2019-06

NETWORK SHAPING

Barrow, Howard J., III

Monterey, CA; Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

NETWORK SHAPING

by

Howard J. Barrow III

June 2019

Thesis Advisor:	David L. Alderson Jr.
Co-Advisor:	Daniel Eisenberg
Second Reader:	Jeffrey A. Appleget

THIS PAGE INTENTIONALLY LEFT BLANK

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

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2019	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE NETWORK SHAPING	**5. FUNDING NUMBERS**
6. AUTHOR(S) Howard J. Barrow III	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

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

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

13. ABSTRACT (maximum 200 words)

Network flow problems can be used to address any of the phases of the military joint operation model, but they do a poor job with the transition from interdiction to restoration activities. Previous research identifies methods to find the best- and worst-case scenarios for a given network, but do not show how interdiction activities affect restoration activities (or vice versa) and/or how we can make sense of these interactions in military planning. We develop a method of metagraph analysis to study various performance thresholds in a flow network and identify ways to interdict and restore systems not previously discussed in the literature. The presence of states not identified by traditional network flow problems indicates that, from an operational planning perspective, alternatives exist that may improve the attack and defense of a flow network. This result suggests that traditional interdiction and restoration methods prescribe only a subset of joint operational activities, and military operations would benefit from expanding analysis to consider more options. We define at least two ways to identify these options and conclude that there are system states not identified by traditional methods that can inform new ways to shape flow networks for military operations.

14. SUBJECT TERMS network, interdiction, critical threshold	**15. NUMBER OF PAGES** 71
	16. PRICE CODE

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

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

NETWORK SHAPING

Howard J. Barrow III
Major, United States Army
BA, Florida State University, 2006
MS, Florida Institute of Technology, 2015

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
June 2019

Approved by: David L. Alderson Jr.
Advisor

Daniel Eisenberg
Co-Advisor

Jeffrey A. Appleget
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Network flow problems can be used to address any of the phases of the military joint operation model, but they do a poor job with the transition from interdiction to restoration activities. Previous research identifies methods to find the best- and worst-case scenarios for a given network, but do not show how interdiction activities affect restoration activities (or vice versa) and/or how we can make sense of these interactions in military planning. We develop a method of metagraph analysis to study various performance thresholds in a flow network and identify ways to interdict and restore systems not previously discussed in the literature. The presence of states not identified by traditional network flow problems indicates that, from an operational planning perspective, alternatives exist that may improve the attack and defense of a flow network. This result suggests that traditional interdiction and restoration methods prescribe only a subset of joint operational activities, and military operations would benefit from expanding analysis to consider more options. We define at least two ways to identify these options and conclude that there are system states not identified by traditional methods that can inform new ways to shape flow networks for military operations.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

One of the most prevalent models in the study of military operations is the network flow problem. Network flow problems encompass a wide array of military applicability, from force projection and operational planning to supply chain logistics and resource scheduling. Traditional network interdiction techniques (i.e., attacker-defender models) have focused on identifying the set of nodes and/or arcs whose removal hurts to the performance of a system in the worst possible manner. The implementation of that worst-case scenario represents the overall success or failure of a strategy.

However, the military is responsible for not only the domination of the network, but also the repair, recovery, and transition of it back to civil authorities. The military develops and conducts operational planning to execute six joint operational phases – (I) shaping, (II) deterring, (III) seizing initiative, (IV) dominating, (V) stabilizing, and (VI) enabling civil authority. While Phases II and III focus on interdicting flow networks to break system components and minimize the performance of the system, Phases IV and V focus on the opposite: the restoration of the flow network back to a functioning state that maximizes performance on the system. Successful military operations depend on the "operational art" to effectively plan and execute interdiction activities and then transition to restoration activities that enable civil stability and governance.

Unfortunately, a lack of knowledge about the relationship between network interdiction and restoration currently hinders joint operations. Traditional interdiction and restoration methods identify the way to interdict a flow network that inflicts the most damage and the best possible way to restore it, but provide no decision-support for managing the transitions between interdiction to restoration activities. Instead, there may be other options than just the best- or worst-case to achieve a desired operational result. In particular, Phase I shaping activities would benefit from a greater number of interdiction and restoration options to determine the best course of action to execute military operations if and when plans need to change. Identifying whether there are flow network states that provide additional options not offered by interdiction and restoration models is critical to military operational art.

This thesis focuses on answering a single motivating question to inform network shap-

ing activities: ***Are there system states not normally identified via traditional [interdiction/restoration] methods that are operationally relevant and inform new approaches to joint operations?***

Our approach to answer this question starts with creating and analyzing a *metagraph* for a given flow network. A metagraph is a network representation of system states, where metagraph vertices encode which arcs are functioning or failed in a flow network and edges represent the interdiction and restoration activities relating one state to another (e.g., by failure or repair of an arc). Constructing a metagraph allows us to define a performance threshold, τ, for the flow network and color every vertex in the metagraph as either Green (i.e., meeting the performance threshold) or Red (i.e., not meeting the performance threshold). Using a colored metagraph, we are able to show how any interdiction or restoration may transition a flow network into a failed or functioning state. We demonstrate the creation of a metagraph by enumerating every state in a four-edge and nine-edge flow network.

We define two new measures to analyze a colored metagraph and identify system states relevant for military operations. The first is performance distance, δ, which measures the difference between flow network performance and the performance threshold τ. The second is Threshold Hamming Distance, θ, which measures how many interdictions or restorations are needed for a given flow network to cross the performance boundary. We use these two measures to identify two system states and their corresponding interdiction and restoration sets that would otherwise not be identified by traditional methods. The first type of states are those which only one of their interdiction-related or restoration-related edges cross the performance boundary. These states help identify which arcs in the original flow network are the most critical to ensure the system retains or loses nominal performance. The second kind of states are those which all of the interdiction-related edges (for Green) or all of the restoration-related edges (for Red) cross the performance boundary. These states help identify groups of nodes and arcs in the original flow network that are most important for quick system failure or recovery, respectively.

We demonstrate the importance of metagraph analysis using δ and θ by identifying critical system states for a notional 18-Edge Infrastructure Flow Network. Traditionally, a military analyst might use interdiction methods to find critical sets of arcs that, if interdicted simultaneously, would produce worst-case failure for the 18-Edge Flow Network. Using the

methods developed herein, we are able to expand upon previous analysis to provide more depth of knowledge about how to interdict and restore network performance. Specifically, by identifying Green and Red states that share a single transition edge across the performance boundary, we are able to pinpoint which arc in the 18-Edge Flow Network is most important for interdiction. Second, by identifying Red states where any restoration allows the flow network to regain nominal performance, we identify clusters of nodes and arcs that are most relevant for network recovery. Together, our metagraph analysis pinpoints new ways to interdict and restore network performance not previously discussed in the literature, and provides new options for military operations that seek to produce a particular operational effect.

Overall, the methods and analysis presented herein provide an answer to our guiding question: ***Yes, there are system states not normally identified via traditional [interdiction/restoration] methods that are operationally relevant to joint operations.*** These states include, but are not limited to, those identified using δ and θ with respect to a performance boundary τ. The new methods and the identification of alternative network states for the 18-Edge Infrastructure Flow Network provide a basis for redefining network shaping in military operational planning. Our conclusions suggest that traditional interdiction and restoration methods represent only a subset of network shaping activities. We more broadly define network shaping as real operational activities to interdict and restore a flow network supported by understanding how to traverse through the metagraph via combinations of interdictions and restorations. This broader definition implies that there may be a significant number of ways to shape networks that remain undiscovered.

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgments

A profound thank you to Dr. Alderson and Dr. Eisenberg for their patience, expertise, and guidance.

Thank you to my wife, JoAnna. Nothing I accomplished here would be possible without her love and support. Thank you Audrey, Ward, and Henry for reminding me of what is truly important.

Thank you to all the friends I made here. Any success I had as student is due in large part to Rory Page, Jared Kassulke, Aaron Devig, and James Streams.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1:
Introduction

Mathematical models are fundamental in the study of military operations research. As discussed by Dantzig (1963) in his seminal work, *Linear Programming and Extensions*, military operations became too complicated for a single commander to effectively plan and account for every variable. Algorithms that could quickly solve these types of problems became necessary for leaders and commanders to fight and win the nation's wars. Commanders required solutions that showed them the best way to complete an objective and the way to most impede the enemy.

1.1 Network Flows in Military Applications

One of the most prevalent models in the study of military operations is the *network flow problem*. Network flow problems encompass a wide array of military applicability, from force projection and operational planning to supply chain logistics and resource scheduling. In general, Ahuja et al. (1993) break network flow problems into three categories: *shortest path problems* that focus on identifying particular paths through flow networks, *maximum flow problems* that focus on maximizing a desired kind of flow over a network from sources of production to demand (e.g., fuel access), and *minimum cost problems* that focus on minimizing the difficulty of getting resources from points of production to demand (e.g., electric power flow). Each of these problems apply to a diversity of military operations, such as finding the fastest way to reach an objective, the best application of combat power, or the most efficient way to distribute supplies.

Two important military applications arose from the study of network flow problems: attacking networks and defending networks. These network flow applications develop when combined with the basic nature of military operations, specifically offense and defense. Attacking a network seeks to "deliberately damage the system" (Alderson et al. 2015) whereas defending a network protects the system. The military uses both of these applications to decide how and where to strike but also how and where to protect its own assets. But the use of only an attacker or only a defender model neglects that planning is not conducted in a vacuum and, colloquially speaking, the enemy gets a vote.

1.1.1 Network Interdiction

The goal of attacking a network is often *network interdiction*, for example to minimize the performance of a network. It can be accomplished by the degradation or complete elimination of part of the network. Cochran et al. (2011) succinctly write:

> The mathematical study of interdiction has focused primarily on network interdiction, in which an enemy's activities are modeled using the constructs of network optimization (e.g., maximum flows, multicommodity flows, and shortest paths), and in which attacks target the network's components to disrupt the network's functionality.

The study of the effects of an attack lead to the discovery of the worst-case disruption (i.e., the attack that most negatively affects the network as a whole). The formulation of attacker models allows the user to "assess the extent to which system operation is resilient" and enables the identification of the events that "reduce the capacity of the system to the lowest possible point" (Alderson et al. 2014). From a military perspective, the worst-case disruption represents the best case for the attacker and the worst case for the defender.

Traditional network interdiction techniques (i.e., attacker-defender models) have focused on identifying the set of nodes and/or arcs whose removal hurts the performance of a system in the worst possible manner. The implementation of that worst-case scenario represents the overall success or failure of a strategy. However, the military is responsible for not only the domination of the network, but also the repair, recovery, and transition of it back to civil authorities.

1.1.2 Network Defense and Restoration

Network defense refers to the variety of measures taken to prevent the degradation or disruption of a flow network. The defense of a network sometimes revolves around the prevention of the attacks. However, the defender can also limit the damage to the network by fortifying (or "hardening") parts of the system so as to maximize its performance in spite of an attack. A defender model is the converse of an attacker model as it maximizes the flow through the system. The combination of the models is often known as an attacker-defender model (Brown et al. 2006).

Network restoration involves the repair of components that have stopped working or declined in performance. The goal of restoration is to return the system as a whole to the state of optimal performance, given problem-specific constraints. If network interdiction is about breaking system components to minimize the performance of the system, then network restoration is the opposite: deciding what components to restore to maximize performance on the system.

1.2 Attack and Restoration in Military Operations

Modern military doctrine describes a notional joint operation model comprised of six phases (Joint Chiefs of Staff 2017); see Figure 1.1.

Phase 0: Shaping Activities is used to "help identify, deter, counter, and/or mitigate competitor and adversary actions that challenge country and regional stability." This is the planning step for an attacker or defender.

Phase I: Deter "prevents an adversary's undesirable actions, because the adversary perceives an unacceptable risk or cost of acting." This is the second stage of a bi-level problem when the attacker or defender prepares for a response to the action the other has taken.

Phase II: Seize the Initiative is the "rapid application of joint combat power" and "may be required to delay, impede, or halt the enemy's initial aggression and to deny the enemy its initial objectives." During this phase the attacker "seeks to degrade enemy capabilities."

Phase III: Domination occurs when the attacker is successful at "overmatching enemy capabilities at critical times and places."

Phase IV: Stabilize is "characterized by a shift in focus from sustained combat operations to stability activities."

Phase V: Enable Civil Authority is "to help the civil authority regain its ability to govern and administer the services and other needs of the population."

What happens in one phase affects what happens in a later phase. As noted by Hart et al. (2014):

It is the operational art that links a Phase II decision to destroy a bridge so that the enemy cannot use it for a counter attack to a Phase IV decision to rebuild the bridge so that commerce can use it. When practiced well, operational art never results in someone coming up to a piece of destroyed infrastructure and saying, "Dang, I sure wish we hadn't done this!" or "Boy, was that an expensive decision for a marginal gain."

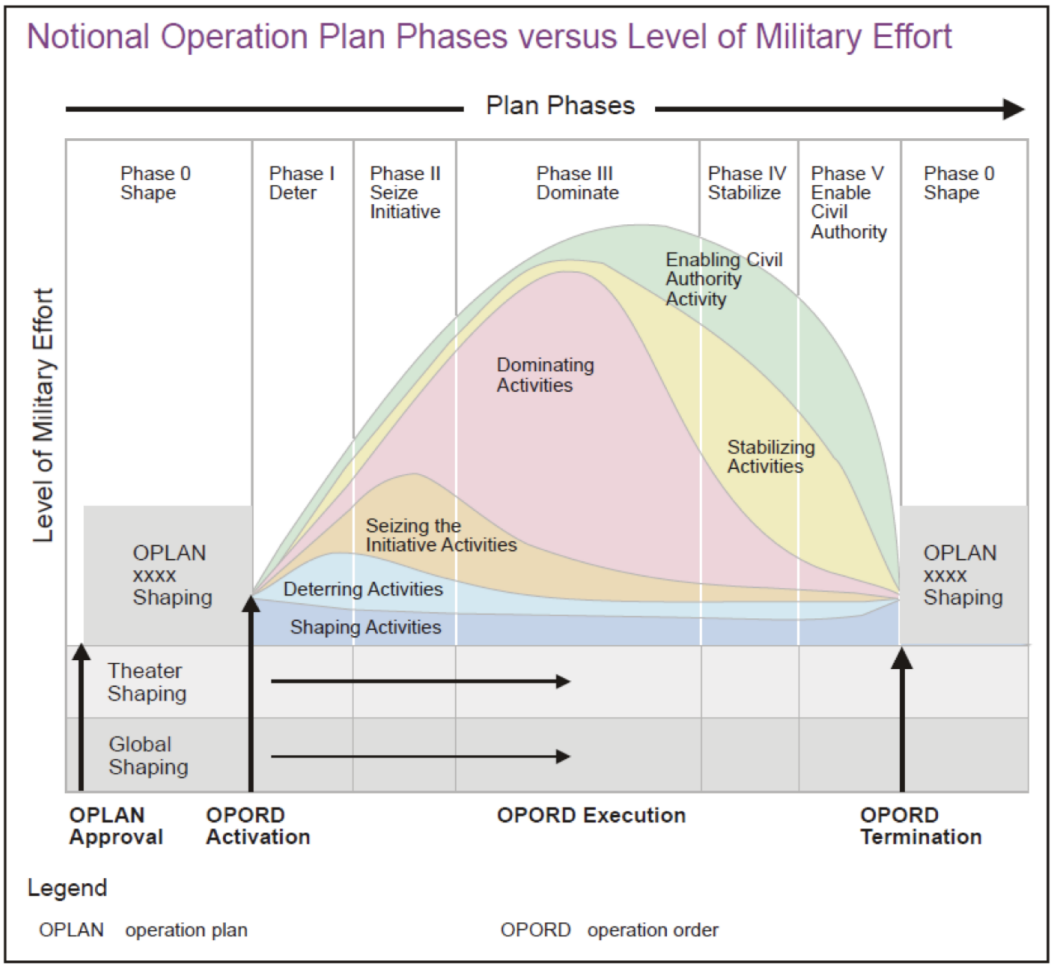

Figure 1.1. Notional Phases of a Joint Operational Plan (Hart et al. 2014). This figure "depicts six general groups of military activities that typically comprise a single joint combat operation. The model applies to large-scale combat operation as well as to a combat operation relatively limited in scope and duration. It shows that emphasis on activity types shifts as an operation progresses" (Joint Chiefs of Staff 2017).

Using the Notional Operation Plan Phases for reference, network interdiction begins during Phase 0: Shape. That is when the planning occurs for the operation. The transition to Phase I: Deter happens when movement begins to address the expected actions of the adversary. For example, you prepare for the enemy's most likely or most dangerous courses of action by rearranging your own forces. The attacks planned during Phase 0 are conducted in Phase II: Seize the Initiative.

Network restoration takes place during the later phases. Phase IV: Stabilize is the planning phase for restoration. It is the identification of the non-functional components of the networks and developing a strategy to return them to functional. Phase V: Enable Civil Authorities is the action phase for network restoration. The plans outlined in Phase IV are enacted and once functionality is restored to pre-determined threshold, control is returned to the local populace.

Network flow problems can be used to address any of the phases of the joint operation model, but they do a poor job with the transitions. Phases 0 and I represent the exhaustive enumeration of possible actions. Phases II and III directly refer to minimizing performance through worst-case targeting. During those phases, commanders seek to cause the greatest impact for minimal cost. Phases IV and V require commanders to recover the functionality lost by their own actions in the previous phases. This process of "break it bad" then "fix it fast" is inherently inefficient (Hart et al. 2014).

Previous research identifies methods to find the best- and worst-case scenarios for a given network, but does little to link these studies together into what Hart calls the, "operational art" (Hart et al. 2014). In other words, we understand how to interdict networks and how to restore them, but we do not know how interdiction activities affect restoration activities (or vice versa) and/or how we can make sense of these interactions in military planning, i.e., in Phase 0: Shaping. For example, previous research does not address whether it is necessary to maximize damage to a network when the ultimate result may be a need to restore its function. Moreover, there is no research on how the loss of specific nodes and/or arcs may push systems near, but not past, their critical threshold for operational failure. This lack of knowledge leaves planners with limited options and may be constraining shaping activities in unforeseen ways. Gaining new knowledge on the interactions between interdiction and restoration has the potential to expand the options for operational planners to develop

improved actions that “set conditions for successful theater operations" (Joint Chiefs of Staff 2017).

1.3 Thesis Goals

This thesis explores the concept of “network shaping", that is, identifying ways to [interdict/restore] network components to move a network [toward/away from] an operationally relevant performance threshold. This thesis focuses on identifying operational states associated with military network flow problems and how these states correspond to potential interdiction and restoration activities. A core question answered by this thesis is: ***Are there system states not normally identified via traditional [interdiction/restoration] methods that are operationally relevant and inform new approaches to joint operations?*** Answering this question will identify whether current research provides a comprehensive view of operational planning or if there are additional ways to “shape” networks not considered in the literature. Moreover, answering this question will “provide a deeper, and common, (sic) understanding" of military operations, which the Joint Chiefs of Staff (2017) states is the minimum expectation for the OPLAN Shape phase.

The primary contributions of this thesis are (i) formalization of the concept of “network shaping", (ii) developing corresponding mathematical models and solution techniques, (iii) demonstration of the technique and its benefits through illustrative examples.

CHAPTER 2:
Literature Review

2.1 Network Interdiction: An Abbreviated Review

Alderson et al. (2014) and Schrijver (2002) thoroughly detail the history of network interdiction problems beginning in 1954. Some of the most important early work was completed by Ford and Fulkerson (1954), Fulkerson and Dantzig (1954), and Dantzig and Fulkerson (1955) of the RAND Corporation, who defined the maximum-flow, minimum-cut theorem. Their work led to finding what was termed the *most-vital arc* that Wollmer (1963) called "the link, which if removed, would reduce the capacity of the network the most" (Wollmer 1963). Wollmer (1964) generalized his findings to identify the *n most-vital arcs*. He postulated

> given a maximum flow network from which *n* links are to be removed, which *n* arcs, if removed, would reduce the maximum flow from source to sink the most and what would be the maximum flow?

and discovered that the minimum cut is also the shortest path through the dual of the original maximum-flow problem (Alderson et al. 2013). Wollmer (1968) further contributed a stochastic nature to the maximum-flow interdiction problem.

McMasters and Mastin (1970) introduced a variation of the Wollmer (1963) algorithm that directly addressed the nature of limited resources available to combatant commanders by imposing a cost for the interdiction. Then, Ratliff et al. (1975) improved the efficiency of the Wollmer (1963) algorithm by identifying the cuts that damaged the system the most. That process"involves sequentially modifying the network so as to make this cut eventually become the cut with the smallest capacity." Ratliff et al. (1975) identified

> those *n* arcs whose simultaneous removal from the network causes the greatest decrease in the throughput capability of the remaining system between a specified pair of nodes.

Corley and Chang (1974) similarly found the method that identified most-vital nodes by altering the original Wollmer (1963) algorithm so that each node was represented by two different nodes, connected by a single arc, that acted as a node in the network and solved using the Ratliff et al. (1975) most-vital arcs algorithm.

2.1.1 Resurgence of Network Interdiction

Interest in network interdiction continued into the 1980s and 1990s as the methods were used to support United States' War on Drugs activities to interdict the illicit movement of cocaine from Latin America (Washburn and Wood 1995). Wood (1993) studied network interdiction to help quell "the flow of drugs and precursor chemical moving through river and road networks in South America" by reviewing "deterministic network interdiction models, devise new solution techniques for these models, and develop new models and solution techniques." Beginning with the the 'well-known max flow-min cut theorem' that he calls "the simplest network interdiction problem," Wood (1993) generalized the problem that "is shown to be easily modified and extended to handle variants." Washburn and Wood (1995) recognized that the essence of the "evader" and "interdictor" problem fit "into the form of a two-person zero-sum game (Washburn and Wood 1995). They realized that "drug smugglers must be considered to be intelligent adversaries who know or can learn about an interdictor's strategy" (Washburn and Wood 1995). Cormican et al. (1998) solved a stochastic version of the problem and Israeli and Wood (2002) developed a new method for "interdicting a transportation network in order to maximize the shortest path length between two specified nodes." Lim and Smith (2007) expands Wood (1993) "in the specific context of multicommodity flow networks."

2.1.2 Application to Critical Infrastructure

Network interdiction problems are now used to solve problems that arise within infrastructure systems. Brown et al. (2005) points out that "infrastructure that resists single points of random failure, or whose cutsets have low occurrence probabilities, may not survive a malicious, intelligent attack" and identifies a series of lessons learned:

- *The attacker has the advantage.*
- *Some systems are naturally robust, while others are not.*
- *Hardening an infrastructure system from attack can be expensive.*

- *The data are out there, and if we can get them, anybody can.*
- *The answers are not always obvious.*
- *Malicious, coordinated attacks can be much more damaging than random acts of nature.*
- *Reliability is not the answer.*
- *The right redundancy may be the answer.*
- *Secrecy and deception may be valuable.*
- *Worst-case analysis using optimization is key to a credible assessment of infrastructure vulnerability, and to reducing that vulnerability.*

Brown et al. (2006) adds the following lessons to the aforementioned:

- *High-fidelity models are achievable.*
- *Heuristics and rules of thumb are useful, but not for identifying vulnerability.*
- *An appropriate level of redundancy or reorganization could be inexpensive.*

These lessons have been applied to a variety of infrastructure systems. Salmerón et al. (2004) developed a model that identifies critical system components for power grids and Salmerón et al. (2009) solved a large, real-world version of the power grid problem. Church et al. (2004) applied the lessons to "identify the most critical facility assets in a service/supply system" and Church and Scaparra (2006) "identifies the most cost-effective way of allocating protective resources among the facilities of an existing but vulnerable system."

Snyder et al. (2006) addresses the idea of supply chain disruption and shows that "these systems can often be made substantially more reliable with only small additional investments in infrastructure." Murray et al. (2007) applies an optimization approach towards network interdiction of a telecommunications system. Alderson et al. (2011) uses a three-tiered approach (Defender-Attacker-Defender) to a transportation network.

2.2 A State-Space View of System Operation

Recent research taking a state-space view on how the availability of arcs in a flow network impact system performance suggest that interdiction studies may provide only narrow recommendations for military operational planning. In general, the operation of an infrastructure network can be represented both by its *performance* (e.g., objective value) and by

the *operational state of its components* to achieve a given performance. Each infrastructure system has many operational states, where each state represents a combination of available or interdicted nodes and/or edges. For example, when one assumes that interdiction only affects network edges and edges are either available or unavailable (i.e., take on a binary value), the infrastructure network operational state can be represented by "a binary vector of length k, where k is the number of [edges] in the infrastructure system" Schulze (2014). Here, the given infrastructure system posses 2^k possible operating states (Schulze 2014) representing all combinations of available and unavailable edges, which is a very large number for even small networks. Solving network interdiction problems like those described in Section 2.1 identifies particular operational states that correspond to the worst-case loss of system performance, and thus find a proverbial worst-case needle-in-the-haystack of system states. Despite the practical importance of identifying these system states for both offensive and defensive military actions, optimization-based techniques provide little information about the many other system states that may achieve similar or identical operational results. Instead, taking a "state-space view" of infrastructure operations requires forgoing optimization-based methods for those that focus on directly on analyzing system states and their relationships.

Schulze (2014) examined the differences in results for restoration of a damaged infrastructure system using both a mixed-integer linear program (MILP) and a graph-based approach for representing and analyzing the systems' operational state space. Schulze determined that both algorithms found optimal solutions to a deterministic model, but each had different shortcomings. Specifically, the MILP solved the problem much faster than the graph-based approach, but failed to respond to dynamic changes. The graph-based approach, while slower, was able to solve a problem given dynamic changes (Schulze 2014).

Brendecke (2016) advanced research on infrastructure operational state space by adding a stochastic element to the restoration model and including components in new, repaired, and broken states. The Brendecke model allows nodes/arcs to degrade and fail and then optimizes the system based upon the failure/degradation. Using Markovian principles, the network only exists in a single state at a given time and future states depend only upon the current state. Each state then can be classified as a success or failure, as can subsequent states. Brendecke demonstrated that considering the state of multiple arcs in a network flow system is a difficult task. Arcs can take on more than two states and those states have a

probabilistic failure rate. The use of non-binary edges makes the process very complicated and computationally difficult.

Clark (2017) built upon the work of Schulze (2014) by developing a model and algorithm that produced a parametric view of resilience Clark (2017) using graph-based approaches to studying an operational state space. This work linked network flow performance to operational state, where results present a range of outcomes for a given system based on the availability of network components. This 'state space view' permits decision makers to see a "range of possible attacks or failures" Clark (2017) that expands beyond the limited view provided by optimization-based methods. Clark also demonstrated the impact of redundancy on an infrastructure network by showing that systems with a high number of redundant paths require more attacks to cause system failure. Likewise, systems with "highly connected areas" require more attacks to cause failure.

The tradeoffs between optimization and state space methods revealed by Schulze (2014), Brendecke (2016), and Clark (2017) signify that there is an opportunity to develop novel network shaping techniques for military operations such as Augmented Target Systems Analysis. Schulze (2014) suggests that established interdiction techniques may be inappropriate for dynamic infrastructure environments, where a state-space view is more appropriate. Brendecke (2016) shows that optimal infrastructure repair and replace policies are computationally difficult to achieve and may not be assumed in practice. Clark (2017) shows that considering network performance and operational state together reveals a greater range of possible attacks and helps identify where infrastructure redundancies and clustering may impede military goals. In all cases, each study suggests that a state space view of infrastructure networks reveals new information that is otherwise lost when using established interdiction methods. Moreover, they reveal the potential to consider different kinds of shaping activities that remain otherwise unstudied, such as determining ways to interdict adversarial systems into operating states that have good network performance, but are vulnerable to few, targeted losses. Determining the breadth of possible network shaping techniques will improve our understanding of Network Interactions, Vulnerabilities, Robustness & Resilience, particularly as they pertain to critical infrastructure systems and campaign analysis.

2.3 Contributions of this Thesis

Previously studied work focused on identifying worst-case interdiction or fastest possible restoration, but these may be narrow network shaping goals for military operations. Complete interdiction, the best-case scenario for an attacker, creates issues in restoration activities required for Phase IV and V military operations and may be detrimental to the success of military operational plans. The current literature does not account for this relationship between interdicting and restoring a system, more specifically it is missing the relationship between the objective of operator models and their operational effects across an OPLAN. We seek to find new ways to consider interdiction or restoration activities that suit a broader range of military operations that require both interdiction and restoration. Furthermore, we seek to characterize the breadth of operational effects that can be studied with network flow problems to understand the full spectrum of possible military operations.

This thesis tackles these goals by linking network performance with an operational state space view of component availability. Specifically, we identify how different state transitions via network interdiction and network restoration relate to different network shaping activities and goals. We demonstrate that network shaping activities must consider both how to traverse a state space (via interdiction and restoration) and what boundaries between functional performance states and non-functional performance states matter. Whereas past research implicitly assumed that worst-case interdiction or best-case restoration was the primary operational goal, we identify at least two types of system states that are important for military operations and would be overlooked by established interdiction and restoration methods. We conclude that the goal of network shaping activities, then, is to ensure that interdiction and restoration activities help transition infrastructure networks effectively and efficiently from undesired to desired operational states. Established network interdiction and restoration literature only considers narrow subset of possible network flow problems relevant to network shaping activities.

CHAPTER 3:
Model

In this thesis, we focus on the study of network flow problems relevant for the modeling and analysis of critical infrastructure systems, called *operator models*. The *operator* is the one who makes decisions about how to manage flows (Alderson et al. 2015). The operator must route commodities through the system while taking into account the constraints within the network. Thus the operator model is the constrained optimization problem for a specific network flow model.

3.1 Flow Network

As a starting point, we consider the Operator Model from Alderson et al. (2015) to solve a minimum cost network flow problem that is formulated as follows.

Indices and Sets

$n \in N$	nodes (alias *i,j*)
$[i, j] \in E$	undirected edge between nodes i and j where $i < j$
$(i, j) \in A$	directed arc from i to node j; $[i, j] \in E \leftrightarrow (i < j) \wedge ((i, j) \in A \wedge (j, i) \in A)$

Data [*units*]

c_{ij}	per unit cost of traversing edge $[i, j] \in E$ [cost-units]
u_{ij}	upper bound on total (undirected) flow on edge $[i, j] \in E$ [cost-units]
$\hat{x}_{ij}$	1 if edge $[i, j] \in E$ is damaged, 0 otherwise [binary]
q_{ij}	per unit penalty cost of traversing arc $(i, j) \in A$ if damaged [cost-units]
d_n	supply at node $n \in N$ (-demand for $d_n < 0$) [flow-units]
p_n	per unit penalty cost for demand shortfall $n \in N$ [cost-units]

Decision variables [*units*]

Y_{ij}	directional flow on arc $(i, j) \in A$ [flow-units]
S_n	unit shortfall at node $n \in N$ [flow-units]

Formulation

$$\min_{Y,S} \sum_{[i,j]\in E} \left[\left(c_{ij} + q_{ij}\hat{x}_{ij} \right) Y_{ij} + \left(c_{ji} + q_{ji}\hat{x}_{ji} \right) Y_{ji} \right] + \sum_{n\in N} p_n S_n \tag{3.1}$$

$$\text{s.t.} \sum_{(n,j)\in A} Y_{nj} - \sum_{(i,n)\in A} Y_{in} - S_n \leq d_n \qquad \forall\, n \in N \tag{3.2}$$

$$0 \leq Y_{ij} + Y_{ji} \leq u_{ij} \qquad \forall\, [i,j] \in E \tag{3.3}$$

$$S_n \geq 0 \qquad \forall\, n \in N \tag{3.4}$$

Discussion

The objective function (3.1) represents the total cost of sending flow over the network plus any penalties for demand shortfall. Constraint (3.2) is a balance-of-flow constraint, and constraint (3.3) defines the upper bound for flow on each edge. Stipulation (3.4) provides non-negativity for the shortfall variables.

3.2 A Four-Edge Flow Network

Consider the simple four-edge network in Figure 3.1, with behavior governed by the minimum cost network flow problem from Section 3.1. As supply enters the network, it flows through $k = 4$ undirected edges based upon the demands that are present and in a manner that minimizes the total system cost. We assume that each edge has a per-unit traversal cost $c_{ij} = 1$, an upper bound on undirected flow $u_{ij} = 15$, and per-unit penalty cost $q_{ij} = 10$.

The cost of sending flow over an edge $[i, j]$ is dictated by the value $\hat{x}_{i,j}$, which represents whether the edge is damaged ($\hat{x}_{ij} = 1$) or undamaged ($\hat{x}_{ij} = 0$). We say that the *state* of the network as a whole is given by the sequence of individual $\hat{x}_{ij}$ values for all the edges in the network. We refer to this sequences of values as a *bitstring* of length k. For example, the bitstring '0000' refers to the state where all four edges are undamaged, whereas the bitstring '1111' refers to the state where all four edges are damaged.

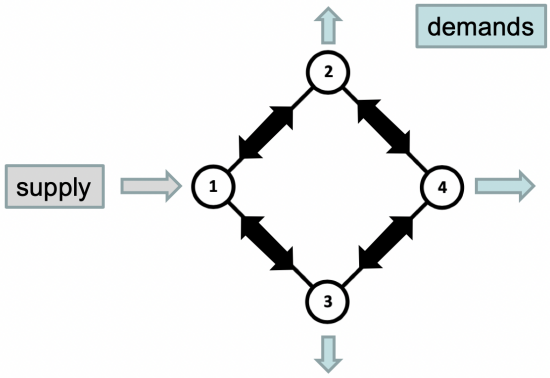

Figure 3.1. Four-Edge Flow Network. This figure shows the basic functionality of our flow network. There are four nodes connected by four edges. Supply enters at Node 1 and Nodes 2, 3, and 4 have demands that must be met.

When an edge in the network becomes damaged (i.e., interdicted) the state in the network changes. We use the bitstring to record changes to the system state. If, for example, starting with all edges unbroken, the second edge, [1,3] is interdicted, the system state changes from '0000' to '0100'. Equivalently, we say that *the system has transitioned* from state '0000' to state '0100'. Similarly, if we repair the interdicted edge, then the system transitions from '0100' back to '0000'. We assume damage or repair happens to only one edge at a time, so each transition involves a change in only a single element of the bitstring.

Given that each edge is either broken or unbroken there are 2^k different system states for a network. There are $2^4 = 16$ total system states for the four-edge network. We refer to the collection of possible state values as the *state space*, denoted S, for the system. Thus, in general $|S| = 2^k$. We index individual states as $s \in S$.

Let $p(s)$ denote the *performance* of the system in state s, assumed to be the minimum operating cost given the state of individual network edges. Solving the operator model (3.1)-(3.4) for the system in Figure 3.1 using the parameters in Table 3.1 and Table 3.2 for each possible state yields the performance values in Table 3.3.

Node (n)	Supply (d_n)	Penalty (p_n)
1	3	10
2	-1	10
3	-1	10
4	-1	10

Table 3.1. Node Data for the Four-Edge Flow Network. This table identifies the parameters for the nodes in our Four-Edge Flow Network. The origin node, node 1, has the initial supply which must flow through the network to meet the demand of the other nodes which each require 1 unit of supply. For each node, there is a penalty of 10 if the demand is unmet.

Edge $[i, j]$	Cost (c_{ij})	Capacity (u_{ij})	Penalty (q_{ij})
$[1, 2]$	1	15	10
$[1, 3]$	1	15	10
$[2, 4]$	1	15	10
$[3, 4]$	1	15	10

Table 3.2. Edge Data for the Four-Edge Flow Network. This table identifies the edge parameters for our Four-Edge Flow Network. Every edge in this network has a per-unit flow cost of 1 and has a capacity of 15.

State	$p(s)$
0000	4
0001	4
0010	4
0011	12
0100	6
0101	13
0110	21
0111	21
1000	6
1001	21
1010	13
1011	21
1100	30
1101	30
1110	30
1111	30

Table 3.3. Performance of Four-Edge Flow Network for Each State. This table shows the performance values for all 2^k possible states in the system. The best case (lowest cost) performance value, 4, reflects the cost for all four nodes to meet demand requirements. The worst-case (largest cost) performance value, 30, occurs whenever both the edges out of node 1 are blocked resulting in unmet demand (and a penalty of 10) for the other three nodes.

3.3 Metagraph of System States

The set of system states and possible transitions between them can be represented as a *metagraph* of vertices (states) and edges (transitions between states). Figure 3.2 is a visual representation of the metgraph for the four-edge system, showing all $2^4 = 16$ possible states for the flow network in Figure 3.1. The vertices in the metagraph are the individual system states $s \in S$, and the edges are the transitions between them. Transitions from 0s to 1s occur when edges are interdicted (resulting in movement from top to bottom of the metagraph in Figure 3.2). Transitions from 1s to 0s occur when edges are restored (resulting in movement from bottom to top of the metagraph in Figure 3.2).

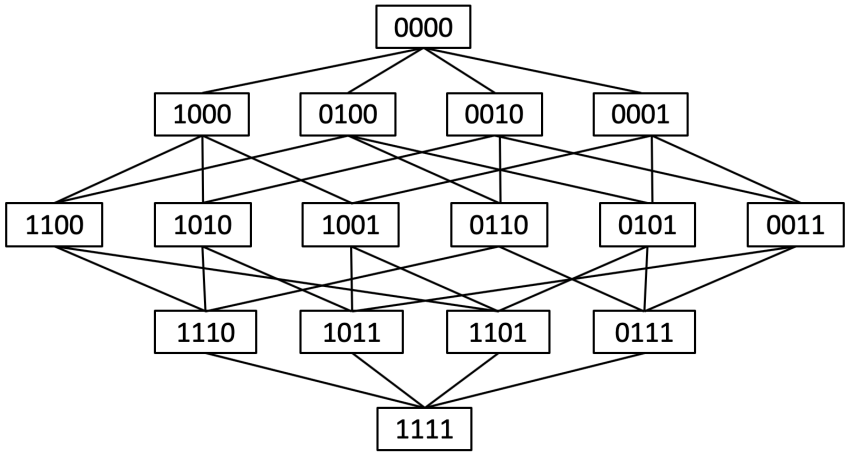

Figure 3.2. Metagraph for Four-Edge Flow Network. This figure shows every possible state in which a four-edge flow network could exist. Each row, or level, corresponds to the number of interdicted edges, zero interdictions at the top and four at the bottom.

The values in Table 3.3 allow us to associate each state in the metagraph with a corresponding performance of the underlying flow network. In general, we assume that the performance of some states is acceptable, while the performance of other states is unacceptable. For example, for a minimum cost flow problem, we might consider a cost above a particular *threshold* to be too high.

Let τ represent the threshold parameter used to categorize states based on their performance. Since we are minimizing cost, we consider a vertex "Green" if $p(s) \leq \tau$ and color it "Red" if $p(s) > \tau$ (see Figure 3.3). Operationally, we can think of each Green vertex as a state for which the underlying flow network achieves *mission success*, whereas each Red vertex

represents a state for which the underlying flow network has *mission failure*. In this simple example of a four-edge flow network, we can establish a clear line separating SUCCESS and FAILURE; we refer to this line as the *boundary* (see Figure 3.3).

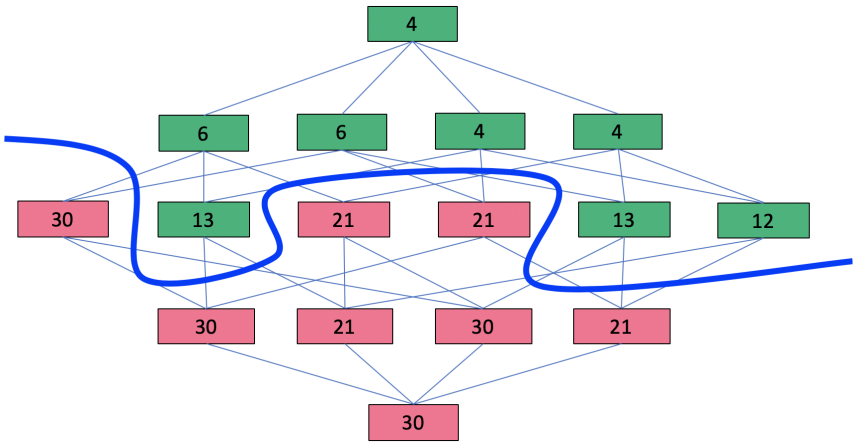

Figure 3.3. Colored Metagraph with Performance Boundary. We label each vertex in the metagraph with its performance from from Table 3.3 and color it accordingly. Green vertices have performance values less than the threshold, τ=20, and Red vertices have performance values that are higher. The blue line in this figure shows τ=20, the boundary between success and failure.

3.4 Distance to the Boundary

The inclusion of a performance threshold creates a strict boundary between operating states that achieve mission success and those that suffer mission failure. A natural question becomes: how "close" to the boundary is a given vertex in the metagraph? For a Green state, how close is it to becoming Red? For a Red state, how close is it to becoming Green?

We use two notions of distance to characterize how "close" a metagraph state is to transitioning across the performance boundary. The first is is based on the difference in performance of their states, and the second is based on the number of transitions that separate them in the metagraph.

3.4.1 Performance Distance

We use the term *performance distance* to represent the difference in overall performance that a given vertex is from the threshold. Specifically, for each state s we define $\delta(s) = |p(s) - \tau|$.

This provides one measure how "close" the vertex corresponding to state s is to switching from success to failure or vice versa. For a performance threshold $\tau = 20$, Table 3.4 lists the performance distance $\delta(s)$ for each state $s \in S$.

Figure 3.4 shows the metagraph with each vertex labelled with its corresponding performance distance. We observe that different vertices in the metagraph that are adjacent to the performance boundary can have considerably different $\delta(s)$ values. This suggests the need for an additional measure of distance.

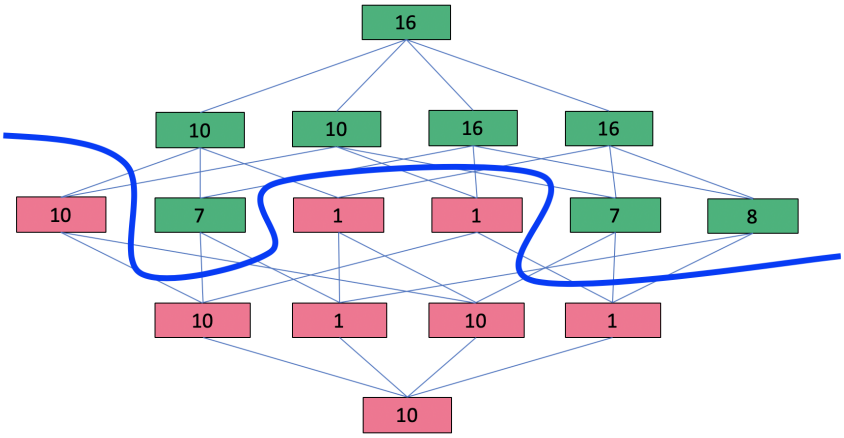

Figure 3.4. Colored Metagraph with Performance Distance. This figure replaces the performance values with performance distances, δ, for each vertex in the system. The performance distance is the difference between the performance value and the boundary threshold, τ=20 in this case.

3.4.2 Threshold Hamming Distance

Movement along the links of the metagraph correspond to transitions between states. The shortest path between two states in the metagraph is known as the *Hamming Distance* as first introduced by Hamming (1950). We denote the *Threshold Hamming Distance* as the minimum number of transitions from a given vertex (corresponding to interdictions or restorations in the underlying flow network) for the system state to transition across the boundary. A vertex in the metagraph adjacent to the boundary has a Threshold Hamming Distance of one, meaning that only one interdiction or restoration is necessary to cause the system state to transition from a Green state to/from a Red state. A vertex with a Threshold Hamming Distance of two requires two restorations or interdictions to transition across the

boundary. We use $\theta(s)$ to represent the Hamming distance to the boundary for state s. Table 3.4 lists the $\theta(s)$ values for the states in the metagraph corresponding to the four-edge flow system.

State	$p(s)$	$\delta(s)$	$\theta(s)$
0000	4	16	2
0001	4	16	1
0010	4	16	1
0011	12	8	1
0100	6	14	1
0101	13	7	1
0110	21	1	1
0111	21	1	1
1000	6	14	1
1001	21	1	1
1010	13	7	1
1011	21	1	1
1100	30	10	1
1101	30	10	1
1110	30	10	1
1111	30	10	2

Table 3.4. Performance Value, Performance Distance, and Threshold Hamming Distance for each Vertex in the metagraph of the Four-Edge Flow Network. Given a performance threshold $\tau = 20$, the performance distance $\delta(s) = |\tau - p(s)|$ does not tell a complete story for how "close" a vertex is to crossing the threshold. For example, both states '0000' and '0001' have the same performance value and performance distance but a different Threshold Hamming Distance $\theta(s)$, which implies that they are not equivalent states. State '0001' is one transition closer to crossing the boundary.

The metagraph in Figure 3.5 illustrates the Threshold Hamming Distance for each vertex. We see that all but two vertices, '0000' (all links restored/available) and '1111' (all links interdicted), are just one transition from crossing the threshold. These vertices have Threshold Hamming Distances of two as they are two transitions from crossing the boundary,τ=20.

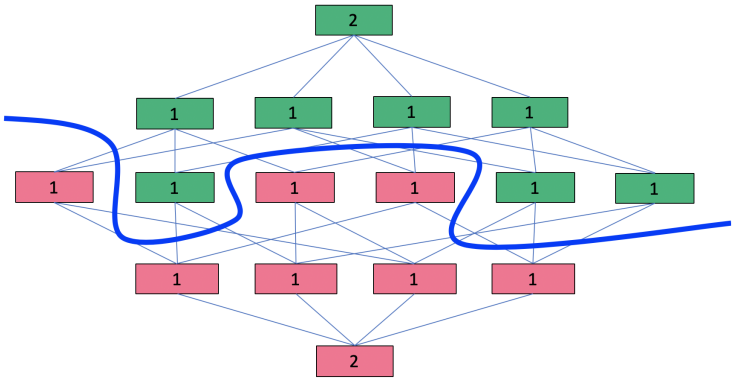

Figure 3.5. Colored Metagraph with Hamming Distance to Boundary. This figure shows the Hamming Distance for each vertex in the system. All nodes except '0000' and '1111' are adjacent to the boundary, τ=20, and have Hamming Distances of 1, which means that they are all one transition away from crossing the boundary.

3.5 A Nine-Edge Flow Network

Consider the slightly larger, nine-edge flow network in Figure 3.6, with node data defined in Table 3.5. We assume that each edge has a per-unit traversal cost $c_{ij} = 1$, an upper bound on undirected flow $u_{ij} = 15$, and per-unit penalty cost $q_{ij} = 10$.

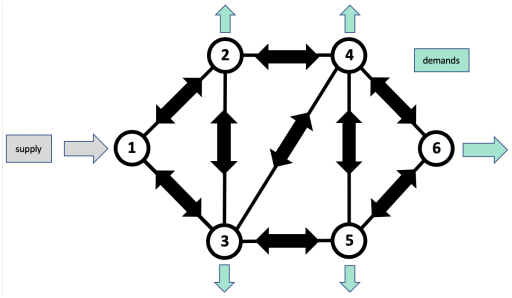

Figure 3.6. Nine-Edge Flow Network. This network has six nodes and nine edges. Like the smaller flow network in Figure 3.1, supply enters at Node 1 and Nodes 2, 3, 4, 5, and 6 have demands that must be met.

This system has a total of $2^9 = 512$ states. For each state, we run the Operator Model from Alderson et al. (2014) to solve the minimum cost network flow problem. The resulting performance values range from 9 (the optimal min-cost performance) to 50 (worst-case performance). We assume $\tau = 22.5$, which is the median performance value, represents a performance threshold that separates mission success from mission failure.

Node (n)	Supply (d_n)	Penalty (p_n)
1	5	10
2	-1	10
3	-1	10
4	-1	10
5	-1	10
6	-1	10

Table 3.5. Nine-Edge Flow Network Node List. These are the node parameters used to evaluate the Metagraph for the Nine-Edge Flow Network. The origination node, node, has the initial supply which must flow through the network to meet the demand of the other nodes which all require 1 unit of supply. For each node, there is a penalty of 10 if the demand is unmet.

We build the metagraph and calculate both the Performance Distance and the Threshold Hamming Distance for each state. With $\tau = 22.5$, we observe $\delta(s) \in [1.5, 27.5]$ for $s \in S$. However, we observe the Hamming Distances, $\theta(s) \in [1, 3]$, meaning that every vertex is within three transitions of the boundary. Figure 3.7 illustrates the metagraph for our nine-edge flow system using the application *gephi* (Bastian et al. 2018). We color each vertex green or red as appropriate, and we manipulate the size of each vertex to correspond with its theta value. Larger vertices correspond to larger θ values.

The nine-edge flow network is not particularly large, but its metagraph is too complicated to visually identify important structural features. The histogram in Figure 3.8 provides a different view of key features.

The bottom row of Figure 3.8 shows the distribution of performance values $p(s)$ for all $s \in S$. Each of the plots above it shows the distribution of $p(s)$ partitioned for the values $\theta(s) = 1, 2, 3$. The blue dotted line is the boundary threshold, $\tau = 22.5$, and we therefore color the different halves of each distribution green or red, accordingly.

Figure 3.8 shows that although there are vertices in the metagraph that have both relatively large δ and large θ, as well as those that have both relatively small δ and large θ, these two measures need not be correlated. That is, there exist vertices in the metagraph that have relatively large δ but relatively small θ. This means that there are some states that appear "far from the boundary" in terms of their performance (i.e., they would need their

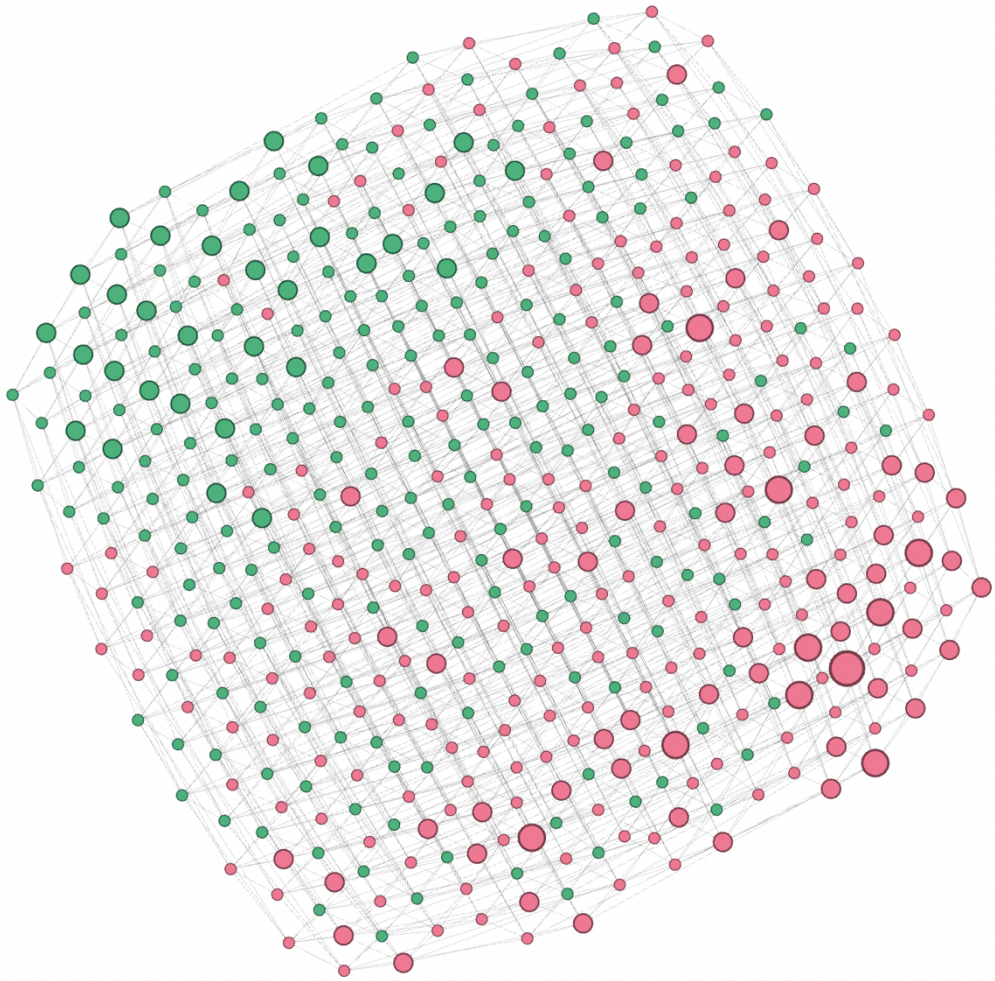

Figure 3.7. Metagraph for Nine-Edge Flow Network. Vertex color indicates which states of the 512 states are above or below the performance threshold. Vertex size shows Threshold Hamming Distance (larger circles are farther from the boundary). In general, visual inspection is insufficient for any flow network of considerable size.

performance to change considerably before they crossed the threshold), but in fact require only a small number of transitions to cross the threshold (and are therefore quite "close" based on Hamming distance). For example, there exist vertices in the metagraph that have performance $p(s) = 9$ (which is the lowest possible operating cost) with corresponding performance distance $\delta(s) = 13.5$ (the largest possible margin for a Green state), but also have $\theta(s) = 1$, meaning that they are adjacent to the boundary and require only a single

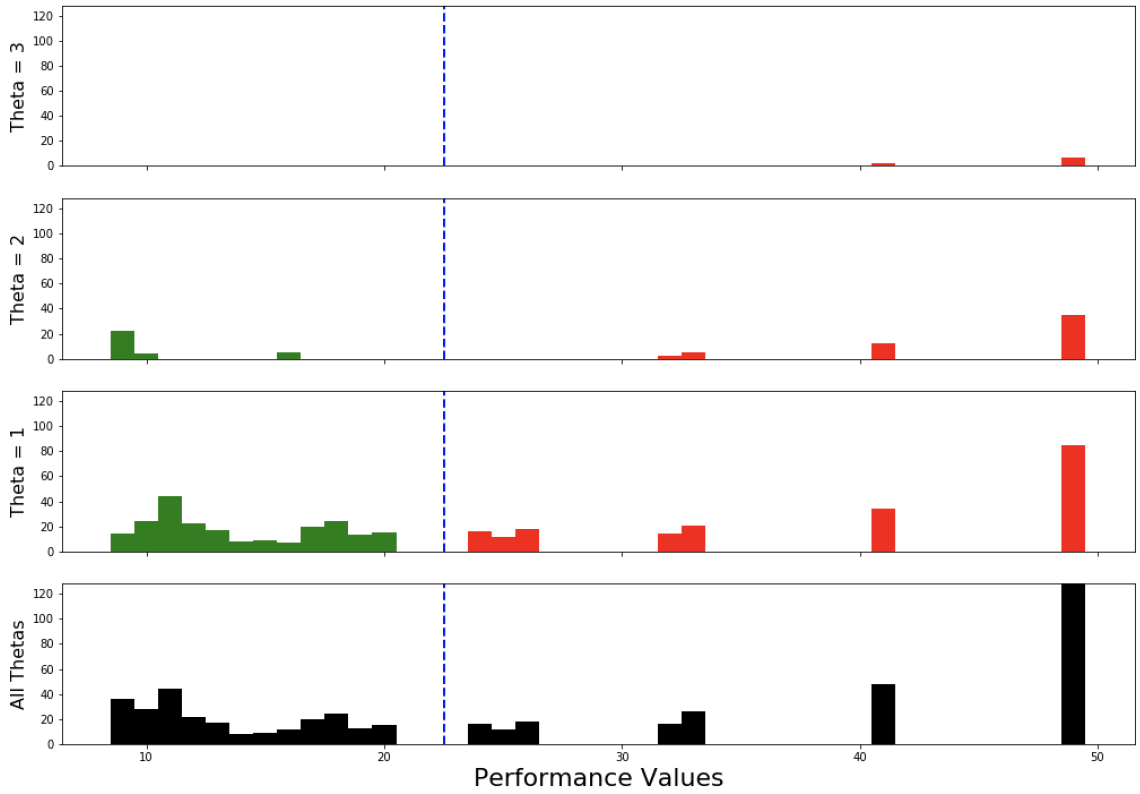

Figure 3.8. Histograms of Metagraph Vertices by Threshold Hamming Distance ($\tau = 22.5$). This collection of histograms shows the total counts of vertices in the metagraph with a given performance value (bottom row), as well as separating them by Threshold Hamming Distances $\theta = 1$, 2, or 3 (top three rows, as indicated). The histogram labeled $\theta = 1$ shows the distribution of vertices by performance value for all vertices that are only one transition from crossing the performance boundary. The histograms labeled $\theta = 2$ and $\theta = 3$ show the distribution of vertices by performance value that are two or three transitions away from crossing the performance boundary, respectively.

transition to change from Green (success) to Red (failed).

The existence of such vertices in the metagraph confirms that (1) this view of the state space can yield novel insights into how we think of individual system states as being desirable or not, and (2) analysis of the metagraph can potentially reveal new ways to perform shaping activities in network flow systems.

3.5.1 Analysis of the Metagraph to Identify Important System States

Our analysis of the metagraph begins with an investigation of individual vertices. Each vertex in the metagraph is connected to k other vertices, these connections represent the possible interdictions and/or restorations for a given system state. We call the number of interdictions (i.e., number of failed edges) at a given vertex the *level* (denoted λ) of the vertex. For example, vertex '011000000' has $\lambda = 2$. Of the k possible transitions out of each vertex, there are λ that correspond to a restoration and $k - \lambda$ that correspond to an interdiction. In general, there are $k + 1$ total levels in a given metagraph, and the number of vertices per level is $\binom{k}{\lambda}$. Figure 3.9 shows the distribution of states by level, with each colored according to whether they are above or below the threshold ($\tau = 22.5$).

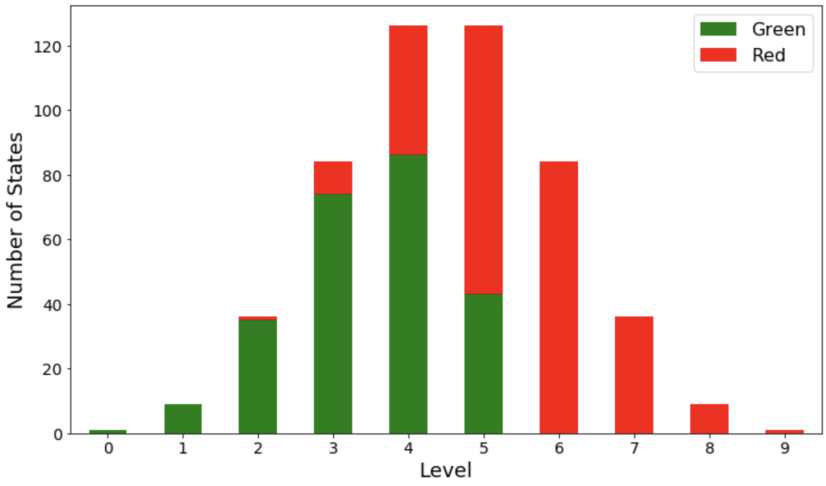

Figure 3.9. Histogram of States in Metagraph for Nine-Edge Flow Network, by Level and Color. The $2^k = 512$ states of the metagraph are distributed across $(k + 1)$ levels, with the height of each color representing the relative number of states that are above or below the threshold ($\tau = 22.5$). This figure shows the symmetrical nature of a metagraph.

To identify important vertices for intediction and restoration activities, we are interested in the relationship between system performance and the transition edges of each vertex.

3.5.2 State Transitions Across the Boundary

Vertices in the metagraph that are adjacent to the boundary are of particular interest because they represent states for which action on a single edge (interdiction for a Green vertex or restoration for a Red vertex) can cause the system to transition across the performance

threshold. Although these boundary-adjacent states are easily identified by their Threshold Hamming Distance, $\theta = 1$, this distance metric does not indicate how many edges represent potential transitions across the boundary. Although some vertices have only a single way to transition across the performance boundary, many have multiple ways of doing so.

Figure 3.10. Histogram of Transition Edges that Cross the Boundary ($\tau = 22.5$) for Metagraph Vertices with $\theta(s) = 1$. Of the 512 metagraph states for the Nine-Edge Flow Network, only 417 have $\theta = 1$, (i.e., are adjacent to the performance boundary). Each histogram shows the number of potential transitions that cross the performance boundary for Green vertices (217 states) and Red vertices (200 states). Although some vertices have only a single way to transition across the performance boundary, many have multiple ways of doing so.

In the metagraph for the Nine-Edge Flow Network, 417 of 512 states are adjacent to the boundary. Of these, there are 217 Green vertices and 200 Red vertices. Figure 3.10 shows the counts according to the number of transition edges that each has crossing the boundary. We observe that although some vertices only have a single potential transition across the boundary, many vertices have multiple edges that cross the boundary. Both are important from a network shaping perspective.

Vertices With a Single Transition Edge Across the Boundary

A Green vertex in the metagraph with only a single edge that crosses the boundary is important from a network shaping perspective because it is boundary-adjacent (and therefore

vulnerable to targeted interdiction) while having the fewest possible ways to transition across the boundary (and is therefore relatively robust to non-targeted interdiction). Preventing the system from transitioning across this edge will keep the system in a Green state. In the metagraph for the Nine-Edge Flow Network, there are 62 Green vertices with only a single edge that crosses the boundary.

Conversely, a Red vertex in the metagraph with only a single edge that crosses the boundary is important from a network shaping perspective, because it requires targeted restoration to improve system performance to reach mission success (a single, non-targeted restoration is unlikely to suffice). Preventing the system from transitioning across this edge will keep the system in a Red state. In the metagraph for the Nine-Edge Flow Network, there are 26 Red vertices with only a single edge that crosses the boundary.

Interestingly, there are only two places in this particular metagraph where vertices on opposite sides of the performance boundary share their only connection across the boundary. Figure 3.11 illustrates one of these. The Green vertex with state '101100000' (performance $p(s)$ =18.0) has only a single transition across the boundary to Red vertex with state '111100000' (performance $p(s)$ =50.0), and vice versa. Denying an adversary the ability to 'move' along edge [101100000, 111100000] means that the path across the performance boundary will require at least two transitions instead of one. Deliberate actions that 'move' the system state in the metagraph (or deny the adversary the ability to do so) demonstrate an important idea for network shaping not discussed in the interdiction and restoration literature.

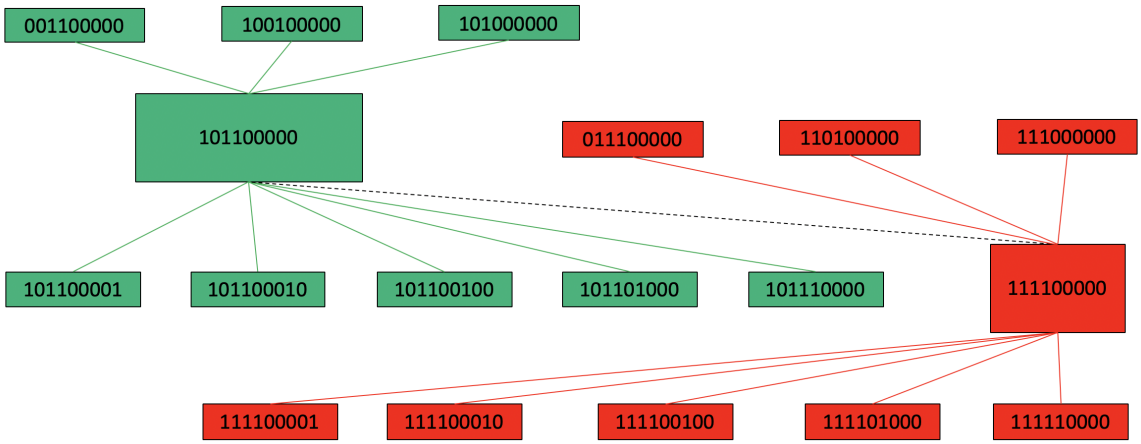

Figure 3.11. Connected Vertex Pair with Only One Transition Across the Boundary. These vertices have just one transition that crosses the threshold AND that connection also has only one transition across the boundary. Preventing 'movement' along edge [101100000, 111100000] means that the path across the performance boundary will require at least two transitions instead of one.

Vertices With Multiple Transition Edges Across the Boundary

Metagraph states that have many transition edges that cross the performance boundary are also important to identify for network interdiction and restoration because there are multiple ways in which a single action (interdiction for a Green vertex or restoration for a Red vertex) can cause the system to transition across the performance threshold. However, of special interest are vertices for which *all* of the interdiction-related edges (for Green) or *all* of the restoration-related edges (for Red) cross the performance boundary. For a vertex at level λ, if all $k - \lambda$ interdiction-related edges cross the performance boundary, then *any single interdiction will cause the system to transition from Green to Red*. Conversely for a vertex at level λ, if all of the restoration-related edges cross the performance boundary, then *any single restoration will cause the system to transition from Red to Green*.

Figure 3.12 illustrates the case of vertex '011011011' whose performance $p(s) = 26 > \tau$ and therefore represents mission failure. Figure 3.12 also shows local connectivity from this vertex in the metagraph, where the optimal solution for the original Nine-Edge Flow Network is also shown for each state. We observe that *restoring any of the broken edges from state '011011011' improves the performance of the system sufficient for the system*

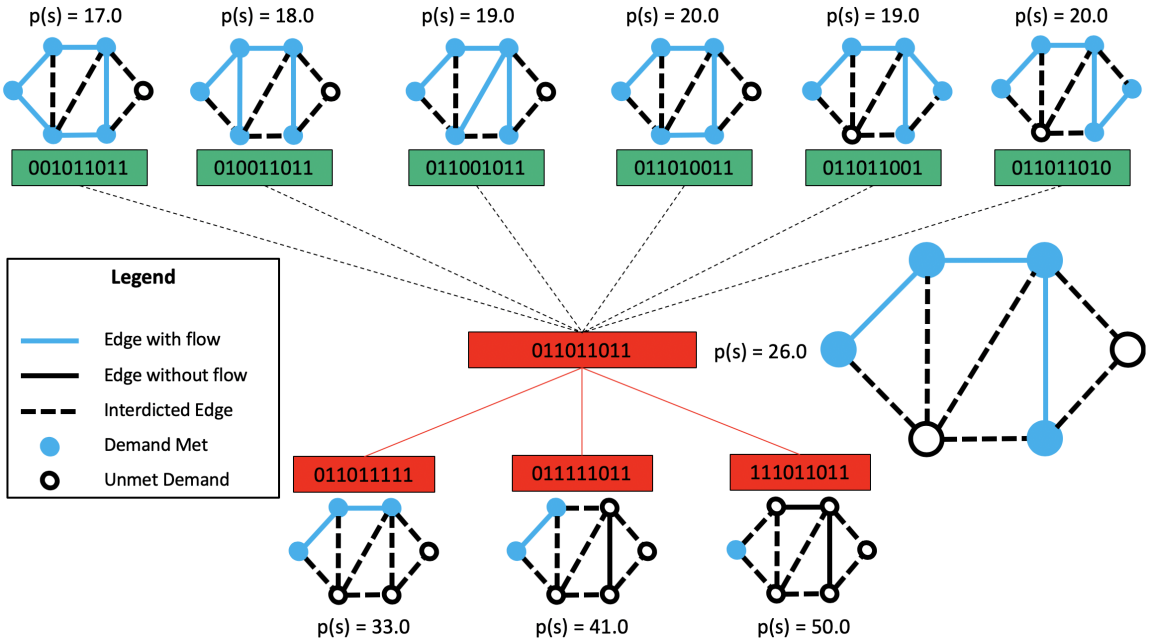

Figure 3.12. Vertex '011011011' with all Transitions. Each state is shown with its metagraph representation. Blue lines show the flow through the network amd blue filled nodes have their demand met. Dashed black lines show interdicted edges and solid black lines show edges that are intact but cannot be reached and do not have flow traversing the edge. Empty black circles show nodes that have unmet demand.

to transition from mission failure to mission success. Thus, although vertex '011011011' corresponds to mission failure, its required performance is easily restored. Thus, this state might be an attractive goal for an operation that requires to "break it bad" but later must "fix it fast."

In the metagraph for the Nine-Edge Flow Network, there are 43 Green vertices for which any additional interdiction results in a transition across the performance boundary. These vertices correspond to states where system performance meets mission requirements but is fragile to any additional loss in the original flow network. In contrast, the metagraph has only 13 Red vertices where restoration of any single broken component in the flow network returns the system to a state where performance meets mission requirements (e.g., Figure 3.12). When the system is in one of these states, it can be more easily repaired. Taken together, this suggests potential benefit from focusing on state transitions and operational art for guiding network shaping activities, rather than simply network performance.

CHAPTER 4:
Analysis

4.1 Notional Eighteen-Edge Infrastructure Flow Network

In this chapter, we apply the metagraph and its measures of distance to the notional fuel network from Alderson et al. (2015).

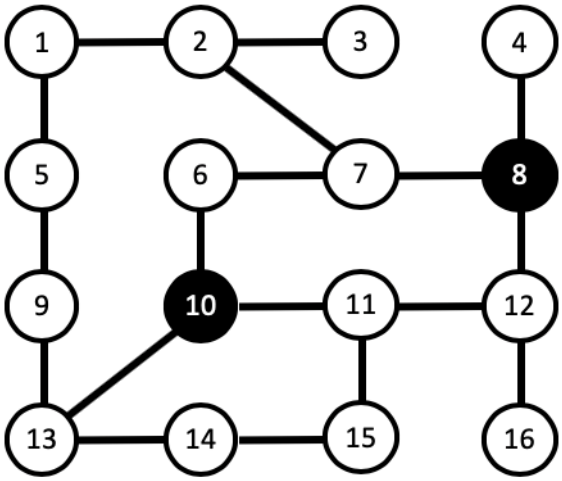

Figure 4.1. The Eighteen-Edge Flow Network. This network has 16 nodes and 18 edges. Supply, fuel in this case, enters at nodes 8 and 10. Each demand node requires 1 barrel of fuel.

The notional fuel network delivers fuel from two supply locations, colored in black, to fourteen demand locations, colored in white. In this system, fuel can travel in either direction through the links that connect the supply and demand nodes but each link has a limited capacity. The scenario has each demand location requiring a single barrel of fuel, each supply location has 10 barrels, and each link can carry a maximum of 15 barrels. Each demand node carries a penalty of $10 if demand is unmet and the cost to send fuel across a link is $1. We assume that each edge has a per-unit traversal cost $c_{ij} = 1$, an upper bound on undirected flow $u_{ij} = 15$, and per-unit penalty cost $q_{ij} = 10$. This operator problem includes the additional constraints that certain links are interdicted and cannot be restored.

This system has a total of $2^{18} = 262,144$ states. For each state, we run the Operator Model

Node (n)	Supply (d_n)	Penalty (p_n)
1	-1	10
2	-1	10
3	-1	10
4	-1	10
5	-1	10
6	-1	10
7	-1	10
8	10	10
9	-1	10
10	10	10
11	-1	10
12	-1	10
13	-1	10
14	-1	10
15	-1	10
16	-1	10

Table 4.1. Eighteen-Edge Flow Network Node List. These are the node parameters used to evaluate the metagraph for the Eighteen-Edge Flow Network. The origination nodes, node 8 and node 10, have the initial supply which must flow through the network to meet the demand of the other nodes which all require 1 unit of supply. For each node, there is a penalty of 10 if the demand is unmet.

# Interdictions	Edges	$p(s)$	Bitstring
1	[4,8]	34	0000100000000000000
2	[2,7],[10,13]	62	0001000000100000000
3	[2,7],[10,13],[11,15]	87	0001000000000101000
4	[2,7],[8,12],[10,11],[10,13]	113	0001000000101100000
5	[6,10],[7,8],[8,12],[10,11],[10,13]	131	0000000111011000000

Table 4.2. Worst-case Interdictions. Each row shows the optimal interdiction for the respective number of interdicitons. The Edges column shows the specific interdicted edges, the performance value is for the specific state represented by the interdicted edges, and the Bitstring column is the corresponding bitstring for the specific state.

from Alderson et al. (2015) to solve the minimum cost network flow problem. The resulting performance values range from 25 (the optimal min-cost performance) to 140 (worst-case

performance).

We enumerate all states in the metagraph and identify the worst possible performance for a given number of interdictions to the network (See: Table 4.2), validating against the results in Alderson et al. (2015). Specifically, the lowest performance values for [1, 2, 3, 4, 5] interdictions to the network are [34, 62, 87, 113, 131] which correspond to system state represented by the bitstrings presented in Table 4.2.

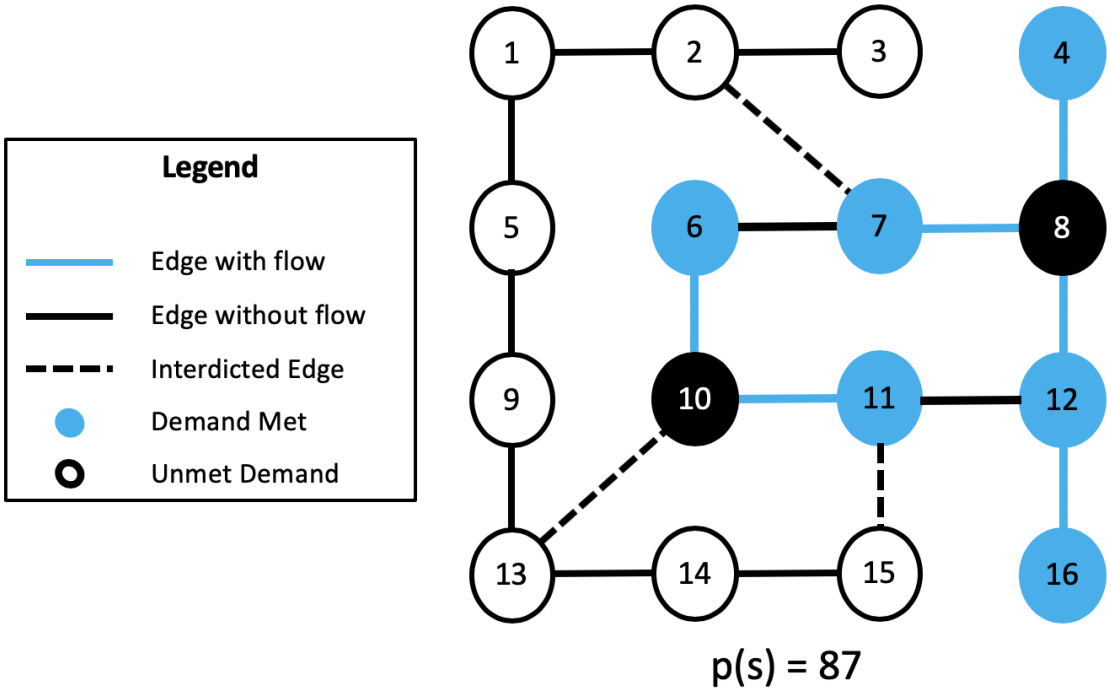

Figure 4.2. Vertex '0001000000000101000': The Worst-Case Performance Value for Three Interdictions. This is the flow network for the worst-case performance value with three interdictions. The blue lines indicate flow through the network and blue-filled nodes have their demand met. The dashed black lines show interdicted edges and solid black lines are intact edges that do not have flow traversing the edge. Empty black circles show nodes that have unmet demand.

Figure 4.2 presents the flow situation for the worst-case interdiction of three arcs in the Eighteen-Edge Flow Network. Figure 4.2 shows the vertex '0001000000000101000' which contains the simultaneous interdiction of edges [2,7], [10,13], and [11,15]. These three interdictions result in the system performance $p(s) = 87$, where nodes [1, 2, 3, 5, 9, 13, 14,

15] have unmet demand.

Despite attacker-defender analysis identifying three edges for worst-case interdiction, the analysis provides no information regarding how to achieve a similar operational result of $p(s) = 87$ when it is not possible to simultaneously interdict [2,7], [10,13], and [11,15]. Specifically, interdiction analysis provides limited insight for other metagraph states that have achieve similar operational effects, whether [2,7], [10,13], and [11,15] are all equally critical to achieve the operational effects depicted in Figure 4.2, or in what way an attacker or defender may want to interdict or restore arcs in response this interdiction. Instead, network shaping methods give us a toolkit to help identify this information that is not considered during normal interdiction analysis and identify more options for network interdiction and restoration.

4.2 Network Shaping Analysis

4.2.1 Analysis of the Metagraph

We begin our analysis assuming a performance threshold $\tau = 87$, which is equal to the worst-case performance value for three interdictions, we assume this is a critical threshold for a military option, specifically, if we can achieve $\tau = 87$, we have imposed a significant cost on the adversary. We build the metagraph and calculate both the corresponding Performance Distance and the Threshold Hamming Distance for each state. We observe $\delta(s) \in [0, 62]$ for $s \in S$. We also observe the Threshold Hamming Distances, $\theta(s) \in [1, 6]$, meaning that every vertex is within six transitions of the boundary. The histogram in Figure 4.3 provides a different view of key features. The bottom row of Figure 4.3 shows the distribution of performance values $p(s)$ for all $s \in S$. Each of the plots above it show the similar distribution, but partitioned for different values $\theta(s) = 1, 2, 3, 4, 5, 6$. The blue line is the boundary threshold, $\tau = 87$, and we therefore color values below and above the threshold green and red, respectively.

Figure 4.3 shows the existence of vertices in the metagraph that have relatively large δ and large θ as well as those that have both relatively small δ and large θ. Like Figure 3.8 in Section 3.5, we observe vertices in the metagraph that have performance $p(s) = 25$ (which is the lowest possible operating cost) with corresponding performance distance $\delta(s) = 53$

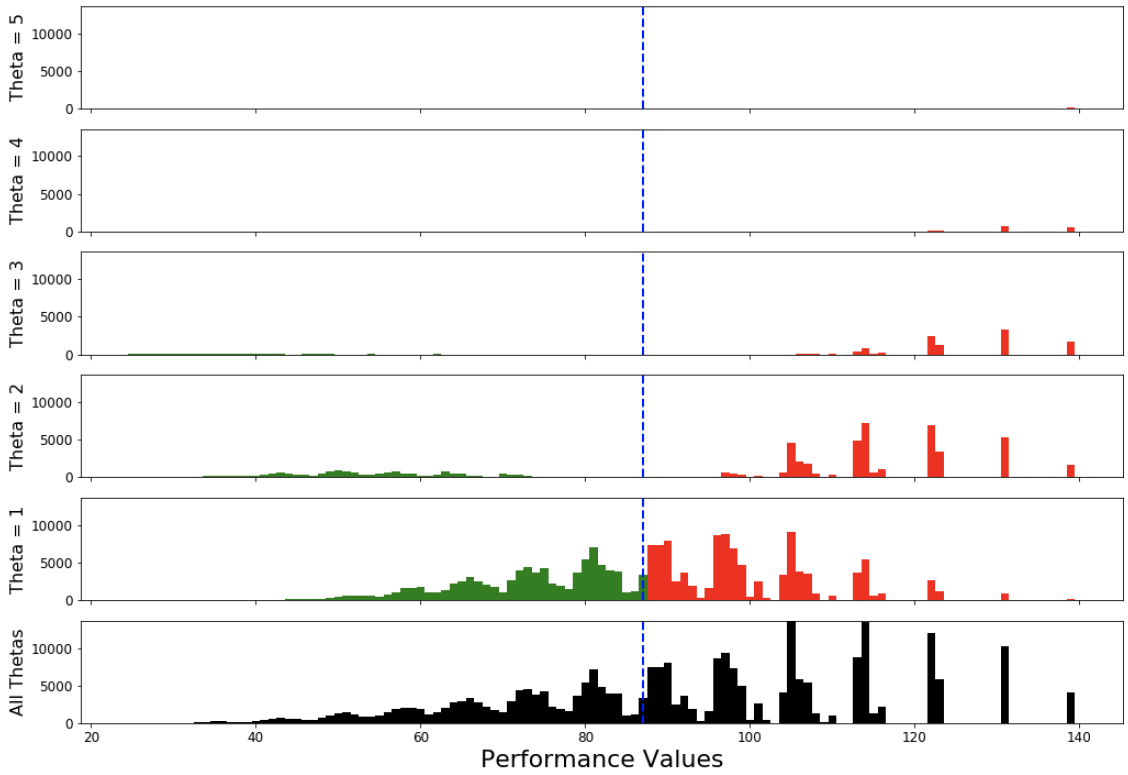

Figure 4.3. Histograms of Metagraph Vertices by Hamming Distance (τ=87). This collection of histograms shows the total counts of vertices in the metagraph with a given performance value (bottom row), as well as separated by Hamming Distances $\theta = 1, 2, 3, 4, 5$ (top five rows, as indicated). The histogram labeled $\theta = 1$ shows the distribution of vertices by performance value for all vertices that are only one transition from crossing the performance boundary. The histograms labeled $\theta = 2, 3, 4, 5$ show the distribution of vertices by performance value that are two, three, four, or five transitions away from crossing the performance boundary, respectively.

(the largest possible margin for a Green state is $\delta(s) = 62$), but also have $\theta(s) = 1$, meaning that they are adjacent to the boundary and require only a single transition to change from Green (success) to Red (failed).

Figure 4.4 shows the distribution of states by level, with each colored according to whether they are above or below the threshold ($\tau = 87$).

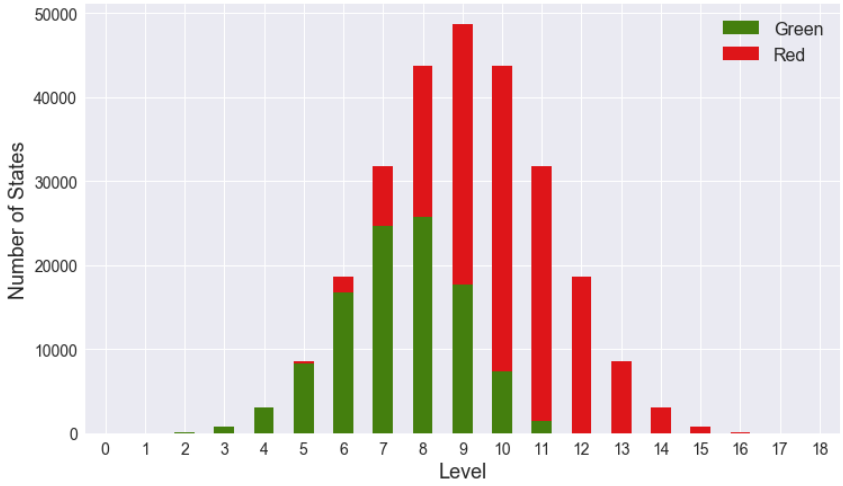

Figure 4.4. Histogram of States in Metagraph for Eighteen-Edge Flow Network, by Level and Color. The $2^{18} = 262,144$ states of the metagraph are distributed across 19 $(k+1)$ levels, with the height of each color representing the relative number of states that are above or below the threshold ($\tau = 87$). This Figure shows the symmetrical nature of a metagraph.

4.2.2 State Transitions Across the Boundary

Vertices With a Single Transition Edge Across the Boundary

The metagraph for Eighteen-Edge Flow Network (τ=87) has 262,144 total vertices and 190,861 states are adjacent to the boundary (i.e., 89,086 Green vertices and 101,775 Red vertices have Threshold Hamming Distance, $\theta = 1$). Figure 4.5 shows the counts of all Green and Red vertices adjacent to the boundary by their number of transition edges that cross the boundary. There are only 253 pairs in which a Green vertex has exactly one Red neighbor and the Red vertex has exactly one Green neighbor. Just like the example shown in Figure 3.11 in Section 3.5.2, these pairs exist at varying levels of the metagraph. The 253 paired Green vertices with only a single transition across the boundary exist at levels [6, 7, 8, 9, 10]. The corresponding 253 Red vertices exist at [7, 8, 9, 10, 11].

The threshold $\tau = 87$ represents the worst case outcome for three interdictions and corresponds to interdicting edges [2,7], [10,13], and [11,15]. The adversary is likely to have realized these edges are critical and fortified their defenses. Interdicting any one of them may require a special type of military operation, such as a special forces raid. Suppose we want to make the network brittle, but still functional, and find a single edge that we can

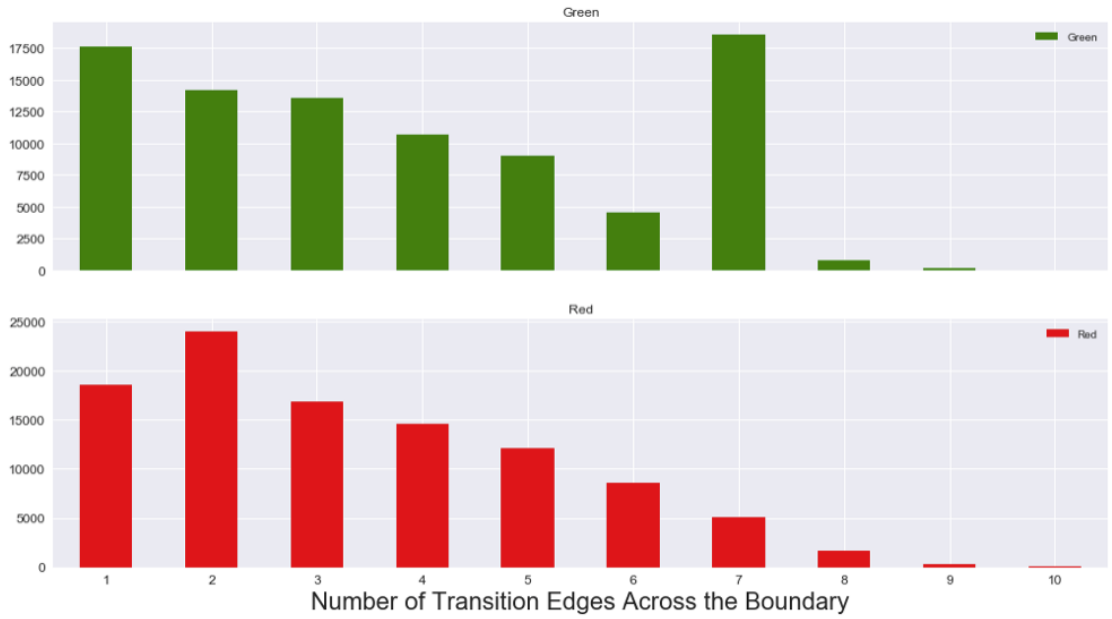

Figure 4.5. Histogram of Transitions Across the Boundary ($\tau = 87$) for Metagraph Vertices with $\theta(s) = 1$. Of the 262,144 metagraph states for the Eighteen-Edge Flow Network, only 190,861 have $\theta = 1$, (i.e., are adjacent to the performance boundary). Each histogram shows the number of potential transitions that cross the performance boundary for Green vertices (89,086 states) and Red vertices (101,775 states). Although some vertices have only a single way to transition across the performance boundary, many have multiple ways of doing so.

interdict at a time and place of our choosing to cause major disruption. Two important questions to answer are: which of critical edge do we need to interdict to achieve this result, and, is the critical edge one of the three fortified edges?

Figure 4.6 shows how using $\delta(s)$ and $\theta(s)$ help identify system states that answer these questions. Figure 4.6 depicts the Eighteen-Edge Flow Network for three different Green vertices [A, B, C] found by first identifying states on the adjacent to boundary with only a single transition edge across the boundary, and then finding neighboring vertices with large $\delta(s)$. These three vertices all have three interdictions ($\lambda = 3$) and $p(s) < \tau$, meaning they are operational (i.e., Green) with performance values of $p(s)$ = [38, 39, 41], respectively. Moreover, the vertices [A, B, C] in Figure 4.6 do not share neighbors with the vertex state in Figure 4.2 in the metagraph, meaning there is no obvious way to transition from any of these states to the worst-case disruption. We find that when we interdict [10,13] for the

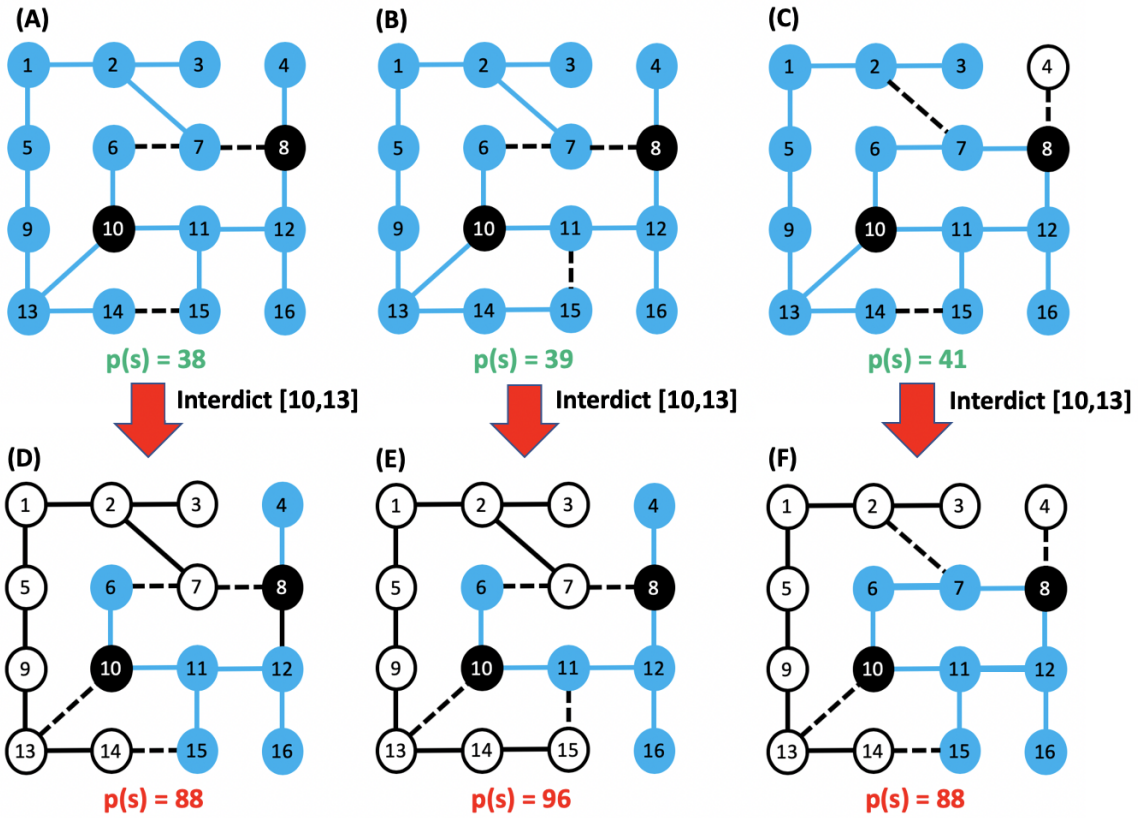

Figure 4.6. Demonstrating the Importance of Edge [10,13] on Flow Network Performance for Situations with More than Three Interdictions. Vertices [A, B, C] have been interdicted three times to have the nominal performance $p(s)$ = [38, 39, 41], respectively. When [A, B, C] have edge [10,13] from the worst-case attack set also interdicted, they produce states [D, E, F] with failed performance of performance $p(s)$ = [88, 96, 88], respectively. Even though edges [2, 7] and/or [11, 15] are also from the worst-case attack set, states [D, E, F] have some combination of these edges still available. This result means that [10,13] may be more important to interdict the 18-Edge Flow Network than other edges. In the figure, blue lines indicate edges that can support flow, dashed black lines indicate interdicted edges, black nodes are supply, blue nodes are met demand, white nodes are unmet demand.

three vertices [A, B, C] producing the three states [D, E, F] also depicted in Figure 4.6, the performance values for each state become $p(s)$ = [88, 96, 88] and now exceed the threshold and become Red vertices. In other words, if the attack budget is increased from three to four interdictions, then it is possible to make the network brittle, but still functional, such that the interdiction of the single edge [10,13] causes major disruption.

Notably, identifying these states in the metagraph provide information regarding the importance of each arc identified in attacker-defender analysis. The original interdiction analysis provided no information regarding which edge of the attack set [2,7], [10,13], and [11,15] affects the flow the most. The states depicted in Figures 4.6 help answer this question. Specifically, the state depicted in Figure 4.6 (D) achieves the desired system performance without interdicting edges [2,7] and [11,15] and instead by interdicting edges [6,7], [7,8], and [14,15]. In contrast, if we are able to interdict [11,15] but not [2,7], Figure 4.6 (E) shows that we can get the same operational effect by interdicting [6,7] and [7,8]. If we can interdict [2,7] but not [11,15], the state depicted in Figure 4.6 (F) shows we can get the same operational effect by interdicting [4,8] and [14,15]. Overall, there are no system states that achieve the operational result of crossing the threshold $p(s) > \tau$ without interdicting [10,13], but several options to achieve that result without interdicting [2,7] or [11,15].

Thus, we can claim that [10,13] is possibly more important than [2,7] or [11,15], despite all three arcs being part of the worst-case interdiction set. Moreover, we can determine that achieving the desired network disruption requires interdiction of the fortified edge [10,13], but not necessarily the other two edges. Figure 4.6 pinpoints how to achieve these operational effects when there are not enough resources for multiple special operations or [2,7] and [11,15] are not available for attack.

Vertices With Multiple Transition Edges Across the Boundary

Of the 89,086 Green vertices with Threshold Hamming Distance, $\theta = 1$, there are 1,252 states in which the number of possible interdiction transition edges exactly matches the number of transitions across the boundary (i.e., number of transition edges is $\delta - k$). The 1,252 vertices exist at Levels [8, 9, 10, 11, 12]. These states, while currently operational (i.e., Green) exist in a precarious situation as the next interdiction will cause system failure.

Likewise, of the 101,775 Red vertices with Hamming Distance, $\theta = 1$, there are 486 states in which the number of possible transitions across the boundary is λ. These Red states exist over 7 different levels, [4, 5, 6, 7, 8, 9, 10]. The Red vertex '0010111110110011010' whose performance $p(s) = 92$ occurs at level 10. That vertex has ten interdictions at edges: [2,3], [4,8], [5,9], [6,7], [6,10], [8,12], [9,13], [11,12], [11,15], and [13,14]. A restoration on any of those edges switches the state from Red to Green. For example the Green vertex '0010111110010011010' whose performance $p(s) = 75$ is best possible restored performance

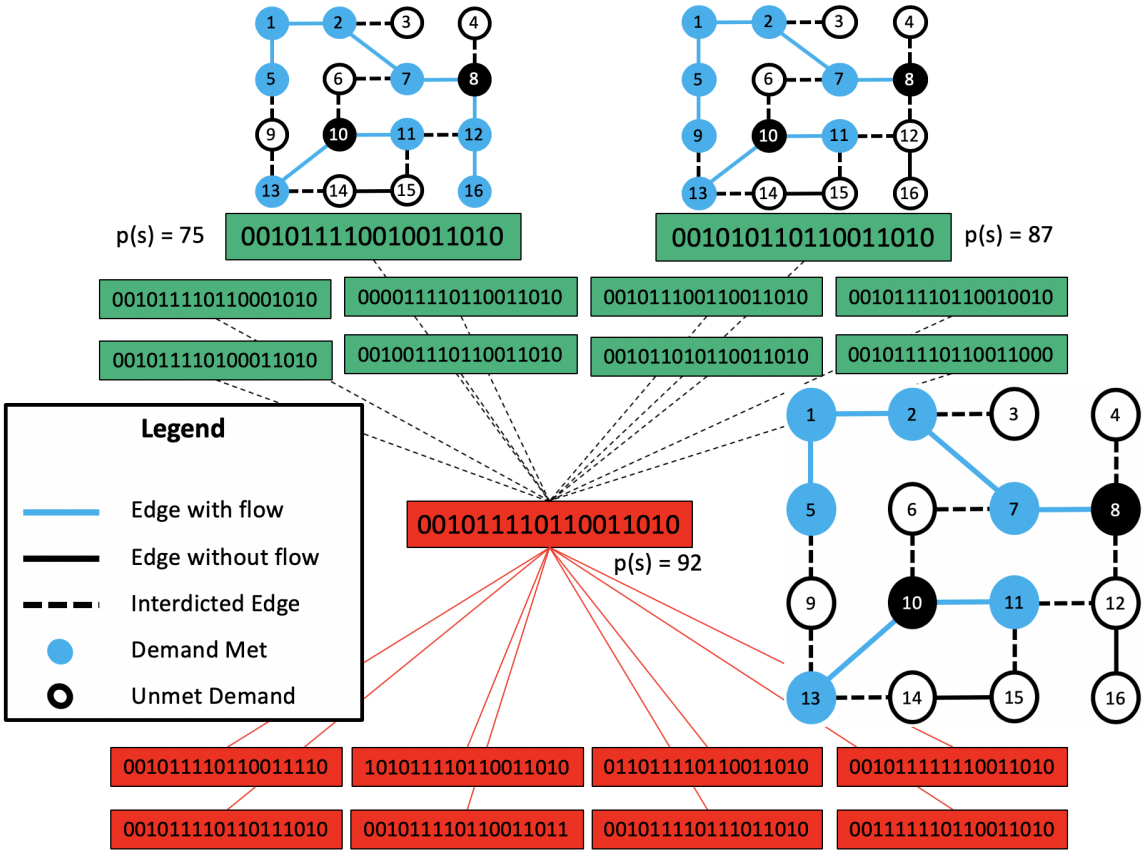

Figure 4.7. Red Vertex with All Interdiction Transitions Crossing the Boundary. This Red vertex '0010111110110011010' has been interdicted ten times, yet if any of the ten interdictions were restored, the system would change to Green. The two Green vertices, '0010111110010011010' and '0010110110110011010', represent the upper and lower bounds for the performance values of those restorations.

value and the Green vertex '0010110110110011010' whose performance $p(s) = 87$ shows the worst restored performance value for the Red vertex '0010111110110011010'. Both bounds of performance values cause the system to change from failure to success.

The Red vertex '0010111110110011010' in Figure 4.7 is very useful if an operator needs to 'fix it fast' as any single restoration changes the state to Green. The Green vertex '0010110110110011010' with $p(s)$=87 in Figure 4.7 has the same performance value as the original, three-interdiction vertex in Figure 4.2 but with seven additional interdictions. The two states share only one interdicted edge, [11,15]. This matters when we consider

the practical applications of these interdictions. Various factors determine how many interdictions an attacker can impose, especially cost. We do not know the costs associated with imposing these interdictions but we know that the results are the same, in terms of performance value. Traditional methods find the efficient means to reach an outcome but military operations planning cannot assume that efficient is feasible. Alternatives are necessary and finding states like those in Figures 4.6 and 4.7 is not possible using only traditional interdiction/restoration methods.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5:
Conclusion and Future Work

5.1 Conclusions

This thesis addresses the relationship between network interdiction and network restoration in the military "operational art" to inform network shaping activities. Operational art is the way the military develops and conducts planning to execute six joint operational phases – shaping, deterring, seizing initiative, dominating, stabilizing, and enabling civil authority (Hart et al. 2014). Operational plans focus on the process of interdicting flow networks (i.e., Phases II and III) to eventually restore them (i.e., Phases IV and V). Traditional interdiction and restoration methods provide important options to achieve these goals by identifying the worst possible way to interdict a flow network and the best possible to restore it.

However, more options might be needed than just the best- or worst-case situation for military operations. Shaping activities that occur before Phases II, III, IV, and V that help operational planners "set conditions for successful theater operations," (Joint Chiefs of Staff 2017) would benefit from a broader range of options, especially when the worst-case interdiction or best-case restoration are not available due to lack of resources or strong enemy defenses. Identifying flow network states that provide additional options than traditional interdiction and restoration models is critical to military operational art. Thus, this thesis focused on answering a single motivation question to inform network shaping activities: ***Are there system states not normally identified via traditional [interdiction/restoration] methods that are operationally relevant and inform new approaches to joint operations?***

Our approach to answer this question focuses on creating and analyzing the metagraph of system states for a flow network. We demonstrate the creation of a metagraph by enumerating every state in a four-edge and nine-edge flow network. Furthermore, we use methods originally implemented in Alderson and Carlyle (2017) to identify a performance threshold and color every vertex in the metagraph as either Green (i.e., functioning) or Red (i.e., non-functioning) with respect to the threshold. We define two new measures to analyze the metagraph to identify states that were adjacent to the performance threshold boundary

between Green and Red states: performance distance, δ, and Threshold Hamming Distance, θ. We then use these two measures to identify attack and defense sets corresponding to system states not identified by traditional methods: vertices on opposite sides of the performance boundary share their only connection across the boundary and vertices for which *all* of the interdiction-related edges (for Green) or *all* of the restoration-related edges (for Red) cross the performance boundary.

These two kinds of metagraph states are relevant for informing military operations. The first state provides an important way to identify critical edges in the flow network that provide more detail on which interdictions and restorations may be more important to achieving a particular operational result. The second provides an target state for military operations during interdiction and restoration, as any single additional change to the flow network will result in a dramatic change in system performance.

To establish the existence and importance of these two states in more realistic systems, we analyze and observe these states in the eighteen-edge notional fuel network. Using the same methods for smaller systems, we are able to pinpoint similar flow network states that achieve similar military operational results. Moreover, we find that these flow network states recommend interdictions and restoration activities that are not necessarily correlated with the worst-case set identified by traditional network interdiction and restoration methods. The presence of these states indicates that, from an operational planning perspective, alternatives exist that may improve the transition from "break it bad" to "fix it fast". Thus, the answer to our guiding question is: ***Yes, there are system states not identified by traditional interdiction and restoration methods that can inform new ways to shape flow networks.***

Both the creation of new analysis methods and the identification of alternative network states for interdiction and restoration provide a basis for better defining network shaping in military operational planning. In general, our conclusions suggest that traditional interdiction and restoration methods represent only a subset of network shaping activities. We more broadly define network shaping as real operational activities to interdict and restore a flow network supported by understanding how to traverse through the metagraph via combinations of interdictions and restorations. Traversing the metagraph is akin to the operational function of Movement and Maneuver which "encompasses the disposition of joint forces to conduct operations by securing positional advantages before or during combat operations" (Joint

Chiefs of Staff 2017). Network shaping, much like maneuvering, is about trying "to achieve a position of advantage in respect to the enemy" (Joint Chiefs of Staff 2017) in the state space for a flow network. Achieving those positions of advantage within the metagraph depends upon the goal. Defenders must find positions that allow freedom of maneuver to successful states whereas attackers seek to impede that same movement from an adversary. Network shaping, like maneuver, "requires designating and then, if necessary, shifting the main effort and applying the principles of mass and economy of force" (Joint Chiefs of Staff 2017). Network shaping uses performance distance and Threshold Hamming Distance in addition to performance value to understand the network.

5.2 Future Work

There are several aspects of this problem that merit further attention.

5.2.1 Effects of the Operational Environment

We consider our flow networks in the most objective means possible. However, the condition of the operational environment would affect the way certain states might be interpreted. For example, a Green vertex in the metagraph with only a single edge that crosses the boundary might be considered 'robust' by a planner. In contrast, a Green vertex for which any interdiction results in a transition across the performance boundary might be considered 'fragile'. This interpretation would impact the the type of state an operational planner might seek.

5.2.2 Enumeration

Our methods for identifying system states require the complete enumeration of a network. The Eighteen-Edge Flow Network demands that we solve the underlying min-cost flow problem 262,144 times to determine the performance for each state, and larger networks grow exponentially making this process time consuming and inefficient. Discovery of a means to find important states without complete enumeration would assist operational planners.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

Ahuja RK, Magnanti TL, Orlin JB (1993) *Network Flows: Theory, Algorithms, and Applications* (Prentice Hall, Upper Saddle River, New Jersey).

Alderson D, Brown G, Carlyle W (2013) Sometimes there is no most-vital arc: Assessing and improving the operational resilience of systems. *Military Operations Research* 18(1), http://doi.org/10.5711/1082598318121.

Alderson D, Brown G, Carlyle W (2014) Assessing and improving operational resilience of critical infrastructures and other systems. *Tutorials in Operations Research* Http://dx.doi.org/10.1287/educ.2014.0131.

Alderson D, Brown G, Carlyle W (2015) Operational models of infrastructure resilience. *Risk Analysis* 36(4), http://doi.org/10.1111/risa.12333.

Alderson D, Brown G, Carlyle W, Wood R (2011) Solving defender-attacker-defender models for infrastructure defense. *Operations Research, Computing,and Homeland Defense* 28–49.

Alderson DL, Carlyle WM (2017) Enumeration and bounding arguments for infrastructure resilience analysis. INFORMS Computing Society Conference, 16 Janualry, Austin, Texas.

Bastian M, Heymann S, Jacomy M (2018) Gephi: An open source software for exploring and manipulating networks, version 0.9.2 201709242018. Accessed April 2019, http://www.gephi.com/.

Brendecke JW (2016) *Optimal repair and replacement policy for a system with multiple components*. Master's thesis, Naval Postgraduate School, Monterey, CA, http://hdl.handle.net/10945/49422.

Brown G, Carlyle W, Salmerón J, Wood K (2005) Analyzing the vulnerability of critical infrastructure to attack and planning defenses. *Tutorials in Operations Research* Https://doi.org/10.1287/educ.1053.0018.

Brown G, Carlyle W, Salmerón J, Wood K (2006) Defending critical infrastructure. *Interfaces* 36(6), http://dx.doi.org/10.1287/inte.1060.0252.

Church R, Middleton R, Scaparra M (2004) Identifying critical infrastructure: The median and covering facility interdiction problems. *Annals of the Association of American Geographers* 94(9):491–502.

Church R, Scaparra M (2006) Protecting critical assets: The r-interdiction median problem with fortification. *Geographical Analysis* 39(2):129–146.

Clark CR (2017) *The threshold shortest path interdiction problem for critical infrastructure resilience analysis*. Master's thesis, Naval Postgraduate School, Monterey, CA, http://hdl.handle.net/10945/56115.

Cochran JJ, Cox LA, Keskinocak P, Kharoufeh JP, Smith JC, Wood RK (2011) Bilevel network interdiction models: Formulations and solutions. *Wiley Encyclopedia of Operations Research and Management Science* Https://doi.org/10.1002/9780470400531.eorms0932.

Corley HW, Chang H (1974) Finding the n most vital nodes in flow networks. *Management Science* 21(3):362–364.

Cormican KJ, Morton DP, Wood RK (1998) Stochastic network interdiction. *Operations Research* 46(2):184–197.

Dantzig G (1963) *Linear Programming and Extensions* (Princeton University Press, Princeton).

Dantzig GB, Fulkerson DR (1955) On the max flow-min cut theorem of networks. Technical Report RM-1418-1, Rand Corporation, Santa Monica, CA.

Ford LR, Fulkerson DR (1954) Maximal flow through a network. Technical Report RM-1400, Rand Corporation, Santa Monica, CA.

Fulkerson DR, Dantzig GB (1954) Computation of maximal flows in networks. Technical Report RM-1400, Rand Corporation, Santa Monica, CA.

Hamming R (1950) Error detecting and error correcting codes. *The Bell System Technical Journal* 29(2), http://doi.org/10.1002/j.1538-7305.1950.tb00463.x.

Hart S, Klosky J, Katalenich S, Spittka, B Wright W (2014) *Infrastructure and the Operational Art* (ERDC/CERL TR-14-14).

Israeli E, Wood RK (2002) Shortest-path network interdiction. *Networks* 40(2):97–111.

Joint Chiefs of Staff (2017) Joint operations. JP 3-0, Washington, DC, https://www.jcs.mil/Portals/36/Documents/Doctrine/pubs/jp3_0ch1.pdf?ver=2018-11-27-160457-910.

Lim C, Smith JC (2007) Algorithms for discrete and continuous multicommodity flow network interdiction problems. *IIETransactions* 39(1):15–26.

McMasters AW, Mastin TM (1970) Optimal interdiction of a supply network. *Naval Research Logistics Quarterly* 17(3):261–268.

Murray AT, Matisziw TC, Grubesic TH (2007) Critical network infrastructure analysis: Interdiction and system flow. *Journal of Geographical Systems* 9(2):103–117.

Ratliff DH, Sicilia GT, Lubore SH (1975) Finding the n most vital links in flow networks. *Management Science* 21(5):531–539.

Salmerón J, Wood K, Baldick R (2004) Analysis of electric grid security under terrorist threat. *IIE Transactions on Power Systems* 19(2):905–912.

Salmerón J, Wood K, Baldick R (2009) Worst-case interdiction analysis of large-scale electric power grids. *IIE Transactions on Power Systems* 24(1):96–104.

Schrijver A (2002) On the history of the trans-portation and maximum flow problems. *Mathematical Programming Series B* 91(3), https://doi.org/10.1007/s101070100259.

Schulze C (2014) *A comparison of techniques for optimal infrastructure restoration*. Master's thesis, Naval Postgraduate School, Monterey, CA, http://hdl.handle.net/10945/44665.

Snyder L, Scaparra M, Daskin M, Church RL (2006) Planning for disruptions in supply chain networks. *Tutorials in Operations Research: Models, Methods, and Applications for Innovative Decision Making* .

Washburn A, Wood K (1995) Two-person zero sum games for network interdiction. *Operations Research* 43(2):243–251.

Wollmer R (1964) Removing arcs from a network. *Operations Research* 12(6):934–940.

Wollmer R (1968) Stochastic sensitivity analysis of maximum flow and shortest route networks. *Management Science* 14(9):1–18.

Wollmer RD (1963) Some methods for determining the most vital link in a railway network. Technical Report RM-3321-ISA, Rand Corporation, Santa Monica, CA.

Wood RK (1993) Deterministic network interdiction. *Mathematical and Computer Modelling* 17(2):1–18.

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California