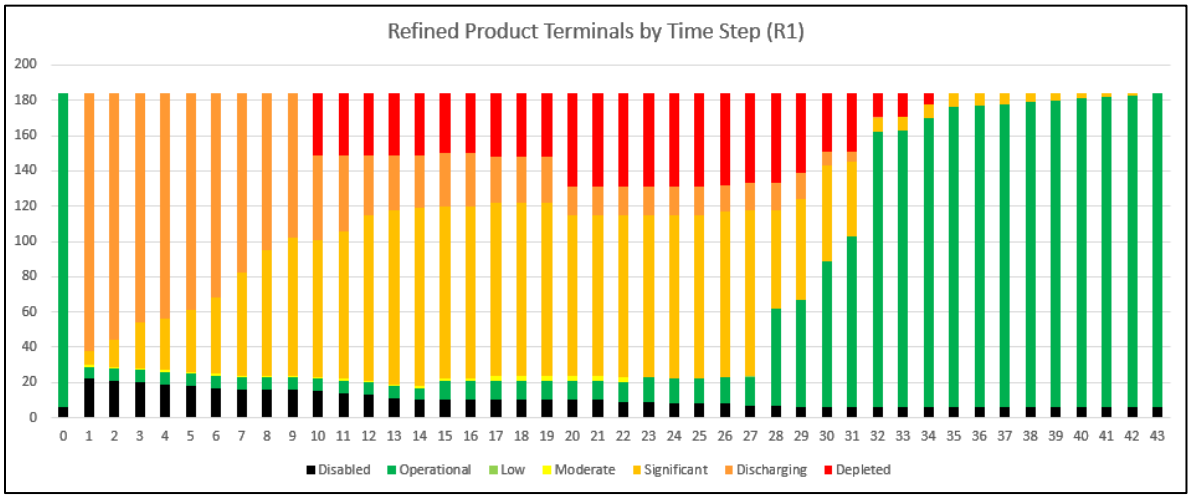

Figure 5-9 R1 Refined Product Terminals by Timestep.

R2 - Restoration with SE Products Pipeline

For the R2 Scenario, the Kinder Morgan SE product pipeline is incorporated allowing, as expected, many airports and terminals to receive fuels throughout the simulation reducing the number of airports and terminals that were critically impacted by the event. In this case only, the Piedmont Triad International and Thurgood Marshall Baltimore-Washington International Airports were predicted to be impact by fuel disruptions, while the Washington-Dulles International Airport fuel disruption is mitigated by its connection to the Kinder Morgan pipeline as shown in Figure 5-10. At timestep 35

(D6:H4), the final airport is returned to operational status. For the refined product terminals, the number of terminals that immediately transition to a discharging state was reduced from 146 to 106, which represents approximately a 28 percent reduction. Again, over the next eight timesteps, segments of the Colonial Pipeline were restored, which enabled many terminals to receive fuels from the pipeline's tank farms. In this case, a maximum of 33 terminals were estimated to go dry, which represents approximately a 37 percent reduction in critically impacted terminals (Figure 5-11).

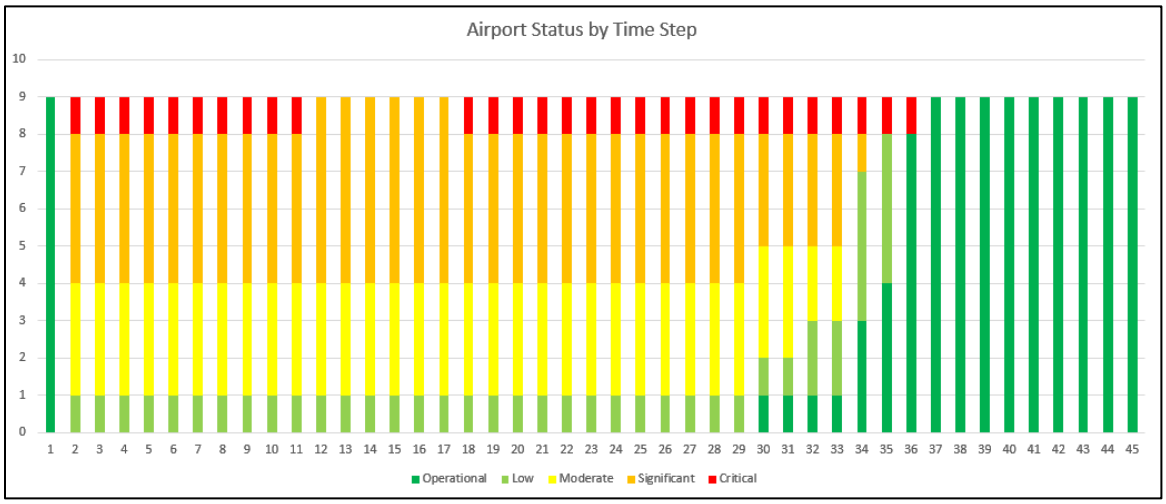

Figure 5-10 R2 Airport State by Timestep.

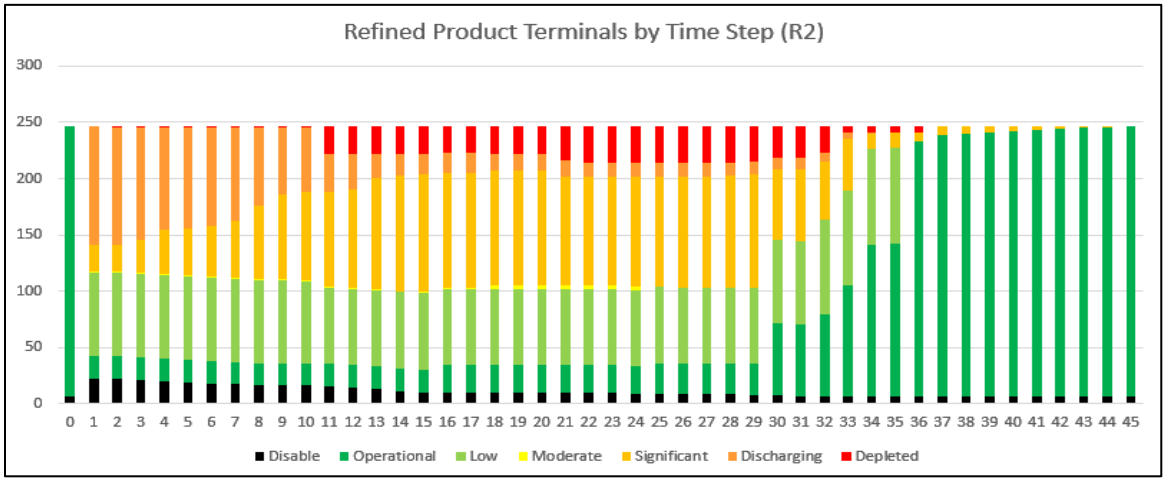

Figure 5-11 R2 Refined Product Terminal State by Timestep.

Discussion

The purpose of this study was to (1) assess the ability of functional-flow network models to simulate the behavior of interconnected infrastructure systems and (2) demonstrate the AHA functional-flow network modeling framework can support risk and resilience assessments. As the Colonial Pipeline

scenarios illustrates, knowledge graphs based on a functional basis for engineered systems provide a robust and scalable approach to simulate system behavior through the application of FFL which makes them ideal for conducting both risk and resilience assessments when high-fidelity operational and engineering information is not available.

The four scenario cases demonstrate how the AHA methodology can be used to assess and evaluate the potential consequences of infrastructure disruptions and the supply chains they support. The methodology provides an effective means to account for functional, non-functional, spatial, and temporal characteristics of interdependent critical infrastructure systems. In addition, incorporating FFL provides emergency management, infrastructure owners, and other organizations with a better understanding of their exposure to potential disruptions so they can plan more effective mitigations.

Further, reviewing the core infrastructure system risk and resilience questions presented in "Chapter 2: Background and Literature Review," the AHA methodology provides a scalable approach to address crisis action, mitigation, and strategic risk concerns, such as identifying systematic important critical infrastructure. The AHA methodology provides an effective approach to answer the following questions:

- If an infrastructure failure occurs what are the national, regional, and local impacts?
  - What is the significance of failure?
- How long until impacts of an infrastructure failure are realized?
  - What mitigations are in place to buffer the event?
- Is there a potential for cascading or escalating failures?

However, there are several limitations of the proposed AHA methodology. First, the ability to account for dynamic behavior exhibited by fuel consumers is currently not captured by the proposed approach. For example, many airlines modified their operational practices to reduce the demand for fuel at affected airports by implementing a practice known as tankering (i.e., carrying extra fuel on an inbound aircraft). In addition, many motorists began to panic buy, which strained the last-mile distribution networks causing localized disruption at gas stations. In the first case, the airlines extended the operational status of the impacted by reducing demand; while in the second case, motorists exacerbated the shortages by outpacing fuel truck deliveries for regional terminals. Adding stochastic-based approaches could help to address this shortcoming. Second, collecting the required information to effectively assess cross-sector dependencies and the potential consequence of their disruption at a national level is still a significant challenge. In the scenario cases presented in this study, actual consumers of refined products from the terminals are not known and is often difficult to

determine from open-source information. Finally, incorporating additional specific properties such as terminal capacity would also provide a refined estimate of impact.

Overall, this study achieved its goal of demonstrating that functional-basis-informed graphs are ideal for describing and analyzing interconnected infrastructure system behavior under all-hazard conditions. Functional-basis-informed graphs provide an optimal structure for modeling function, commodity, and service flows of interconnected systems and facilitate scalable and repeatable assessments of system behaviors suitable for vulnerability, consequence, and risk analysis.

Conclusion

In this chapter, the AHA methodology was applied to the refined fuel pipeline systems and validated against the Colonial Pipeline ransomware attack. Like many other hazard events, the Colonial Pipeline attack highlighted significant regional dependencies on a single critical infrastructure system, which, when disrupted, had far reaching impacts including cascading and escalating failures of other critical infrastructure systems. Understanding critical dependencies and the consequence of disruption is essential for effective policymaking, continuity of operation planning, community resilience planning, and emergency management and response. For example, policy makers need to fully understand the refined fuel systems and the markets they support to appropriately incentivize investments in resilience enhancement by the system owners and the communities they support. Similarly, community resilience planners need sufficient understanding of the systems supporting their region to plan and prepare for potential disruption.

This will require developing effective decision support methods and tools that can be used by both decision makers and analysts across industry and government. Successful approaches will accept varying levels of data fidelity, reducing the burden of information sharing on critical infrastructure owner and operators, and provide the ability to evaluate different courses of action related to both policy and infrastructure investment.

Finally, additional efforts will need to be made to address the community resilience planners' need for enhanced information sharing with the legal and information security professionals' desire to limit or totally restrict sharing.

Chapter 6: Preliminary Cyber-Physical Functional-Flow Model Analysis

As discussed throughout this dissertation, infrastructure is ubiquitous in modern societies, and their reliable and resilient operation is of paramount importance to national security and economic vitality [2]. Recent trends have been to optimize the operation of these systems by integrating information and communication technologies, resulting in a tight coupling of cyber and physical components. These cyber-physical systems (CPSs) consist of computational and communication systems embedded within physical systems to monitor, coordinate, and control the continuous dynamics of the physical system [127, 128]. CPSs are commonly referred to as ICS, supervisory control and data acquisition (SCADA) systems, and distributed control systems. CPSs promise to provide increased capacity, reliability, and efficiency over physical systems alone. One example is the proposed smart grid. However, the integration of cyber technology has the potential to expose these systems to interruptions in the underlying information and communication components due to equipment failure or malicious intent and may lower their resiliency if not properly designed and secured [129-131].

Over the last couple of decades numerous cyberattacks have been reported that targeted CPS, including Natanz (Stuxnet), Ukraine (BlackEnergy 3), and the Oldsmar Water Plant compromises. In each of these cases, the attackers were able to deny, disrupt, or destroy the intended operation of the impacted CPS, demonstrating their vulnerability to malicious attacks [132-134]. Further, cases like the Ukraine power grid attack highlight the potential for widespread consequences of a successful attack. In order to reduce the potential for high-consequences events, engineers and operators need effective methods to identify and mitigate risks of successful attacks in the design and operation of CPS. In response, the U.S. Department of Energy developed the National Cyber-Informed Engineering (CIE) Strategy to help increase security, reliability, and resilience in American's energy sector through awareness, education, design, and assessment of SICI [135].

To achieve many of the goals outlined by the DOE's CIE Strategy, new methods, techniques, and tools are needed to design future systems, identify existing vulnerabilities, and train the next generation of CPS engineers, operators, and emergency responders. These stakeholders require highly scalable and customizable modeling/simulation and training environments that mimic realistic CPS behavior, including their spatial and temporal dynamics. Ideally, the evaluation environment would be a replica of the actual systems under evaluation; however, it would be unrealistic to develop physical copies of all interconnected systems for design, assessment, and training purposes. Software-based simulation provides an alternative to physical systems; however, these modeling and simulation

approaches are also problematic because CPSs are composed of the process equipment, control devices, software, and information networks that operate in continuous and discrete time [127, 128, 134]. Thus, Lee contends that modeling CPS behavior requires understanding both continuous processes, as well as discrete events which require utilizing and linking multiple categories of model types which present implementation challenges. Rai and Shu categorize the different modeling and simulation types as (1) physics based, (2) state machine based, (3) rule or agent based, and (4) data driven [136]. However, these options do not consider the potential for hardware-in-the-loop (HIL) approaches that would enable scalability and more closely mimic the operational process environment.

HIL-based approaches have been shown to be effective in conducting both security and risk research [137-139]. Potteiger et al. evaluated a HIL-based approach to evaluate the effects of human-in-the-middle attacks on railway network behavior by demonstrating its ability to capture the attack propagation as well as the systems cyber and physical behavior [139]. Similarly, Liu et al. leveraged a HIL co-simulation environment to model a cyberattack on a power-system CPS to evaluate the impact of communication disruptions on system response [138]. HIL simulation capabilities also provide ideal training environments for system operators due to the enhanced ability to account for physical system behavior without having to build a replica of an entire system.

In this chapter, a HIL-based environment that extends the DHS Control Environment Laboratory Resource (CELR) environment by integrating the AHA simulation capability is proposed. CELR is an HIL environment that was developed by DHS to provide ICS stakeholders with a resource to perform security research and training related to cyberattack scenarios. CELR incorporates physical control system equipment with connected information technology (IT) to emulate actual sector-specific processes using OT found throughout U.S. infrastructure. Such a capability has the potential to create a highly customizable and immersive environment that could be tailored for CPS design, mitigation, and training needs. Incorporating the AHA capability provides the ability to simulate cascading impacts that could affect CPS operations or be affected by disruptions in a CPS. This initial effort focuses on the evaluation of the AHA time-dependent FFL for modeling CPS behavior for the purpose of risk identification and training. This research further establishes that functional-basis-informed graphs provide an optimal structure for modeling function, commodity, and service flows of interconnected systems and facilitate scalable and repeatable assessments of system behaviors suitable for vulnerability, consequence, and risk analysis.

CELR Testbed and Proposed HIL Architecture

The proposed architecture seeks to integrate the AHA simulation capability within the CELR environment to enhance consequence analysis and training scenario generation for cyber-based disruptions.

Control Environment Laboratory Resource Overview

The CELR is an ICS testbed environment developed by CISA for the purpose of conducting ICS and SCADA systems security research. The environment is also intended to aid in training government and industry stakeholders by allowing them to experience simulated kinetic effects of successful cyberattacks on control system environments. Currently, the CELR has testbed resources for oil and natural gas compressor stations, electrical substations, building automation, and chemical manufacturing, as well as the supporting network system devices.

AHA Simulation Environment

The AHA simulation environment provides a potential mechanism to enhance consequence analysis and training beyond the CELR physical hardware environment. AHA leverages a robust knowledge graph for the development of dependency models for critical infrastructure and supply chain analysis. Integrating the AHA capability could provide enhanced ability to (1) develop testing and training scenarios, (2) visualize results, and (3) assess consequences beyond the CELR testbed environment. These capabilities would allow for both cyber-informed engineering prototyping and advanced training experiments designed to evaluate the effects of cyberattacks on a CPS. By leveraging the AHA knowledge graph approach, scenarios can by designed to test and compare various CPS reference architecture to identify potential vulnerabilities and assess their risk and resilience.

Case Study

The objective of this case study is to test the ability of the AHA knowledge graph to model CPSs and if the proposed FFL is suitable for CPS behavior simulation research. The case study is based on a notional natural gas pipeline system which leverages the CELR natural gas compressor station platform and the Integrated Enterprise SCADA System Architectures for Safe and Efficient Pipeline Operations proposed by Schneider Electric and Cisco Systems [142].

CELR Oil & Natural Gas Pipeline Platform Overview

The CELR Oil & Natural Gas (ONG) Pipeline Platform provides a physical representation of a commonly configured pipeline compressor station and associated ICS and communications network, including the human-machine interface, which can be used for demonstrations and simulations of cybersecurity attacks to cause visible physical effects. A process flow diagram is shown in Figure 6-1.

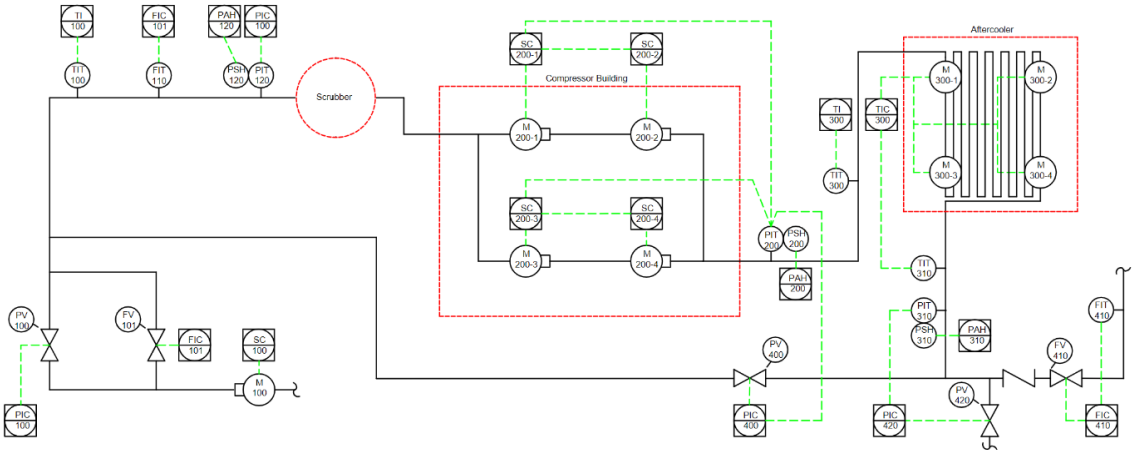

Figure 6-1 CELR ONG Platform Pipeline and Instrumentation Diagram.

Natural gas enters the compressor station through the M-100 compressor shown on the bottom left of the diagram. After the gas enters the station, it passes to pressure valve (PV-100) and flow valve (FV-100) which can be adjusted on the human-machine interface (HMI) to influence the pressure or flow rates. The valves supply the main line that flows into the simulated scrubber. Temperature (TI-100), flow (FIC-101), and pressure (PIC-100) transmitters relay the respective instrumented rates at different portions of the process to ensure they are within identified thresholds for safe operations.

From the scrubber, gas flows into the compressors (M-200-1, 2, 3, and 4) which increase the pressure to push the natural gas through the pipeline to the next hub or compressor station. From the compressor, the line runs through a pressure valve (PV-400) followed by additional instrumented temperature (TI-300), flow (FIT-410), and pressure (PIT-310) transmitters for the second portion of the process. Cooling fans (M-300-1, 2, 3, and 4) are also represented to ensure the temperature of the gas is maintained at optimal temperatures to maximize efficiencies in transport. Gas is then pumped through the flow valve (FV-410) for distribution to the customer or through the pressure valve (PV-420) to the flare stack to burn off excess gas or impurities in the system.

The platform control component hardware contains a PLC, HMI, and supporting field devices. The PLC is an Emerson Bristol Babcock ControlWave process automation controller (PAC). The PAC is housed in a 10-slot chassis and interfaces with the field devices via the input/output (I/O) modules. The HMI is a Maple Systems 15” or an OASyS HMI. Communication between the PAC, HMI, and OT network devices is accomplished through an integrated 10/100M Ethernet Interface in the CPU Module. Field devices are monitored and/or controlled by the PAC I/O modules via pre-wired cables,

except for PV-420, which is connected via a Prosoft Technologies 900 MHz wireless radio kit to the PAC. This provides valve position control and feedback via 900 MHz radio communications.

Electric Connected Pipeline Reference Architecture

The Electric Connected Pipeline Reference Architecture developed by Cisco and Schneider Electric was designed to enable pipeline operations and management while maintaining system integrity, safety, security, and reliability. The architecture provides a comprehensive design guide for communication network required for operations and management of a pipeline system. In such, it provides an ideal structure for modeling a notional system that includes its physical assets as well as its cyber components. The reference architecture considers control centers, compressor stations, pump station, metering stations, PIG stations, terminal stations, and block valve stations.

AHA Notional Pipeline System

For this research, a notional natural gas pipeline systems model was developed leveraging AHA methodology described in "Chapter 3: All-Hazards Analysis (AHA) Methodology." The notional system model was based on the natural gas system, communication component, and control system component knowledge models, which were validated against CELR and the Cisco reference architecture. The corresponding functional dependency model is shown in Figure 6-2 and consists of a control center, compressor station, two pipeline segments, two substations, and their internal components. Each of the compressor station components could be mapped to the physical resource devices for HIL research and training purposes.

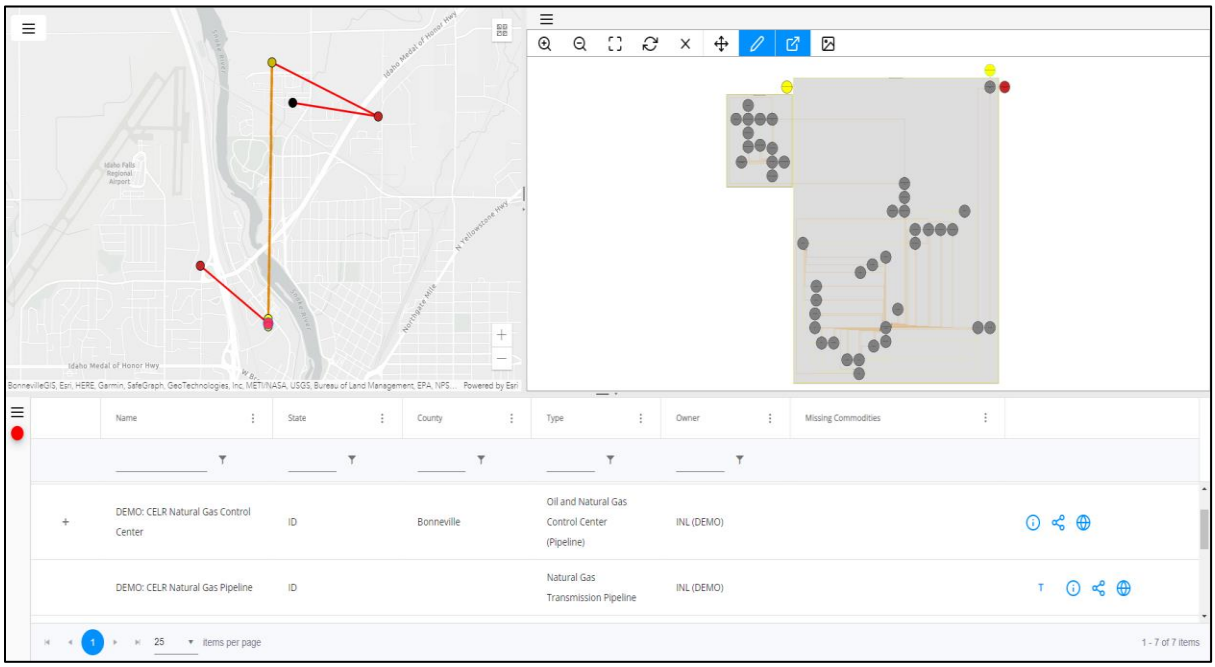

Figure 6-2 Notional CPS Natural Gas System Model.

Test Case 1: Physical Pipeline Disruption

Test case 1 was developed to test the time-dependent FFL ability to evaluate the impacts of a physical process disruption on the OT and IT functions of the notional CPS. In this case, the pipeline segment responsible for feeding natural gas to the compressor station was disabled. This would be representative of a pipeline rupture or leak. The results of the test case are shown in Table 6-1. The disruption of the pipeline impacts the compressor station components that require natural gas to maintain a normal operational state, and because the strength of dependency is considered critical, the compressors, control valves, scrubber, and aftercooler are set to a critical state. As constructed, the model indicated metering equipment would experience low impacts; however, the absence of gas through a control valve would not impact the operational state of a metering device. The leak detection and historian components also were shown to be impacted due to the failure propagation of the metering devices via degraded metering requirements not being met.

Test Case 2: Network Router Disruption

Test case 2 was developed to test the time-dependent FFL ability to evaluate the impacts of a communication network device disruption on the OT and IT functions of the notional CPS. In this case, the control center aggregation router was disabled. The aggregation routers consolidated traffic from local area networks that is intended to be transmitted over wide area networks and vice versa. The results of the test case are shown in Table 6-1. In test case 2, we observe impacts to the function

of the process equipment and the SCADA components within the compressor station as well as in the SCADA and decision support domains of the control center. This test case demonstrates that the FFL is able to capture component impact from the disruption of the single path for control and metering between the control center and the compressor station.

Test Case 3: Programmable Logic Controller Disruption

Test case 3 was developed to test the time-dependent FFL ability to evaluate the impacts of a control device disruption on the OT and IT functions of the notional CPS. In this case, the PLC controlling the compressor station was disabled. The results of the test case are shown in Table 6-1. The disruption of the PLC impacts the compressor station components that require control signals to maintain a nominal operational state and because the strength of dependency is considered critical for compressors, control valves, and cooling fans. This test case demonstrates that the FFL can capture component impact from the disruption of the PLC.

Test Case 4: Engineering Workstation Disruption

Test case 4 was developed to test the time-dependent FFL ability to evaluate the impacts of a engineering workstation disruption on the OT and IT functions of the notional CPS. In this case, the workstation provides control logic to the PLC and was intended to represent a workstation compromise. The results of the test case are shown in Table 6-1 and have similar results to test case 3. However, this test case highlights the need for critical analysis of model results. The simple disruption of an engineering workstation would not result in impacts described by the model; however, if the disruption was due to a threat actor who had the intention of modifying control logic, it is plausible.

Table 6-1 CPS Component States by Test Case.

CPS Component States by Test Case						
			Case 1	Case 2	Case 3	Case 4
Components	Component Type	t=0	t=1	t=1	t=1	t=1
Compressor Station						
M-200-4 Compressor	Compressor	OP	Crit	Sig	Crit	Crit
M-200-3 Compressor	Compressor	OP	Crit	Sig	Crit	Crit
M-200-2 Compressor	Compressor	OP	Crit	Sig	Crit	Crit
M-200-1 Compressor	Compressor	OP	Crit	Sig	Crit	Crit
M-100 Compressor	Compressor	OP	Crit	Sig	Crit	Crit
FV-410	Control Valve	OP	Crit	Sig	Crit	Crit

Table 6-1 Continued.

CPS Component States by Test Case						
			Case 1	Case 2	Case 3	Case 4
Components	Component Type	t=0	t=1	t=1	t=1	t=1
Compressor Station						
PV-420	Control Valve	OP	Sig	Sig	Sig	Sig
PV-400	Control Valve	OP	Crit	Sig	Crit	Crit
FV-101	Control Valve	OP	Crit	Sig	Crit	Crit
PV-100	Control Valve	OP	Crit	Sig	Crit	Crit
M-300-4 Fan	Cooling Fan	OP	Nom	Sig	Crit	Crit
M-300-3 Fan	Cooling Fan	OP	Nom	Sig	Crit	Crit
M-300-2 Fan	Cooling Fan	OP	Nom	Sig	Crit	Crit
M-300-1 Fan	Cooling Fan	OP	Nom	Sig	Crit	Crit
Aftercooler	Gas Cooler	OP	Crit	Sig	Crit	Crit
Scrubber	Gas Scrubber	OP	Crit	Sig	Crit	Crit
Maple Systems HMI	Human-Machine Interface (HMI)	OP	Sig	Sig	Crit	Crit
PIT-200 Meter	Meter	OP	Low	Low	Low	Low
TIT-310 Meter	Meter	OP	Low	Low	Low	Low
FIT-410 Meter	Meter	OP	Low	Low	Low	Low
PIT-310 Meter	Meter	OP	Low	Low	Low	Low
TIT-300 Meter	Meter	OP	Low	Low	Low	Low
PIT-120 Meter	Meter	OP	Low	Low	Low	Low
FIT-110 Meter	Meter	OP	Low	Low	Low	Low
TIT-100 Meter	Meter	OP	Low	Low	Low	Low
CELR Compressor Firewall	Network Firewall	OP	Nom	Nom	Nom	Nom
CELR Compressor Aggregation Router	Network Router	OP	Nom	Nom	Nom	Nom
CELR Compressor Network Switch	Network Switch	OP	Nom	Nom	Nom	Nom

Table 6-1 Continued.

CPS Component States by Test Case						
			Case 1	Case 2	Case 3	Case 4
Components	Component Type	t=0	t=1	t=1	t=1	t=1
Control Center						
CELR SCADA Application Server (Leak Detection)	Control System Applications Server	OP	Sig	Sig	Sig	Sig
CELR Master SCADA Historian	Control System Historian	OP	Sig	Sig	Sig	Sig
CELR ETRM System	Database Server	OP	Nom	Nom	Nom	Nom
CELR SCADA Domain Controller	Domain Controller	OP	Nom	Nom	Nom	Nom
CELR Engineering Workstation	Engineering Workstation	OP	Nom	Nom	Nom	Dis
CELR Decision Support Firewall	Network Firewall	OP	Nom	Nom	Nom	Nom
WAN Firewall	Network Firewall	OP	Nom	Nom	Nom	Nom
CELR SCADA Firewall	Network Firewall	OP	Nom	Nom	Nom	Nom
CELR Decision Support Router	Network Router	OP	Nom	Nom	Nom	Nom
CELR SCADA Router	Network Router	OP	Nom	Nom	Nom	Nom
CELR SCADA MCC Aggregation Router	Network Router	OP	Nom	Dis	Nom	Nom
CELR Operator Workstation	Operator Workstation	OP	Nom	Nom	Crit	Nom
CELR Nomination System	Web Application Servers	OP	Nom	Nom	Nom	Nom

Discussion

The four test cases demonstrate how the AHA knowledge graph and time-dependent FFL provide an approach to evaluate CPS systems. These test cases provide evidence that the FFL could be used to enable cyber-informed engineering and potentially enhance training by integrating with CELR or other HIL resources. Careful analysis of the propagation paths can inform system design, inform mitigation requirements, or aid in understanding the consequence of disruptions. In addition, this method overcomes the spatial and temporal limitations described by [45]. However, there are several limitations of the current approach which include the ability to evaluate combinatorial degradations and modeler bias. For example, the loss of single metering dependency would have little or no impact on overall pipeline operations, but if the ability to receive readings from multiple sensors was impacted, the event would have significant impacts to pipeline operations.

Conclusions

This chapter demonstrated how the AHA time-dependent simulation capability can be used to model and simulate disruptions of CPS functions and their potential impacts for risk and resilience assessments of a notional natural gas pipeline by leveraging the Electric Connected Pipeline Reference Architecture developed by Schneider Electric and Cisco and CELR Natural Gas Compressor Skid. In addition, this research proposed a HIL CPS architecture that could be used to enhance CPS security research, mitigation testing, and stakeholder training.

By demonstrating how the AHA knowledge graph and time-dependent FFL could be used to test CPS designs, this research provides evidence that functional-basis-informed graphs provide an ideal structure for modeling function, commodity, and service flows of interconnected systems and facilitate scalable and repeatable assessments of system behaviors suitable for vulnerability, consequence, and risk analysis. In addition, a potential approach for constructing a HIL-based CPS that can enable future consequence-driven cyber-informed engineering research was provided.

Future work will consider the possibility of HIL simulation to more accurately account for the continuous nature of physical processes. Capabilities like the DHS CELR might provide ideal mechanisms to create highly customizable and immersive environments that could be tailored for CPS design, mitigation, and training needs.

Chapter 7: AHA Data Collection and Processing

The process of assessing the risks and resilience of interconnected infrastructure at scale is challenging because there is not a comprehensive data set that describes all infrastructure and their dependency relationships [8]. This is partly due to their dynamic nature, spatial distribution, and diverse ownership, with approximately 85 percent of the critical infrastructures in the United States owned and operated by the private sector entities. This makes collecting information on their existence, location, condition, and dependency relationships difficult to achieve at scale. However, for many infrastructure sectors, there are significant amounts of information contained in both structure and unstructured data sources, such as regulatory databases and after-action reports. However, the ability to effectively leverage this publicly available information requires new methods and tools to sift through, extract, and transform the content pertaining to infrastructure into actionable knowledge. A knowledge system such as this requires the development of novel approaches that leverage advances in the fields of NLP, IE, and IR. This chapter describes the proposed AHA TAS and the Infrastructure Miner (*I-Miner*) algorithm that was developed to address the initial requirement to identify and extract references to named infrastructures from unstructured text.

Natural Language Processing for Critical Infrastructure Information

This section discusses the NER Critical Infrastructure project, which includes both the Corpus Development and Training Tool (CDTT) and the Infrastructure Miner (I-Miner) algorithm. CDTT is a java-based application designed for corpus development to facilitate training of OpenNLP supervised document categorization and named entity recognition models. I-Miner is a supervised, named entity classification system developed to optimize and address issues with extracting named infrastructure from web content. I-Miner leverages the OpenNLP Libraries [33], in conjunction with sector-specific infrastructure dictionaries, to detect named infrastructure more efficiently with minimal training. I-Miner has also been integrated with CDTT to facilitate rapid testing. Figure 7-1 outlines the CDTT and I-Miner process flow.

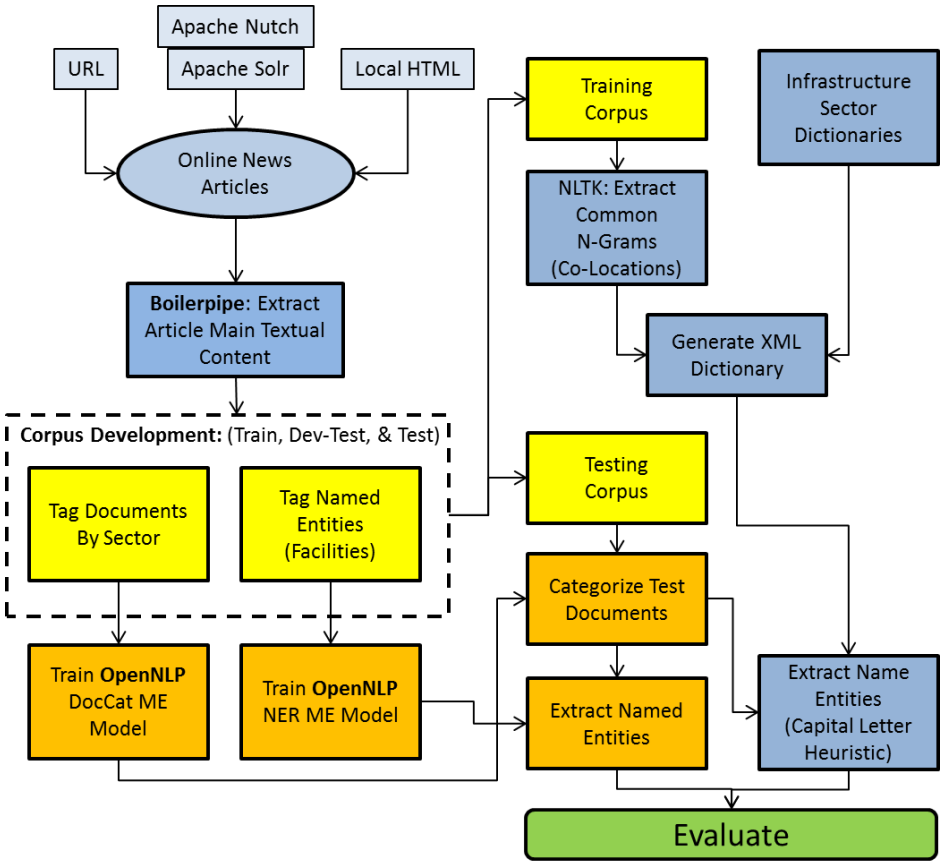

Figure 7-1 I-Miner Process Flow.

As implemented, potentially configurable parameters are static. The *DocumentCategorizerME* parameters are set to 8 for cutoff and 1,000 for iterations. The NameFinderME parameters are set to only process the FAC tag and the default of 100 iterations. NER is an NLP task and is commonly used for IE and retrieval applications, such as web content mining and knowledge base population (KBP) [1][2]. NER addresses the need to identify and tag certain types of entities in unstructured text resources. The simplest approach to NER is using a gazetteer of all known entities of interest to iteratively search each resource for entities contained within the gazetteer. However, this approach is challenged by the difficulties of creating and maintaining comprehensive lists of all possible entities, as well as resolving ambiguities [3]. To overcome these limitations, methods for automatically labeling textual features as named entities were developed based on machine-learning algorithms, such as MAXENT [4][5]. These models are most often trained using a manually annotated corpus of documents and typically include the following entities: people, locations, and organizations [6]. Currently, there are a number of trained NER models freely available such as those included with Apache's OpenNLP Name Finder [7]. While these models work very well in many applications that require the labeling of the above mentioned entities, they are limiting in domain-specific applications

[8]. To utilize this type of algorithm, it would require that a new corpus of documents be collected and manually annotated to train a new model specific to that domain. This is the preferred approach when the use case and domain is well-defined; however for dynamic use cases, this becomes restrictive due to the potential for overfitting. For example, in the infrastructure domain, a researcher may be only interested in certain type or class of infrastructure, so the ability to design a method or tool that provides flexibility and reduces the need to collect and annotated a significant number of corpora is desired. In order to realize the benefits of web content, data, and documents, more efficient methods for web content mining must be developed.

In this section, the Infrastructure Miner algorithm is presented as a method to optimize and address the issues with extracting named infrastructure from web content. Infrastructure Miner combines existing robust NER models provided with OpenNLP Name Finder [7] and the Stanford Named Entity Recognizer [9] with a keyword infrastructure list and several heuristics in order to efficiently detect infrastructure with minimal training.

This next section will cover related work, and the following is a section on “Corpus Development.” The “Algorithm” section is dedicated to an in-depth discussion of the Infrastructure Miner algorithm. Results are present in the “Experimental Results” section, and future work is cover in this chapter’s final section.

Related Work

The algorithm contains as its essential idea the concept of using classifiers to identify and extract infrastructure from web content. It is, therefore, worthwhile to review the concept of the classifier and examine the way it is used in the Infrastructure Miner algorithm. The two approaches covered in this section are the Stanford Named Entity Recognizer [9] and the Apache OpenNLP Name Finder [7].

The Stanford NER is a probabilistic-named entity classifier based on CRF [9]. It incorporates non-local structures using Gibbs Sampling, a Markov chain Monte Carlo algorithm, and simulated annealing to produce long distance dependency models often found in natural languages and has been demonstrated to be effective for named entity extraction.

The OpenNLP Name Finder is based on the MAXENT algorithm [4]. MAXENT is a statistical technique that maintains as much uncertainty as possible based on a set of constraints to classify without any prior assumptions about the probability distribution. For a comprehensive review of MAXENT as it applies to NLP, review Berger et al. [10].

Corpus Development

The test corpus was constructed from 100 web-based new articles collected over a 1-month period that specifically identified named infrastructures, for example the “Golden Gate Bridge.” Care was taken to ensure that no single source (e.g., *New York Times*) contributed a significant number of articles. Each article was examined and manually annotated to identify named infrastructures. During the annotation process, a keyword list of specific infrastructure was also generated. See Table 7-I for a complete list. The test corpus contained 180 total named infrastructures and a total of 393,114 words.

Algorithm

The algorithm can be broken down into four parts as describe in Figure 7-2 These are the classifiers, the acronym heuristic, the capital letter heuristic, and false positive removal. The process begins with a document being passed into the algorithm. This study is limited to HTML documents since web content is being targeted which is usually in HTML format. The documents used were downloaded from the Internet and stored on the computer running the algorithm. Next, the HTML formatting is stripped; HTML Cleaner [11] was used to accomplish this, and the text is formatted to construct a continuous string of text. Excessive spacing is removed leaving a single space between words. This is necessary to keep from introducing unconventionally structured content to the classifiers. The text is passed into both the Stanford NER classifier and the Apache OpenNLP Name Finder classifier. The classifiers have been given the models necessary to allow them to look for people, organizations, and locations. Obtaining these entities is done with the motivation of extracting the complete name of the infrastructure being searched for. It can be noted that a large amount of infrastructure is named in the structure of having one of these entities (people, organizations, or locations) followed by the type of infrastructure. For example, “George P. Burdell Bridge” is the name of a person followed by the type of infrastructure that is being searched for.

Pseudo Code for Infrastructure Miner
1. Input: a web document *d*
2. L is an empty list
3. LI is a list containing infrastructure
4. LS is an empty list for solutions
5. do
6. read in document
7. strip HTML formatting
8. eliminate extra spacing
9. compute Stanford classification
10. compute Apache OpenNLP classification
11. add the entities found to L
12. for each entity e in L
13. for each infrastructure i in LI
14. if e is next to i in d
15. add e to LS
16. end for
17. end for
18. compute Acronym Heuristic
19. compute Capital Letter Heuristic
20. add the entities found to LS
21. compute false positive removal on LS
22. Output: list LS containing the infrastructure found

It is important to note that the classifiers will often extract the whole name of the infrastructure meaning the substring extracted already includes the type of infrastructure being looked for. This is addressed by first checking to see if the last word of the entity found is in the list of infrastructure desired. The whole name is added to the list of solutions if this is the case. If this is not the case, then the algorithm sees if the named entity found by the classifiers is next to an infrastructure in the text. If this is the case, the entity is added to the list of solutions. The performance is highly dependent on the general type of infrastructure being in the list the algorithm uses. New infrastructure can be readily added to this list. If the structure is not in this list, it is highly unlikely that the algorithm will find it. This is a defining feature because it allows the user to narrow down the type of infrastructure that will

be searched for. This can optimize computational complexity, memory usage, and time because unneeded structures will not be looked for. The selected infrastructure systems for this study are presented in Table 7-1.

Table 7-1 Selected Infrastructure Systems.

Sector	Infrastructures
Air Travel	Airport, international airport, airfield, and airstrip.
Examples: "Boston International Airport," "John Doe Airfield," and "Idaho Falls Airport."	
Water Works	Aqueduct, treatment plant, wastewater treatment plant, water treatment plant, water treatment facility, water reclamation facility, wastewater control facility, sewage treatment plant, sewage plant, reservoir, reservoir canal, bridge, dam, tunnel, dock, basin, drainage, flow control structure, diversion, levee, canal, canal enclosure, seaport, retaining wall, retaining walls, culvert, embankment, flood bank, and stop bank, flowline.
Examples: "Point of the Mountain Water Treatment Plant," "Rhode Island Water Reclamation Facility," "Red Basin," and "Idaho Wastewater Control Facility."	
Power Generation	Plant, electrical generating station, power plant, nuclear power plant, power facility, power station, generation station, generating station, solar power plant, and wind farm.
Examples: "Utah Electrical generating station," "Austin Nuclear Power Plant," and "Colorado Generation Station."	
Research Facilities	National laboratory, national laboratories, and observatory.
Examples: "Idaho National Laboratory," "Sandia National Laboratories," and "Australia Observatory."	
General Services	Landfill, subway, tower, police station, railroad, fire department, fire station, hospital, medical center, medical care center, hospital center, teaching hospital, healthcare clinic, health care center, and medical agency.
Examples: "New York Subway," "Dalton Fire Department," and "California Teaching Hospital."	

The first heuristic to be discussed is the acronym heuristic. This is based on the observation that some text's infrastructure will occasionally have an acronym of its whole name immediately after the infrastructure has been mentioned. It is posited that following a brute force search for an infrastructure, if there is an acronym next to the name of the infrastructure, then it is a sound assumption that an abbreviation of the infrastructure's name is contained in the acronym. This can be taken advantage of because the first letter within the acronym will indicate how far back to look within the text to extract the whole name of the infrastructure. In summary, the process is to look for the first word starting with the first letter of the acronym found and then select the whole phrase starting from the word that starts with the first letter in the acronym and ending with the name of the infrastructure.

The next heuristic used is the capital letter heuristic. A brute force search for the infrastructure being looked for is performed, and the indexes of each are stored. The words to the left of the structure are examined, and if they start with capital letters, they keep being incorporated into a possible entity until no more capital letters at the beginning of a word are found. Words such as "of the" and "of" are skipped in the sense that they do not need to be capitalized, and this heuristic will keep searching for words starting with a capital letter after skipping these words because they are often a part of an entity. For example, for the entity "Point of the Mountain Aqueduct," if the words "of the" were not skipped, this heuristic would only extract "Mountain Aqueduct" because it would find that "of the" do not begin with capital letters and thus assume it has already extracted the entire entity. The found entities are then put into the list of possible solutions in which both the classifiers and acronym heuristic have deposited their findings.

Finally, the list of possible solutions must be processed for false positives. The first task is to remove duplicates from the list. Duplicates can arise because each of the classifiers could have identified the same entity or because either of the heuristics could have identified the same entity. Next, it was noted that many of the entities in the list started with the word "the;" this seems to be an error that is being made by the classifiers. The word "the" is removed, and the remaining of the string is placed back into the list, but first one should check that it is not already present in the list, as this would make it a duplicate. All the instances of infrastructure that do not carry any additional information with them are also removed.

There seems to be an error being made by both the classifiers and the heuristics, which has been designated as the "chaining problem." This occurs whenever there is a list of infrastructure in the text or when an error has been made in the initial processing and formatting of the text causing the

concatenation of multiple words without correct spacing in between them. The entire sequence is identified as an entity, and because it ends with an infrastructure, the sequence is incorrectly designated as a solution. The chaining problem is partly addressed by iterating over the proposed solutions and checking if a single unit contains more than one infrastructure. If this is the case, the algorithm backtracks starting from the end of the string toward the beginning of the string dividing the unit into fragments upon encountering an infrastructure present in the list mentioned previously. This mechanism mostly works if the chain was caused by a list of infrastructure. For the other type of chain, the one caused by formatting errors, the algorithm does not currently have an effective method of unraveling it. One procedure that appears to work decently well is to just remove the chain from the list of possible solutions if it is significantly longer than the rest. At this point, the algorithm is finished, and the remaining units in the list of solutions is the output.

Experimental Results

Infrastructure Miner's performance was evaluated on the test corpus. Table 7-2 presents a condition matrix of the results, which was achieved by running the algorithm against all 100 articles. Of the 180 manually identified entities, 157 were correctly identified, 23 were missing, and 72 were falsely identified. Table 7-3 presents the Infrastructure Miner's performance metrics.

Table 7-2 Condition Matrix of the Results.

		Condition	
		Positive	Negative
Outcome	Positive	157	72
	Negative	23	NA

Table 7-3 Infrastructure Miner's Performance Metrics.

Performance Metrics	
Precision	0.68
Recall	0.87
F-Measure	0.76

It is pertinent to describe what has been defined as a "correct" entity. For a sequence to be considered correct, it must be the entire name of the infrastructure and nothing else before or after it. It is

important to note that many of the false positives found in the study were truncated sections of the whole name of an entity. It appears to be relatively simple to resolve this by checking if an entity is a substring of another and eliminating it if it is. However, the algorithm does not know whether the smaller sequence is the correct entry, and the longer one has unneeded information. This has been left as is in an attempt to consolidate the struggle between precision, in other words obtaining all the entities present and recall. For this study, precision is the most important metric since the objective is to correctly extract all the infrastructure available in the text.

There is an overwhelmingly large portion of negative cases as shown by the true negative measure, which is at 99.9 percent. Consequently, the accuracy is also at 99.9 percent because the negative cases were accurately categorized as being negative, and the positive samples can hardly disturb the accuracy because of their small weight in proportion to the negative samples. This highlights the difficulty of obtaining positive samples since they are truly rare in relation to the total number of samples. In fact, when the articles were parsed by hand, a total of 180 instances of infrastructure were found. For simplicity, it can be assumed that each entity is two words in length. This would mean the proportion of positive samples to negative samples is 0.00009. This is given by taking the 180 entities and dividing them by 196,557, which is the total number of words present in the articles divided by two since it is being assumed that an entity is two words in length.

Special attention was given to selecting a wide variety of articles to submit to the algorithm in this study. The motivation for this is that it creates a more realistic testing environment given the purpose of this work. This poses a few challenges, especially for the initial parsing of the document. Different newspapers structure their content in different manners. The structures of the web pages also differ, which creates a challenge in regard to parsing and setting up the text for processing. The sidebars are particularly difficult to parse since they can also contain infrastructure. They are not in a regularly structured manner, which makes it difficult for the classifiers to pinpoint an entity since they do not have context to work with. The heuristics are much more efficient at pulling the infrastructure from the sidebars, but they can still be challenged by the structuring. Entities not visible but are present on the web page are also problematic since they cannot be accounted for in the initial manual parsing but make themselves existent when they are read by a computer. These were counted as missed entities in the interest of this study’s integrity.

The infrastructure miner framework has a few limitations that need to be discussed. The most pertinent one is that the text being analyzed must explicitly name a structure for it to be extracted. Another limitation which may in fact be a feature depending on the user case is that only the

infrastructure residing in the list will be distinguished as being an entity. This is a limitation if one wishes to look for all the existent infrastructure in a text and a feature if one wants to look for specific types of infrastructure. The general recognition technique is not exclusively limited to infrastructure. This domain has been targeted, but any entity which follows the pattern of having people, organizations, or locations preceding the general type of entity or for which the heuristics can apply can be found using Infrastructure Miner.

Future Work

One of the most crucial areas in need of further work is false positive removal. Specifically, the problem of having an entity that is a truncated version of a whole entity being present in the list of solutions along with the whole entity needs to be addressed. This algorithm does not have any way of differentiating between these. Another area for improvement is the need for a better parser and HTML format remover. It would be of benefit to have a parser that could distinguish between main body text and sidelines in a web news article. This would give way to allow for creating techniques for mining specifically the sidelines and keep from confusing the classifiers with unconventionally structured text. Some of the HTML formatting code was not removed by the stripper used; this could also help improve the metrics.

This algorithm cannot extract non-explicitly mentioned infrastructure. This is a more advanced problem which is worth looking into for the sake of obtaining more data from the processed texts. Finally, this algorithm is the first step toward a complete solution to the problem of analyzing infrastructure interdependence in each locale. Connecting the found entities with each other in respect to interdependence is the next step.

Chapter 8: Conclusions and Future Work

Research Overview and Objectives

Understanding the risk to and resilience of the interconnected infrastructure systems and the supply chain they enable are essential for federal, state, and local governments, as well as for the infrastructure owner and operators themselves. However, the wide range of spatial, temporal, operational, organizational, and interdependent characteristics of these interconnected systems poses unique challenges. Analysis of their risks and resilience is further complicated by the convergence of cyber-physical technologies, completing business demands, and the ever evolving threat and hazard landscape, where unseen interdependencies can result in uncontrollable cascading impacts [140].

In response, considerable resources have been expended to reduce risk and increase resilience of America's infrastructure systems; however, to address this grand challenge, federal agencies, state and local government, infrastructure owners and operators, security, emergency management, and business continuity functions still seek to [106]:

- Expand the visibility of risks to infrastructure, systems, and networks to help mitigate risks
- Advance risk analytic capabilities and methodologies
- Enhance security and risk mitigation guidance and impacts
- Build greater stakeholder capacity in infrastructure and network security resilience.

To address these objectives, academic, government, and private sector organizations have conducted significant amount of research to better understand and characterize the risk to and resilience of infrastructure systems, which have resulted in developing analytic frameworks, system-of-system modeling techniques, and advance decision support systems. This dissertation seeks to add to and advance this body of knowledge by proposing a novel all-hazard analysis methodology and KMS that encodes foundational engineering principles into a comprehensive knowledge graph for the purpose of understanding infrastructure behavior to better inform risk and resilience decision-making, as well as crisis action response. Due to the flexibility of the approach, it can be easily extended to analyze business function for continuity of operations planning, as well as incorporate additional modeling and simulation capabilities through an integrated python interface.

The remainder of this chapter reviews the research objectives outlined in "Chapter 1: Introduction" and discusses how the research presented in this dissertation addresses those objectives in defense of the following thesis statement.

Thesis: *Functional-basis-informed graphs are ideal for describing and analyzing interconnected infrastructure system behavior under all-hazard conditions. Functional-basis-informed graphs provide an optimal structure for modeling function, commodity, and service flows of interconnected systems and facilitate scalable and repeatable assessments of system behaviors suitable for vulnerability, consequence, and risk analysis.*

Objective 1: A functional basis for engineered infrastructure systems was developed to facilitate a scalable, robust, and repeatable process for the development of dependency models of interconnected infrastructure systems.

"Chapter 3: All-Hazards Analysis (AHA) Methodology" describes the development process for and resulting functional basis for engineered infrastructure systems generated by leveraging the AHA methodology. Currently, the functional basis contains three core facility functions, which include produce, store, and transport, and 288 unique dependency types, which represent the flows. These have resulted in the development of 330 unique dependency profiles that describe the general functional requirements and outputs of infrastructure assets and component types. This approach has been shown to provide a robust and adaptable knowledge model required to facilitate the collection, storage, and analysis of dependency information suitable for risk and resilience analysis in support of crisis action and strategic risk mitigation activities through the use case presented in "Chapter 5: Application of the AHA Methodology to the Colonial Pipeline System." "Chapter 6: Preliminary Cyber-Physical Functional-Flow Model Analysis" further validates the approach by applying it to CPSs.

Objective 2: A functional-flow network modeling framework was developed to model the behavior of engineered infrastructure systems for the purpose of risk and resilience assessments.

"Chapter 3: All-Hazards Analysis (AHA) Methodology" describes the development of the AHA methodology which models infrastructure dependencies as functional flows of commodities, services, or datum between infrastructure entities. The methodology provides a structured approach to decompose infrastructure systems into their core functional assets and components through a dynamic application function-based engineering design. Encoding the formal function representations from the functional basis as dependency profiles in a hierarchical taxonomy provides a standardized and scalable set of infrastructure models needed to support system-of-systems behavior modeling. Furthermore, this approach provides the flexibility to model systems at different levels of fidelity based on the best available data, while still providing actionable information to the risk and resilience decision-making process. Finally, the knowledge graph concept provides an efficient structure for

encoding the knowledge model, storing the modeled infrastructure information, and linking the entities to knowledge artifacts in the metamodel. To date, over 1.3 million nodes, approximately 1 million dependency relationships have been added to the AHA knowledge base.

Objective 3: The ability of functional-flow network models to simulate the behavior of interconnected infrastructure systems, including their scalability and robustness, was assessed.

"Chapter 3: All-Hazards Analysis (AHA) Methodology" describes the development of the simple cascade and time-dependent cascade simulation approaches currently implemented in the AHA Framework and validates them against the 2021 Colonial Pipeline ransomware attack in "Chapter 5: Application of the AHA Methodology to the Colonial Pipeline System." This application demonstrated that both methods provide the ability to generate actionable information to inform the risk and resilience decision process. For example, the qualitative results of the simple simulation demonstrate the ability to rapidly understand the range of a particular systems of interest's influence and identify the potential for cascading impacts. The time-dependent cascade simulation can provide time sequenced cascades and incorporate corrective course-of-actions if desired. A benefit of this semiquantitative approach is that it is tolerant to missing temporal information defaulting to simple cascade method as necessary. This feature can be used as a guide to identify potential knowledge gaps and focus future data gathering activities and analysis. In addition, to the Colonial Pipeline dependency model described in this dissertation, the AHA Framework has also been utilized to inform real-world risk and resilience decision-making, as well as table top exercises like the Army Cyber Institute's Jack Voltaic 3.0 exercise [141].

Objective 4: A graph-based KMS was developed to enable the collection, processing, and analysis of structured and unstructured infrastructure data required to model infrastructure behavior under all hazards.

Chapters 4 and 7 describe the AHA-KMS and TAS, respectively. "Chapter 4: AHA Knowledge Management System" provides an overview the AHA-KMS architecture and overview of the graphical user interfaces that facilitate the dynamic developing of the knowledge model, its population, and infrastructure behavior simulations. While "Chapter 7: AHA Data Collection and Processing" provides an overview the natural language process pipeline for the processing of internet-based artifacts including document categorization and NER. Initial results demonstrate the potential these approaches have in reducing the amount of time an analyst would be required to review, process, and synthesize the large volumes of infrastructure data contained on the world wide web.

Contribution and Limitations

The primary contribution of this dissertation is the development of a novel all-hazard analysis methodology and KMS for interconnected infrastructure systems. The methodology leverages research findings from multiple disciplines to construct function-based dependency models of infrastructure systems and simulate their behavior to identify cascading consequences of their disruption. The results of these simulations have the potential to inform both strategic risk and resilience decision-making processes at the federal, state, and local levels of government, as well as used to inform response-and-recovery courses of action during a crisis. In addition, this research demonstrates that the use of engineering design principles can overcome incomplete knowledge of the actual system, asset, or component under investigation.

However, a limitation of this research includes a lack of actuarial data on the system states and action taken by the Colonial Pipeline, terminal operators, and airports operators during the response-and-recovery efforts incident to assess model results more accurately.

Future Research

Events like the Colonial Pipeline ransomware attack, Texas polar vortex, Hurricane Maria, and many other have demonstrated the significant risk our infrastructure systems and supply chains where unseen interdependencies can result in uncontrollable cascading impacts. To mitigate the potential impacts of future events like these, additional research must be conducted to develop methods to enhance:

- The ability to design more resilient infrastructure solutions, identify potential vulnerabilities in existing systems, and evaluate mitigation strategies to better inform government and infrastructure owners on ways to optimize their investments
- The ability to better characterize and assess existing infrastructure systems and their dependency relationships for risk and resilience decision-making
- The understanding of the operational environment that influences the interdependent CPS's behavior
- The ability to more accurately model cyberattacks events utilizing HIL approaches
- The knowledge base population and visualization by leveraging advancements in machine learning.

References

[1] S. V. Buldyrev, R. Parshani, G. Paul, H. E. Stanley, and S. Havlin, "Catastrophic cascade of failures in interdependent networks," *Nature*, vol. 464, no. 7291, pp. 1025-1028, 2010.

[2] PCCIP. (1997). *Critical Foundations: Protecting America's Infrastructure*.

[3] P. P. Directive, "Presidential policy directive (PPD)-8: National preparedness," in *US Department of Homeland Security*, ed: The White House Washington, DC, 2011.

[4] U. S. D. o. H. Security, "Federal Continuity Directive 1: Federal Executive Branch National Continuity Program and Requirements1 (FCD 1)," in *Federal Executive Branch National Continuity Program and Requirements*, F. E. M. Agency, Ed., ed, 2017.

[5] U. S. D. o. H. Security, "Threat and Hazard Identification and Risk Assessment Guide: Comprehensive Preparedness Guide 201," 3rd ed: Homeland Security Washington, DC, USA, 2018.

[6] P. P. Directive, "Presidential policy directive (PPD)-21: Critical infrastructure security and resilience ", ed: The White House Washington, DC, 2013.

[7] U. S. D. o. H. Security, *National Infrastructure Protection Plan*. US Department of Homeland Security, 2013.

[8] M. Haggag, M. Ezzeldin, W. El-Dakhakhni, and E. Hassini, "Resilient cities critical infrastructure interdependence: a meta-research," *Sustainable and Resilient Infrastructure*, pp. 1-22, 2020, doi: 10.1080/23789689.2020.1795571.

[9] M. Ouyang, "Review on modeling and simulation of interdependent critical infrastructure systems," *Reliability Engineering & System Safety*, vol. 121, pp. 43-60, 2014/01/01/ 2014, doi: https://doi.org/10.1016/j.ress.2013.06.040.

[10] S. M. Rinaldi, J. P. Peerenboom, and T. K. Kelly, "Identifying, understanding, and analyzing critical infrastructure interdependencies," *IEEE control systems magazine*, vol. 21, no. 6, pp. 11-25, 2001.

[11] G. Satumtira and L. Dueñas-Osorio, "Synthesis of Modeling and Simulation Methods on Critical Infrastructure Interdependencies Research," in *Sustainable and Resilient Critical Infrastructure Systems: Simulation, Modeling, and Intelligent Engineering*, K. Gopalakrishnan and S. Peeta Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 1-51.

[12] C. W. King, J. D. Rhodes, and J. Zarnikau, "The Timeline and Events of the February 2021 Texas Electric Grid Blackouts," Energy Institute, University of Texas at Austin, 2021.

[13] O. Heino, A. Takala, P. Jukarainen, J. Kalalahti, T. Kekki, and P. Verho, "Critical Infrastructures: The Operational Environment in Cases of Severe Disruption," *Sustainability*, vol. 11, no. 3, p. 838, 2019. [Online]. Available: https://www.mdpi.com/2071-1050/11/3/838.

[14] INCOSE. "Engineered Systems." https://www.incose.org/about-systems-engineering/system-and-se-definition/engineered-system-definition (accessed June 14th, 2021.

[15] R. Setola and M. Theocharidou, "Modelling dependencies between critical infrastructures," in *Managing the complexity of critical infrastructures*: Springer, Cham, 2016, pp. 19-41.

[16] "supply chain," ed.

[17] U. S. D. o. H. Security, "Supply Chain Resilience Guide," F. E. M. Administration, Ed., ed. Washington, DC, USA: Homeland Security, 2019.

[18] J. Boyens, C. Paulsen, R. Moorthy, N. Bartol, and S. Shankles, "NIST Special Publication 800-161, Supply Chain Risk Management Practices for Federal Information Systems and Organizations," *NIST. April*, 2015.

[19] U. D. o. H. S. R. S. Committee, "DHS Risk Lexicon," *Homeland Security*, 2010.

[20] C. Curt and J. M. Tacnet, "Resilience of critical infrastructures: Review and analysis of current approaches," *Risk Analysis*, vol. 38, no. 11, pp. 2441-2458, 2018.

[21] R. E. Bloomfield, P. Popov, K. Salako, V. Stankovic, and D. Wright, "Preliminary interdependency analysis: An approach to support critical-infrastructure risk-assessment," *Reliability Engineering & System Safety*, vol. 167, pp. 198-217, 2017.
[22] G. Giannopoulos, R. Filippini, and M. Schimmer, "Risk assessment methodologies for Critical Infrastructure Protection. Part I: A state of the art," *JRC Technical Notes*, vol. 1, no. 1, pp. 1-53, 2012.
[23] R. Francis and B. Bekera, "A metric and frameworks for resilience analysis of engineered and infrastructure systems," *Reliability Engineering & System Safety*, vol. 121, pp. 90-103, 2014/01/01/ 2014, doi: https://doi.org/10.1016/j.ress.2013.07.004.
[24] N. Goldbeck, P. Angeloudis, and W. Y. Ochieng, "Resilience assessment for interdependent urban infrastructure systems using dynamic network flow models," *Reliability Engineering & System Safety*, vol. 188, pp. 62-79, 2019/08/01/ 2019, doi: https://doi.org/10.1016/j.ress.2019.03.007.
[25] R. Guidotti, "Regional risk and resilience analysis of interdependent critical infrastructure," University of Illinois at Urbana-Champaign, 2018.
[26] W. Liu and Z. Song, "Review of studies on the resilience of urban critical infrastructure networks," *Reliability Engineering & System Safety*, vol. 193, p. 106617, 2020.
[27] F. Petit *et al.*, "Resilience measurement index: An indicator of critical infrastructure resilience," Argonne National Lab.(ANL), Argonne, IL (United States), 2013.
[28] C. Poulin and M. B. Kane, "Infrastructure resilience curves: Performance measures and summary metrics," *Reliability Engineering & System Safety*, vol. 216, p. 107926, 2021.
[29] J. Wang, W. Zuo, L. Rhode-Barbarigos, X. Lu, J. Wang, and Y. Lin, "Literature review on modeling and simulation of energy infrastructures from a resilience perspective," *Reliability Engineering & System Safety*, vol. 183, pp. 360-373, 2019/03/01/ 2019, doi: https://doi.org/10.1016/j.ress.2018.11.029.
[30] S. Saidi, L. Kattan, P. Jayasinghe, P. Hettiaratchi, and J. Taron, "Integrated infrastructure systems—A review," *Sustainable Cities and Society*, vol. 36, pp. 1-11, 2018.
[31] I. Hernandez-Fajardo and L. Dueñas-Osorio, "Probabilistic study of cascading failures in complex interdependent lifeline systems," *Reliability Engineering & System Safety*, vol. 111, pp. 260-272, 2013.
[32] D. D. Dudenhoeffer, M. R. Permann, and M. Manic, "CIMS: A framework for infrastructure interdependency modeling and analysis," in *Proceedings of the 2006 winter simulation conference*, 2006: IEEE, pp. 478-485.
[33] S. E. Chang, T. L. McDaniels, J. Mikawoz, and K. Peterson, "Infrastructure failure interdependencies in extreme events: power outage consequences in the 1998 Ice Storm," *Natural Hazards*, vol. 41, no. 2, pp. 337-358, 2007/05/01 2007, doi: 10.1007/s11069-006-9039-4.
[34] D. Mendonça and W. A. Wallace, "Impacts of the 2001 world trade center attack on new york city critical infrastructures," *Journal of Infrastructure Systems*, vol. 12, no. 4, pp. 260-270, 2006.
[35] M. Ouyang, L. Hong, Z.-J. Mao, M.-H. Yu, and F. Qi, "A methodological approach to analyze vulnerability of interdependent infrastructures," *Simulation Modelling Practice and Theory*, vol. 17, no. 5, pp. 817-828, 2009/05/01/ 2009, doi: https://doi.org/10.1016/j.simpat.2009.02.001.
[36] T. Brown, W. Beyeler, and D. Barton, "Assessing infrastructure interdependencies: the challenge of risk analysis for complex adaptive systems," *International Journal of Critical Infrastructures*, vol. 1, no. 1, pp. 108-117, 2004.
[37] S. R. Hirshorn, L. D. Voss, and L. K. Bromley, "NASA systems engineering handbook," 2017.

[38] INCOSE, *International council on systems engineering. systems engineering handbook: a guide for system life cycle processes and activities*, 4th ed. Hoboken, New Jersey, USA: John Wiley & Sons, Inc, 2015.

[39] A. I. McInnes, B. K. Eames, and R. Grover, "Formalizing Functional Flow Block Diagrams Using Process Algebra and Metamodels," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 41, no. 1, pp. 34-49, 2011, doi: 10.1109/TSMCA.2010.2048749.

[40] D. Ward, M. Rossi, B. P. Sullivan, and H. V. Pichika, "The Metamorphosis of Systems Engineering through the evolution of today's standards," in *2018 IEEE International Systems Engineering Symposium (ISSE)*, 1-3 Oct. 2018 2018, pp. 1-8, doi: 10.1109/SysEng.2018.8544426.

[41] J. E. Long, "Relationships between common graphical representations used in system engineering," *INSIGHT*, vol. 21, no. 1, pp. 8-11, 2018.

[42] R. B. Stone and K. L. Wood, "Development of a Functional Basis for Design," 1999. [Online]. Available: https://doi.org/10.1115/DETC99/DTM-8765.

[43] J. Hirtz, R. B. Stone, D. A. McAdams, S. Szykman, and K. L. Wood, "A functional basis for engineering design: reconciling and evolving previous efforts," *Research in engineering Design*, vol. 13, no. 2, pp. 65-82, 2002.

[44] D. van Eck, D. A. McAdams, and P. E. Vermaas, "Functional Decomposition in Engineering: A Survey," 2007. [Online]. Available: https://doi.org/10.1115/DETC2007-34232.

[45] T. Kurtoglu and I. Y. Tumer, "A graph-based fault identification and propagation framework for functional design of complex systems," 2008.

[46] A. Barrat, M. Barthelemy, and A. Vespignani, *Dynamical processes on complex networks*. Cambridge university press, 2008.

[47] M. Barthélemy, "Spatial networks," *Physics Reports*, vol. 499, no. 1-3, pp. 1-101, 2011.

[48] S. Boccaletti *et al.*, "The structure and dynamics of multilayer networks," *Physics Reports*, vol. 544, no. 1, pp. 1-122, 2014/11/01/ 2014, doi: https://doi.org/10.1016/j.physrep.2014.07.001.

[49] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *Journal of complex networks*, vol. 2, no. 3, pp. 203-271, 2014.

[50] S. D. Wolthusen, "Modeling critical infrastructure requirements," in *Proceedings from the Fifth Annual IEEE SMC Information Assurance Workshop, 2004.*, 2004: IEEE, pp. 101-108.

[51] N. K. Svendsen and S. D. Wolthusen, "Connectivity models of interdependency in mixed-type critical infrastructure networks," *Information Security Technical Report*, vol. 12, no. 1, pp. 44-55, 2007/01/01/ 2007, doi: https://doi.org/10.1016/j.istr.2007.02.005.

[52] I. E. E. Lee, J. E. Mitchell, and W. A. Wallace, "Restoration of Services in Interdependent Infrastructure Systems: A Network Flows Approach," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1303-1317, 2007, doi: 10.1109/TSMCC.2007.905859.

[53] B. Arvidsson, J. Johansson, and N. Guldåker, "Critical infrastructure, geographical information science and risk governance: A systematic cross-field review," *Reliability Engineering & System Safety*, vol. 213, p. 107741, 2021.

[54] A. Dierich, K. Tzavella, N. J. Setiadi, A. Fekete, and F. M. Neisser, "Enhanced Crisis-Preparation of Critical Infrastructures through a Participatory Qualitative-Quantitative Interdependency Analysis Approach," in *ISCRAM*, 2019.

[55] R. K. McNally, S.-W. Lee, D. Yavagal, and W.-N. Xiang, "Learning the critical infrastructure interdependencies through an ontology-based information system," *Environment and Planning B: Planning and Design*, vol. 34, no. 6, pp. 1103-1124, 2007.

[56] A. Sweetser, "A comparison of system dynamics (SD) and discrete event simulation (DES)," in *17th International Conference of the System Dynamics Society*, 1999, pp. 20-23.

[57] A. A. Tako and S. Robinson, "The application of discrete event simulation and system dynamics in the logistics and supply chain context," *Decision Support Systems*, vol. 52, no. 4, pp. 802-815, 2012/03/01/ 2012, doi: https://doi.org/10.1016/j.dss.2011.11.015.
[58] A. M. Law, W. D. Kelton, and W. D. Kelton, *Simulation modeling and analysis*, 5 ed. McGraw-Hill New York, 2015.
[59] A. Tomar, H. V. Burton, A. Mosleh, and J. Yun Lee, "Hindcasting the Functional Loss and Restoration of the Napa Water System Following the 2014 Earthquake Using Discrete-Event Simulation," *Journal of Infrastructure Systems*, vol. 26, no. 4, p. 04020035, 2020.
[60] S. Auer, V. Kovtun, M. Prinz, A. Kasprzik, M. Stocker, and M. E. Vidal, "Towards a Knowledge Graph for Science," presented at the Proceedings of the 8th International Conference on Web Intelligence, Mining and Semantics, Novi Sad, Serbia, 2018. [Online]. Available: https://doi.org/10.1145/3227609.3227689.
[61] L. Ehrlinger and W. Wöß, "Towards a definition of knowledge graphs," *SEMANTiCS (Posters, Demos, SuCCESS)*, vol. 48, no. 1-4, p. 2, 2016.
[62] S. Ji, S. Pan, E. Cambria, P. Marttinen, and S. Y. Philip, "A survey on knowledge graphs: Representation, acquisition, and applications," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 2, pp. 494-514, 2021.
[63] T. R. Gruber, "Toward principles for the design of ontologies used for knowledge sharing?," *International journal of human-computer studies*, vol. 43, no. 5-6, pp. 907-928, 1995.
[64] A. C. R. Bryant, R. B. Stone, J. L. Greer, D. A. McAdams, T. Kurtoglu, and M. I. Campbell, "A function-based component ontology for systems design," in *DS 42: Proceedings of ICED 2007, the 16th International Conference on Engineering Design, Paris, France, 28.-31.07. 2007*, 2007, pp. 575-576 (exec. Summ.), full paper no. DS42_P_478.
[65] B. Abu-Salih, "Domain-specific knowledge graphs: A survey," *Journal of Network and Computer Applications*, vol. 185, p. 103076, 2021.
[66] S. Chakrabarti, *Mining the Web: Discovering knowledge from hypertext data*. Morgan Kaufmann, 2002.
[67] R. Kosala and H. Blockeel, "Web mining research: A survey," *ACM Sigkdd Explorations Newsletter*, vol. 2, no. 1, pp. 1-15, 2000.
[68] P. K. Mallick, S. Mishra, and G.-S. Chae, "Digital media news categorization using Bernoulli document model for web content convergence," *Personal and Ubiquitous Computing*, pp. 1-16, 2020.
[69] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to Information Retrieval*. Cambridge University Press, 2008.
[70] T. Poibeau, H. Saggion, J. Piskorski, and R. Yangarber, *Multi-source, multilingual information extraction and summarization*. Springer, 2013.
[71] K. Pol, N. Patil, S. Patankar, and C. Das, "A survey on web content mining and extraction of structured and semistructured data," in *2008 First international conference on emerging trends in engineering and technology*, 2008: IEEE, pp. 543-546.
[72] E. Cambria and B. White, "Jumping NLP curves: A review of natural language processing research," *IEEE Computational intelligence magazine*, vol. 9, no. 2, pp. 48-57, 2014.
[73] Y. Gu *et al.*, "Domain-Specific Language Model Pretraining for Biomedical Natural Language Processing," *ACM Transactions on Computing for Healthcare*, vol. 3, no. 1, pp. 1-23, 2022, doi: 10.1145/3458754.
[74] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 2, pp. 604-624, 2020.
[75] H. Schmid, "Part-of-speech tagging with neural networks," *arXiv preprint cmp-lg/9410018*, 1994.

[76] D. Yarowsky, "Unsupervised word sense disambiguation rivaling supervised methods," in *33rd annual meeting of the association for computational linguistics*, 1995, pp. 189-196.
[77] C. Manning and H. Schutze, *Foundations of statistical natural language processing*. MIT press, 1999.
[78] A. Halevy, P. Norvig, and F. Pereira, "The unreasonable effectiveness of data," *IEEE intelligent systems,* vol. 24, no. 2, pp. 8-12, 2009.
[79] D. W. Otter, J. R. Medina, and J. K. Kalita, "A Survey of the Usages of Deep Learning for Natural Language Processing," *IEEE Trans Neural Netw Learn Syst,* vol. 32, no. 2, pp. 604-624, Feb 2021, doi: 10.1109/TNNLS.2020.2979670.
[80] J. Sinclair, "Corpus and text-basic principles," *Developing linguistic corpora: A guide to good practice,* vol. 92, pp. 1-16, 2005.
[81] W. N. Francis and H. Kucera, "Brown corpus manual," *Letters to the Editor,* vol. 5, no. 2, p. 7, 1979.
[82] J. Pustejovsky and A. Stubbs, *Natural Language Annotation for Machine Learning: A guide to corpus-building for applications*. " O'Reilly Media, Inc.", 2012.
[83] C. N. Kamath, S. S. Bukhari, and A. Dengel, "Comparative Study between Traditional Machine Learning and Deep Learning Approaches for Text Classification," presented at the Proceedings of the ACM Symposium on Document Engineering 2018, Halifax, NS, Canada, 2018. [Online]. Available: https://doi.org/10.1145/3209280.3209526.
[84] M. E. Ruiz and P. Srinivasan, "Hierarchical text categorization using neural networks," *Information retrieval,* vol. 5, no. 1, pp. 87-118, 2002.
[85] P.-Y. Hao, J.-H. Chiang, and Y.-K. Tu, "Hierarchically SVM classification based on support vector clustering method and its application to document categorization," *Expert Systems with applications,* vol. 33, no. 3, pp. 627-635, 2007.
[86] M. Dredze, P. McNamee, D. Rao, A. Gerber, and T. Finin, "Entity disambiguation for knowledge base population," in *Proceedings of the 23rd International Conference on Computational Linguistics*, 2010.
[87] H. Ji, R. Grishman, H. T. Dang, K. Griffitt, and J. Ellis, "Overview of the TAC 2010 knowledge base population track," in *Third text analysis conference (TAC 2010)*, 2010, vol. 3, no. 2, pp. 3-3.
[88] X. Lin, H. Li, H. Xin, Z. Li, and L. Chen, "KBPearl: a knowledge base population system supported by joint entity and relation linking," *Proc. VLDB Endow.*, vol. 13, no. 7, pp. 1035–1049, 2020, doi: 10.14778/3384345.3384352.
[89] D. Rao, P. McNamee, and M. Dredze, "Entity linking: Finding extracted entities in a knowledge base," in *Multi-source, multilingual information extraction and summarization*: Springer, 2013, pp. 93-115.
[90] M. Ehrmann, A. Hamdi, E. L. Pontes, M. Romanello, and A. Doucet, "Named entity recognition and classification on historical documents: A survey," *arXiv preprint arXiv:2109.11406,* 2021.
[91] A. Goyal, V. Gupta, and M. Kumar, "Recent Named Entity Recognition and Classification techniques: A systematic review," *Computer Science Review,* vol. 29, pp. 21-43, 2018/08/01/ 2018, doi: https://doi.org/10.1016/j.cosrev.2018.06.001.
[92] D. Nadeau and S. Sekine, "A survey of named entity recognition and classification," *Lingvisticae Investigationes,* vol. 30, no. 1, pp. 3-26, 2007.
[93] A. P. Quimbaya *et al.*, "Named Entity Recognition Over Electronic Health Records Through a Combined Dictionary-based Approach," *Procedia Computer Science,* vol. 100, pp. 55-61, 2016/01/01/ 2016, doi: https://doi.org/10.1016/j.procs.2016.09.123.
[94] A. Douthat, "Appendix G: The Message Understanding Conference Scoring Software User's Manual," in *Seventh Message Understanding Conference (MUC-7): Proceedings of a Conference Held in Fairfax, Virginia, April 29-May 1, 1998*, 1998.

[95] A. Mikheev, M. Moens, and C. Grover, "Named entity recognition without gazetteers," in *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, 1999, pp. 1-8.
[96] D. Klein, J. Smarr, H. Nguyen, and C. D. Manning, "Named entity recognition with character-level models," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, 2003, pp. 180-183.
[97] L. F. Rau, "Extracting company names from text," in *Proceedings the Seventh IEEE Conference on Artificial Intelligence Application*, 1991: IEEE Computer Society, pp. 29, 30, 31, 32-29, 30, 31, 32.
[98] G. Zhou and J. Su, "Named entity recognition using an HMM-based chunk tagger," in *Proceedings of the 40th annual meeting of the association for computational linguistics*, 2002, pp. 473-480.
[99] J. P. Chiu and E. Nichols, "Named entity recognition with bidirectional LSTM-CNNs," *Transactions of the association for computational linguistics*, vol. 4, pp. 357-370, 2016.
[100] V. Yadav and S. Bethard, "A survey on recent advances in named entity recognition from deep learning models," *arXiv preprint arXiv:1910.11470*, 2019.
[101] X. Schmitt, S. Kubler, J. Robert, M. Papadakis, and Y. LeTraon, "A replicable comparison study of NER software: StanfordNLP, NLTK, OpenNLP, SpaCy, Gate," in *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*, 2019: IEEE, pp. 338-343.
[102] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky, "The Stanford CoreNLP natural language processing toolkit," in *Proceedings of 52nd annual meeting of the association for computational linguistics: system demonstrations*, 2014, pp. 55-60.
[103] A. Berger, S. A. Della Pietra, and V. J. Della Pietra, "A maximum entropy approach to natural language processing," *Computational linguistics*, vol. 22, no. 1, pp. 39-71, 1996.
[104] J. R. Finkel, T. Grenager, and C. D. Manning, "Incorporating non-local information into information extraction systems by gibbs sampling," in *Proceedings of the 43rd annual meeting of the association for computational linguistics (ACL'05)*, 2005, pp. 363-370.
[105] S. Robinson, "Conceptual modelling for simulation Part I: definition and requirements," *Journal of the Operational Research Society*, vol. 59, no. 3, pp. 278-290, 2008/03/01 2008, doi: 10.1057/palgrave.jors.2602368.
[106] (2022). *CISA 2023-2025 Strategic Plan*. [Online] Available: https://www.cisa.gov/sites/default/files/publications/StrategicPlan_20220912-V2_508c.pdf
[107] (2021). *Joint Risk Analysis Methodology (JRAM) Manual*. [Online] Available: https://www.jcs.mil/Portals/36/Documents/Library/Manuals/CJCSM%203105.01A.pdf?ver=y3cH4s5UNyqJAXwxAYCL5Q%3d%3d#:~:text=The%20JRAM%20presents%20a%20common,risk%20communication%20and%20decision%20making.
[108] NIPP, "National Infrastructure Protection Plan: Partnering for Critical Infrastructure Security and Resilience," 2013.
[109] C. NIST, "Community Resilience Planning Guide for Buildings and Infrastructure Systems, Volume II," Special Publication 1190, 2015.
[110] D. o. H. Security, "Risk management fundamentals: Homeland security risk management doctrine," ed: Department of Homeland Security Washington, DC, 2011.
[111] J. Teich, L. Thiele, and E. A. Lee, "Modeling and simulation of heterogeneous real-time systems based on a deterministic discrete event model," presented at the Proceedings of the 8th international symposium on System synthesis, Cannes, France, 1995. [Online]. Available: https://doi.org/10.1145/224486.224535.
[112] T. Kurtoglu, M. I. Campbell, J. Gonzalez, C. R. Bryant, R. B. Stone, and D. A. McAdams, "Capturing empirically derived design knowledge for creating conceptual design

configurations," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2005, vol. 4742, pp. 249-257.
[113] (2013). *Defense Acquisition Guidebook*.
[114] R. M. Blank, "Guide for conducting risk assessments," 2011.
[115] O. Alexander, M. Belisle, and J. Steele, "MITRE ATT&CK® for industrial control systems: Design and philosophy," *The MITRE Corporation: Bedford, MA, USA*, 2020.
[116] A. Amro, V. Gkioulos, and S. Katsikas, "Assessing Cyber Risk in Cyber-Physical Systems Using the ATT&CK Framework," *Preprint at http://dx. doi. org/10.13140/RG*, vol. 2, no. 16531.40484, 2021.
[117] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "MITRE ATT&CK: Design and Philosophy," MITRE CORP MCLEAN VA, 2018.
[118] E. M. Hutchins, M. J. Cloppert, and R. M. 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, no. 1, p. 80, 2011.
[119] J. C. Gill and B. D. Malamud, "Reviewing and visualizing the interactions of natural hazards," *Reviews of Geophysics*, vol. 52, no. 4, pp. 680-722, 2014.
[120] V. Masson-Delmotte *et al.*, "Climate change 2021: the physical science basis," *Contribution of working group I to the sixth assessment report of the intergovernmental panel on climate change*, vol. 2, 2021.
[121] H.-O. Pörtner *et al.*, "Climate change 2022: Impacts, adaptation and vulnerability," *IPCC Sixth Assessment Report*, 2022.
[122] (2015). *Intelligence Community Directive 206. Sourcing Requirements for Disseminated Intelligence Products*. [Online] Available: https://www.dni.gov/files/documents/ICD/ICD%20206.pdf
[123] S. Robinson, "Conceptual modelling for simulation Part II: a framework for conceptual modelling," *Journal of the Operational Research Society*, vol. 59, no. 3, pp. 291-304, 2008/03/01 2008, doi: 10.1057/palgrave.jors.2602369.
[124] P. R. Garvey and C. A. Pinto, "Introduction to functional dependency network analysis," in *The MITRE Corporation and Old Dominion, Second International Symposium on Engineering Systems, MIT, Cambridge, Massachusetts*, 2009, vol. 5.
[125] *Infrastructure Data Taxonomy (IDT)*.
[126] CESER, "CyOTE Case Study: DarkSide," Office of Cybersecuity, Energy Security, and Emergency Response, 2021. Accessed: September 25, 2022. [Online]. Available: https://ceser-design.gravisdev.com/wp-content/uploads/2021/12/DarkSide-CyOTE-Case-Study.pdf
[127] P. Derler, E. A. Lee, and A. S. Vincentelli, "Modeling cyber–physical systems," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 13-28, 2011.
[128] E. A. Lee, "The past, present and future of cyber-physical systems: A focus on models," *Sensors*, vol. 15, no. 3, pp. 4837-4869, 2015.
[129] A. Amro, V. Gkioulos, and S. Katsikas, "Assessing cyber risk in cyber-physical systems using the ATT&CK framework," *ACM Transactions on Privacy and Security*, vol. 26, no. 2, pp. 1-33, 2023.
[130] A. Kim, J. Oh, K. Kwon, and K. Lee, "Consider the consequences: a risk assessment approach for industrial control systems," *Security and Communication Networks*, vol. 2022, 2022.
[131] A. Tantawy, S. Abdelwahed, A. Erradi, and K. Shaban, "Model-based risk assessment for cyber physical systems security," *Computers & Security*, vol. 96, p. 101864, 2020.
[132] R. L. Grubbs, J. T. Stoddard, S. G. Freeman, and R. E. Fisher, "Evolution and Trends of Industrial Control System Cyber Incidents since 2017," *Journal of Critical Infrastructure Policy*, vol. 2, no. INL/JOU-21-65119-Rev000, 2021.

[133] K. Hemsley and R. Fisher, "A history of cyber incidents and threats involving industrial control systems," in *Critical Infrastructure Protection XII: 12th IFIP WG 11.10 International Conference, ICCIP 2018, Arlington, VA, USA, March 12-14, 2018, Revised Selected Papers 12*, 2018: Springer, pp. 215-242.

[134] T. Miller, A. Staves, S. Maesschalck, M. Sturdee, and B. Green, "Looking back to look forward: Lessons learnt from cyber-attacks on industrial control systems," *International Journal of Critical Infrastructure Protection,* vol. 35, p. 100464, 2021.

[135] U.S. Department of Energy (DOE), "National Cyber-Infromed Engineering Strategy," E. S. Office of Cybersecurity, and Emergency and R. (CESER), Eds., ed, 2022, p. 37.

[136] R. Rai and C. K. Sahu, "Driven by data or derived through physics? a review of hybrid physics guided machine learning techniques with cyber-physical system (cps) focus," *IEEE Access,* vol. 8, pp. 71050-71073, 2020.

[137] L. Faramondi, F. Flammini, S. Guarino, and R. Setola, "A hardware-in-the-loop water distribution testbed dataset for cyber-physical security testing," *IEEE Access,* vol. 9, pp. 122385-122396, 2021.

[138] Z. Liu, Q. Wang, and Y. Tang, "Design of a cosimulation platform with hardware-in-the-loop for cyber-attacks on cyber-physical power systems," *IEEE Access,* vol. 8, pp. 95997-96005, 2020.

[139] B. Potteiger, W. Emfinger, H. Neema, X. Koutosukos, C. Tang, and K. Stouffer, "Evaluating the effects of cyber-attacks on cyber physical systems using a hardware-in-the-loop simulation testbed," in *2017 Resilience Week (RWS)*, 2017: IEEE, pp. 177-183.

[140] J. W. Busby *et al.*, "Cascading risks: Understanding the 2021 winter blackout in Texas," *Energy Research & Social Science,* vol. 77, p. 102106, 2021.

[141] E. Mitchell *et al.*, "Jack voltaic 3.0 cyber research report," 2021.