Decision Analysis

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Interdicting Attack Plans with Boundedly Rational Players and Multiple Attackers: An Adversarial Risk Analysis Approach

Eric DuBois, Ashley Peper, Laura A. Albert

To cite this article:
Eric DuBois, Ashley Peper, Laura A. Albert (2023) Interdicting Attack Plans with Boundedly Rational Players and Multiple Attackers: An Adversarial Risk Analysis Approach. Decision Analysis 20(3):202-219. https://doi.org/10.1287/deca.2023.0471

https://pubsonline.informs.org/journal/deca

DECISION ANALYSIS
Vol. 20, No. 3, September 2023, pp. 202–219
ISSN 1545-8490 (print), ISSN 1545-8504 (online)

Interdicting Attack Plans with Boundedly Rational Players and Multiple Attackers: An Adversarial Risk Analysis Approach

Eric DuBois,[a] Ashley Peper,[b] Laura A. Albert[b,*]

[a] The Center for Naval Analyses, Arlington, Virginia 22201; [b] Department of Industrial and Systems Engineering, University of Wisconsin–Madison, Madison, Wisconsin 53706
*Corresponding author
Contact: duboise@cna.org (ED); apeper2@wisc.edu (AP); laura@engr.wisc.edu, https://orcid.org/0000-0001-7079-4473 (LAA)

Received: August 2, 2022
Revised: February 3, 2023
Accepted: February 12, 2023
Published Online in Articles in Advance: March 28, 2023

https://doi.org/10.1287/deca.2023.0471

Abstract. Cybersecurity planning supports the selection of and implementation of security controls in resource-constrained settings to manage risk. Doing so requires considering adaptive adversaries with different levels of strategic sophistication in modeling efforts to support risk management. However, most models in the literature only consider rational or nonstrategic adversaries. Therefore, we study how to inform defensive decision making to mitigate the risk from boundedly rational players, with a particular focus on making integrated, interdependent planning decisions. To achieve this goal, we introduce a modeling framework for selecting a portfolio of security mitigations that interdict adversarial attack plans that uses a structured approach for risk analysis. Our approach adapts adversarial risk analysis and cognitive hierarchy theory to consider a maximum-reliability path interdiction problem with a single defender and multiple attackers who have different goals and levels of strategic sophistication. Instead of enumerating all possible attacks and defenses, we introduce a solution technique based on integer programming and approximation algorithms to iteratively solve the defender's and attackers' problems. A case study illustrates the proposed models and provides insights into defensive planning.

Funding: A. Peper and L. A. Albert were supported in part by the National Science Foundation [Grant 2000986].

Keywords: cybersecurity • security • attacker/defender • optimization

1. Introduction

Cybersecurity is an important concern for governments and organizations throughout the world due to the growing reliance on digital connectivity and the growing number of threats. Cyberattacks are increasingly common, costing the U.S. economy $57–$109 billion in 2016 (Council of Economic Advisors 2018) and affecting systems throughout the economy, including in healthcare (Kruse et al. 2017), energy (Wang and Lu 2013), and industrial control (Knowles et al. 2015). Many possible security controls exist to mitigate these risks (Ross et al. 2021). Cybersecurity planning requires periodically selecting a portfolio of security controls (e.g., on an annual basis), which allows an organization to manage the risk associated with vulnerabilities that have emerged. However, many organizations find it challenging to keep up with selecting and deploying security controls given that they operate in resource-constrained environments (Stevens et al. 2020).

There is a growing body of literature that applies risk analysis techniques to manage cybersecurity risk through the strategic prioritization of security controls. Some of these efforts model attackers as nonstrategic players using probability distributions and prioritize security controls in rank order based on their cost-effectiveness (Hubbard and Seiersen 2016). Increasingly, security controls are prioritized using a structured approach to aid in planning efforts with well-defined goals and threat scenarios (National Institute of Standards and Technology 2018), where attack graphs are used to represent known vulnerabilities and visualize potential mitigations (Lallie et al. 2020). Recent research uses integer programming (Zheng

et al. 2019) and robust optimization (Zheng and Albert 2019b) to select security mitigations using a structured approach based on attack graphs; however, these papers assume attackers are either not strategic or limited to selecting a worst-case scenario.

A stream of papers in the literature explicitly consider adaptive adversaries in cybersecurity planning through the application of adversarial risk analysis (ARA). In these models, a single defender, the security planner, selects security controls that perform well given that adaptive adversaries will attempt to work around any new security controls that are put in place. ARA frameworks are versatile and have motivated defender–attacker models that capture a wide range of conditions and assumptions (Banks et al. 2020). As a result, ARA has been applied to many of these defender–attacker models in various security settings (Rios and Rios Insua 2012) and has been adapted to cybersecurity models with multiple adversaries with different levels of intentionality (Rios Insua et al. 2021). A limitation of ARA approaches is that they enumerate all possible attacks and defenses (Banks et al. 2020), with Wang and Banks (2011) as an exception. Thus, ARA algorithms are intractable when it is not practical to enumerate cybersecurity attack and defense choices. Zheng and Albert (2019a) seek to overcome this challenge by introducing a structure that simplifies computational requirements based on a network interdiction model that delays the attack plans of multiple attackers. The structure introduced by the network interdiction model allows for the use of integer programming algorithms to solve for the defender and attacker strategies.

We build upon previous work by introducing an ARA framework that considers boundedly rational players, including a defender and multiple adversaries, to inform the selection of security controls that interdict adversarial attack plans. Although our approach is motivated by cybersecurity planning, it can be used more broadly in the security context where players seek to maximize or minimize the probability of attack in a multilayer defense system.

1.1. Approach

In the cybersecurity planning problem we consider, a defender seeks to minimize the probability of a successful attack by multiple attackers with different levels of strategic sophistication over a planning horizon by selecting a portfolio of security controls subject to a budget. We adapt an ARA framework (Rios Insua et al. 2021) to capture the strategic selection of a portfolio of security controls given that the defender and the attackers are boundedly rational. This modeling approach allows us to inform defensive decisions and planning against a range of attackers, which more accurately reflects the system we are modeling (Scheibehenne et al. 2010). Because new vulnerabilities emerge on a regular basis, cybersecurity planning should be performed regularly (e.g., annually), and the methods in this paper can aid in this process.

Attack modeling is an important step in cybersecurity planning. In vulnerability analysis, vulnerabilities can be characterized by various steps required to successfully carry out an attack (Schneier 1999), which provides a structured approach to represent attack scenarios. In a graph structure, the nodes represent attack states (e.g., the choice of attack type, the target of the attack, or attack milestones), and edges represent intermediate exploits in an attack. The difficulty of an adversary completing an exploit is captured by a conditional probability of successfully traversing an arc. A path from root to leaf corresponds to an attack against the system, and, therefore, we view an attack as a path in a graph between a source and sink node. Given that there are many adversaries who have different knowledge of the vulnerabilities and different capabilities, the attack graphs may have topology and parameters specific to each adversary.

An adversary's probability of successfully carrying out all exploits in an attack is captured by the probability they traverse the network on the path they select. Security controls interact with attack graphs by decreasing the probability that the completion of individual exploits (arcs that are traversed). Security controls may encourage adversaries to select alternative paths. The defender uses their private information and the paths in the attack graphs that they believe the attacker will choose to determine their choice of security controls.

These strategic interactions motivate the application of the maximum-reliability path interdiction problem to the planning problem under consideration. Network interdiction models, including the maximum-reliability path interdiction, have been widely used to model attacker–defender games, usually assuming two players and rational decision makers (Smith and Song 2020). Although some papers have lifted the assumptions of shared information (e.g., Salmerón 2012) and shared beliefs regarding the probabilities of traversing

edges in the network (e.g., Morton et al. 2007), none have considered boundedly rational players. This paper seeks to fill this gap.

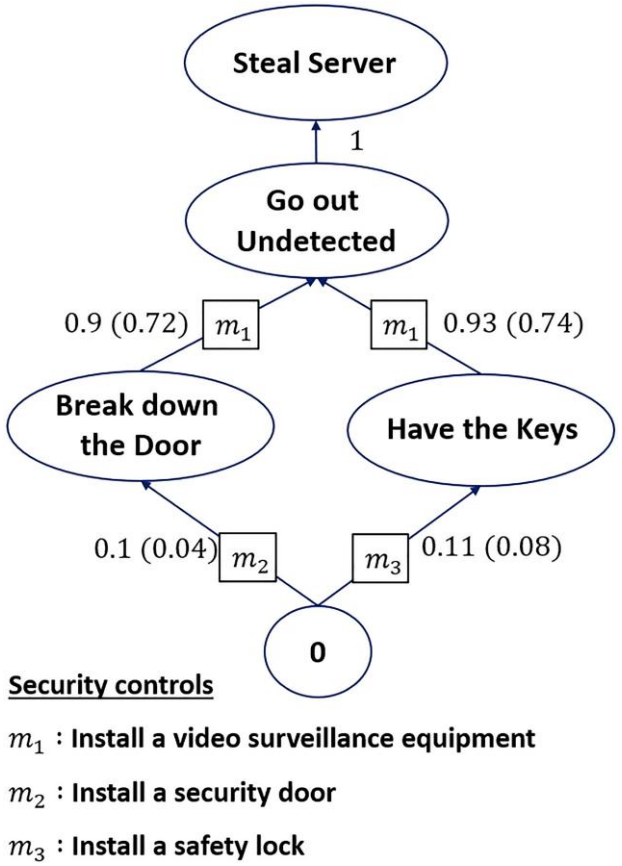
We adapt and apply ideas from the ARA framework presented by Rios Insua et al. (2021) to the maximum-reliability network interdiction problem to support decision making for cybersecurity planning. An ARA approach allows us to consider adversaries that are not rational and who may have varying levels of strategic sophistication, important features of the application under consideration. In particular, we model players who are boundedly rational using cognitive hierarchy theory (Camerer et al. 2004) and level k thinking (Stahl and Wilson 1995).

To illustrate the approach taken in this paper, consider the following example with a single defender and two types of adversaries. In the example, adapted from Bistarelli et al. (2006), adversaries attempt to steal a server. Figure 1 captures the attack graph under consideration, including the exploits, and two pathways to steal the server (reach the top node). There are three security controls (m_1, m_2, and m_3) that are listed along the arcs they interdict. For illustration purposes, we assume the defender can select one control. The uninterdicted traversal probabilities are listed next to each arc, and the interdicted traversal probability is listed in parentheses when the security control is in use. Without any security controls, the left and right pathways to the "steal server" node have traversal probabilities of 0.09 and 0.102, respectively.

In this example, an opportunistic attacker with a low degree of strategic sophistication who ignores possible security defenses selects the right path, because it has a higher uninterdicted traversal probability. A defender who anticipates only attacks from this type of attacker would select m_3, because this mitigation lowers the traversal probability of the right path the most. A slightly more strategic attacker would anticipate m_3 being deployed and would then choose the left path. This would lead a more strategic defender to select control m_2 to defend against attackers attempting only the left path. A defender who anticipates both types of attackers, 25% of whom are somewhat strategic and select the left path and 75% of whom are opportunistic and attempt the right path, would select mitigation m_1.

This simple example based on a single attack graph highlights how a methodology that considers multiple, boundedly rational attackers can be informative for defensive planning decisions. In general, there are many attack graphs that capture various attack vectors as well as adversarial goals and capabilities. The security controls can be specific and delay a single exploit, such as m_2 in the previous example, or many exploits if the controls are cross-cutting, such as deploying multifactor authentication or an employee training program. Singhal and Ou (2017) and Lallie et al. (2020) provide additional guidance surrounding how to model attack graphs.

Figure 1. An Illustrative Example of a Single Attack Graph with Three Security Controls



1.2. Contributions

In summary, this paper makes the following contributions:

- We formulate the security control investment problem in an ARA framework as a maximum-reliability path interdiction game between a single defender and multiple attackers, all of whom are boundedly rational

with differing levels of strategic sophistication. The defender and attackers' problems are formulated as mixed integer programming models.

- We introduce solution techniques based on mixed integer programming algorithms and approximation algorithms. The defender and attackers' problems are solved iteratively, and the inputs to each model are updated after each iteration. We prove that the defender's problem is equivalent to a submodular maximization problem subject to a budget constraint, which enables the use of heuristics for identifying solutions that are at least $1 - 1/e$ of the optimal solution value.
- We apply the modeling approach to a case study. We identify solutions across a range of levels of strategic sophistication and consider the effects of the defender misjudging the sophistication of the attackers.

The organization of this paper is as follows. We first survey the literature in Section 2. In Section 3, we describe the ARA framework. Section 4 introduces the mixed integer programming models that capture the maximum-reliability path interdiction from the attacker and the defender perspectives. We introduce the approximation algorithms in Section 5, and we describe the ARA algorithm in Section 6. We present and analyze the case study and other computational results in Section 7. Section 8 contains concluding remarks.

2. Literature Review

The topics in this paper are related to several different areas of research, including adversarial risk analysis, boundedly rational thinking, security control investment, and maximum-reliability path interdiction. We present a summary of the most relevant papers in the literature.

2.1. Adversarial Risk Analysis

ARA has been widely applied to security applications (Rios and Rios Insua 2012). Several papers use ARA framework for cybersecurity, focusing on defender–attacker games (Wang et al. 2019), insider threat modeling (Joshi et al. 2020), and adversarial machine learning and data manipulation (Caballero et al. 2021). Rios Insua et al. (2021) apply the ARA framework to a cybersecurity resource allocation problem to inform a portfolio of defensive actions, including the purchase of cyber insurance, by including both intentional and nonintentional threats.

As with most ARA models, Rios Insua et al. (2021) enumerate all possible attacks and defenses. Cano et al. (2016) similarly apply ARA to a cybersecurity setting by enumerating specific attacks and defenses to determine an optimal security allocation to minimize disruptions to airport operations. To our knowledge, only Wang and Banks (2011) consider an ARA framework in which the attacks and defenses are not enumerated. They consider the optimal path for a convoy through a network where an attacker has placed several improvised explosive devices at nodes within the network. The defender seeks to minimize the routing cost. By using a network model, they compactly represent many possible convoy routes. Because Wang and Banks (2011) utilize the additive nature of their cost function to efficiently solve their problem, it is not possible to apply their solution method to our problem. In contrast to the existing literature, we consider interdicting attack plans to support cybersecurity planning, and we introduce a methodology to solve defender and attacker problems based on integer programming and approximation algorithms, because enumerating the attack and defense choices is intractable.

2.2. Bounded Rationality

We build on the work of Rothschild et al. (2012), who develop an algorithm for applying bounded rationality, specifically, level k thinking, within an ARA framework. Level k thinking begins with nonstrategic level 0 thinkers who act without regard to other players. In comparison with level k thinking, where a level k player optimizes over only the level $k-1$ opponent, cognitive hierarchy theory assumes that a level k thinker optimizes over a distribution of players between level 0 and level $k-1$ (Camerer et al. 2004). We use this method to model the defender because of the multiple-attacker scenarios they face. Although the logic of level k thinking and cognitive hierarchy theory is theoretically subject to infinite regression, empirically it has been found that most people do not think beyond level 2 or level 3 (Lee and Wolpert 2012). Higher-level thinkers ($k > 0$) assume that their opponent is a level $k-1$ thinker; for example, level 1 players optimize against level 0 players, level 2 players optimize against level 1 players, and so on. Rothschild et al. (2012) create an algorithm for determining the strategy of a level k opponent by using recursion to build belief distributions regarding the attacks or defenses that the player uses. Considering bounded rationality in the adversaries is a novel aspect of our paper. To the best of our knowledge, our paper is the

first to consider boundedly rational players when considering interdependent defensive decisions as considered in network interdiction.

2.3. Security Control Portfolio Selection

A stream of the literature studies how to invest in security controls given a limited budget (Fielder et al. 2016, Zheng and Albert 2019b). Many models consider nonstrategic attacks and do not necessarily select controls that adequately protect against adaptive adversaries (Zheng et al. 2019). Nonstrategic attacks (also called opportunistic or nontargeted) continue to be carried out in roughly the same manner and with the frequency regardless of the defender's security decisions. For example, models using a decision theory framework assume that the defender's decision has no impact on the frequency of each method of attack (Cavusoglu et al. 2008). Although some research seeks to allocate a budget in the presence of strategic attackers and natural disasters (Zhuang and Bier 2007), none of the papers in this area consider how to select a portfolio of security defenses with multiple, boundedly rational adversaries.

2.4. Maximum-Reliability Path Interdiction

Network interdiction models have been widely applied to infrastructure protection and resilience problems, where they inform how to protect critical components in a system to reduce worst-case vulnerability. Network interdiction problems are modeled as Stackelberg or Cournot game models of defender–attacker games, usually assuming two players and rational decision makers. Smith and Song (2020) provide a recent survey of this area. A shortest path interdiction problem may be used to model attacks or projects where the attacker seeks to minimize the time required to complete an attack (traverse the network) and the defender seeks to maximally delay an attack by interdicting (lengthening) edges on the network (Israeli and Wood 2002). The shortest path interdiction problem is an equivalent formulation to the maximum-reliability interdiction problem (Morton et al. 2007). Network interdiction models have been extended to include imperfect and private information (Salmerón 2012). Several researchers have recommended that network interdiction models be extended to consider boundedly rational players (Zhang et al. 2018) and other more realistic features to aid in defensive planning (Albert et al. 2023).

A stream of papers study how to interdict attack graphs to inform defensive cybersecurity planning efforts. Nandi et al. (2016) introduce a bilevel defender–attacker model to help organizations select and deploy security countermeasures by interdicting attack graphs. Letchford and Vorobeychik (2013) introduce a different Stackelberg game in which a defender seeks to interdict an attack plan, lifting the assumption that the game is zero sum. Zheng and Albert (2019a) introduce a bilevel network interdiction model that seeks to identify a portfolio of security controls that maximially delay a large number of adversarial cyberattacks from multiple attackers under uncertainty.

In sum, our paper adds to the literature by combining ARA and cognitive hierarchy theory with network interdiction modeling to inform defensive cybersecurity planning.

3. Adversarial Risk Analysis Framework

In this section, we introduce the ARA framework in this paper as well as how we model the boundedly rational players. ARA takes the perspective of one player, the defender in this paper, and seeks the optimal action for that player based on the actions/reactions they believe the other players will take.

3.1. Player Interactions in the Game

The maximum-reliability network interdiction modeling approach considers a one-off encounter between a single defender and multiple attackers. This is reasonable for cybersecurity investment decisions considered in the case study, where planners must protect information systems from many attacks. In our approach, attackers do not know which security controls have been selected by the defender. The attackers seek to maximize the probability of their attacks succeeding based on perceived defenses, and the defender seeks to prevent an attack. Beliefs about the defender's budget and security control costs are finite and discrete.

We now describe the general form of the game that we consider. There are a set of attackers who each choose a method of attack and target that maximizes their probability of success, which defines their *reliability*. This is equivalent to choosing a path through the network. By choosing a path, the attacker chooses the set of edges they traverse that determines the path's reliability. By selecting controls, the defender reduces the reliability

for each of the edges that the controls cover. The defender selects a portfolio of controls to maximize the conditional probability that an attack is prevented subject to a budget constraint.

Most of the information regarding the structure of this network is treated as shared beliefs, whereas the parameters of the network, such as the edge reliabilities, are private beliefs. Specifically, we assume that the set of nodes, edges, and possible controls are shared between all players. This is equivalent to assuming that the defender knows all possible threats the attackers may consider. The approach informs defensive planning based on known vulnerabilities, and it can be updated to include new vulnerabilities that have been discovered. Other information in the game may be private. This includes the edge reliabilities, the effectiveness of controls at decreasing those reliabilities, the proportion of attacks from each attacker, the cost to institute a control, and the defender's budget.

3.2. Boundedly Rational Players

From the defender's perspective, the attackers may have different levels of strategic sophistication. Therefore, a level k defender plans for level $0, 1, \ldots, k-1$ attackers. An attacker's strategy represents a path. The defender's strategy is a portfolio that consists of a set of controls whose total cost is within the budget.

Because we are seeking an optimal defense, we find strategies for level k defenders up to maximum level K—the defender's level—and strategies for level k attackers up to maximum level $K-1$. Lower level attacker and defender strategies are recorded as the algorithm progresses, which provides a suite of portfolio options corresponding to the differing defender levels of strategic sophistication. By analyzing the various portfolios within this suite, we can make more informed decisions about how the defender's posture should change when confronted by more or less "sophisticated" attackers.

The basic algorithm to compute the strategies of different level k attackers of each type is as follows. We elaborate upon this algorithm in detail in Section 5. We begin with information and beliefs about the system. We start with $k = 0$ and construct level 0 attacker and defender strategies for nonstrategic players. We then increment the value of k by one and compute a level k defender's optimal portfolio using the defender's program, OptDef, presented in Section 4.2. If the strategy for the highest level of defender, K, has been calculated, the algorithm terminates. Otherwise, we calculate the level k attacker's optimal attack path using the attacker's program, OptAtt, presented in Section 4.1. We repeat this step by incrementing k by one and solving the defender's and attackers' problems until the algorithm terminates.

4. Model Formulations

In this section, we formulate a simultaneous single-defender, multiple-attacker game based on a maximum-reliability network interdiction problem. We do so by introducing optimization problems from the attackers' perspective, OptAtt, and the defender's perspective, OptDef. There is a single defender and a set of attackers $\mathcal{A}$. Without loss of generality, each attacker begins at the super source node, 1, and progresses through the graph to the super sink node, n. Each attacker chooses a path of maximum reliability, the probability that they believe they will successfully traverse the graph based on perceived defender decisions.

Notations that reflect the defender's and attackers' beliefs have a D and an A subscript/superscript, respectively. Beliefs that are updated based on the level of strategic sophistication have a k superscript. Notation that captures decision variables does not explicitly include k, D, or A. Table 1 provides a summary of the notation relevant to the parameters and players beliefs as well as the decision variables. For clarity, we use Latin characters for variables and shared information, and we use Greek characters for beliefs.

4.1. Attackers' Problem

We now consider the attackers' problem. We consider a directed acyclic graph $G = (N, E)$ consisting of a finite set of nodes N and directed edges E. The sets of edges leaving and entering a node $i \in N$ are denoted by E_i^+ and E_i^-, respectively. Without loss of generality, we assume that the set of nodes $\{1, \ldots, n\} \in N$ are ordered such that $\delta(i, j, A)$ for all $(i, j) \in E$, and each level 0 attacker selects the edge with the highest noninterdicted reliability from each node, breaking ties randomly. Other methods could be used to set level 0 attacker paths.

We consider the reliability of an attack path for a level k attacker. A level k attacker A believes the level $k-1$ defender has chosen the portfolio $\Omega_A^{k-1} = \{\omega_1, \ldots, \omega_m\}$. The attacker believes that the reliability of edge $(i, j) \in E$

Table 1. Notation

Notation	Definition
Sets: Common information to defender and attackers	
$\mathcal{A}$	Attackers
N	Nodes
E	Edges
$E_i^+(E_i^-) \subseteq E$	Edges that leave (enter) node $i \in N$
M	Set of controls
$M_{ij} \subseteq M$	Subset of controls that interdict edge $(i,j) \in E$
$P(i)$	The set of paths from the source node to node $i \in N$
Level k attacker $A \in \mathcal{A}$ decisions	
Attacker variables	
u_{ij}	Binary variable that is 1 if edge $(i,j) \in E$ is on the attacker's path and 0 otherwise
q_{ij}	Probability that the attack of attacker $A \in \mathcal{A}$ reaches edge $(i,j) \in E$
$\Rightarrow$ level k' (with $k' > k$) defender beliefs	
$\psi_D^{k'}(i,j,A)$	Conditional probability that a level $k' > k$ defender believes a level k attacker $A \in \mathcal{A}$ attempts to traverse edge $(i,j) \in E_i^+$ given that they reach node i
Defender D beliefs	
$\delta_D(i,j,A)$	Reliability of uninterdicted edge $(i,j) \in E$ for attacker $A \in \mathcal{A}$
$\tilde{\delta}_D(i,j,A)$	Reliability of interdicted edge $(i,j) \in E$ for attacker $A \in \mathcal{A}$
$\theta_D(A)$	Conditional probability that attacker A attempts an attack
β_D	Defender's budget
$\kappa_D(m)$	Cost of control $m \in M$
Level k defender decisions	
Defender variables	
w_m	Binary variable that is 1 if control $m \in M$ is chosen and 0 otherwise
x_{ij}	Binary variable that is 1, if edge $(i,j) \in E$ is covered 0 otherwise
$y_{ij}(A)$	Probability that attacker $A \in \mathcal{A}$ reaches uninterdicted edge $(i,j) \in E$
$\tilde{y}_{ij}(A)$	Probability attacker $A \in \mathcal{A}$ reaches interdicted edge $(i,j) \in E$
$\Rightarrow$ level $k+1$ attacker beliefs	
$\Omega_A^k = \{\omega_1, \ldots, \omega_m\}$	Portfolio selected by a level k defender
$\delta_A^{k+1}(i,j\|\Omega_A^k)$	Reliability of edge $(i,j) \in E$ under portfolio Ω_A^k for a level $k+1$ attacker

is $\delta_A^k(i,j|\Omega_A^{k-1})$, its reliability with portfolio Ω_A^{k-1}. We assume that the values of $\delta_A^k(i,j|\Omega_A^{k-1})$ are independent between edges. Given Ω_A^{k-1}, the attacker then believes the reliability of a fixed path p to be $\prod_{(i,j)\in p}\delta_A^k(i,j|\Omega_A^{k-1})$. Each strategic attacker seeks to maximize the conditional probability that their attack succeeds. Using standard approaches based on recursion (Ahuja et al. 1993), we can represent the probability of an attack reaching edge (i,j), captured by $q_{j\ell}$, using the pair of linear inequalities

$$q_{j\ell} \le u_{j\ell},$$
$$q_{j\ell} \le \sum_{(i,j)\in E_j^-} \delta_A^k(i,j|\Omega_A^{k-1}) q_{ij},$$

where the characteristic vector of the path chosen by the attacker is $u \in \{0,1\}^{|E|}$, where $u_{ij} = 1$ if edge (i,j) is in the path, and $u_{ij} = 0$ otherwise.

Using these expressions, we present the OptAtt formulation for level k attacker $A \in \mathcal{A}$ as a maximum-reliability path problem:

$$z_A^k = \max \sum_{(i,n)\in E_n^-} \delta_A^k(i,j|\Omega_A^{k-1}) q_{in} \tag{1}$$

$$\text{s.t.} \sum_{(1,i)\in E_1^+} u_{1i} \le 1, \tag{2}$$

$$\sum_{(i,j)\in E_i^+} u_{ij} \le \sum_{(j,i)\in E_i^-} u_{ji}, \qquad \forall i \in N \setminus \{1\}, \tag{3}$$

$$q_{ij} \le u_{ij}, \qquad \forall (i,j) \in E, \tag{4}$$

$$q_{j\ell} \le \sum_{(i,j)\in E_j^-} \delta_A^k(i,j|\Omega_A^{k-1}) q_{ij}, \qquad \forall j \in N \setminus \{1\}, (j,\ell) \in E_j^+, \tag{5}$$

$$u_{ij} \in \{0,1\}, \qquad \forall (i,j) \in E, \tag{6}$$

$$q_{ij} \ge 0, \qquad \forall i \in N, (i,j) \in E. \tag{7}$$

The objective (1) for an attacker is to maximize the probability that their attack succeeds. Constraints (2) and (3) enforce that the u_{ij} variables properly define a path. Constraint (2) only allows the attacker to choose one attack path, and constraint set (3) preserves the balance of flow in and out of each node. Constraint sets (4) and (5) determine the probability that the attack succeeds. Constraint set (4) allows attacks to progress only along the attack path. Constraint set (5) balances the flow of attack probability in and out of each node. Constraint sets (6) and (7) require variables to take on binary and nonnegative values, respectively. OptAtt is a canonical form maximum-reliability path problem, which can be solved for each attacker as a shortest path problem with Dijkstra's algorithm after a negative logarithm transform of the edge reliabilities (Morton et al. 2007).

After solving OptAtt for each attacker, we use the solution to construct the defender's beliefs about the paths that each attacker $A \in \mathcal{A}$ will take. Recall that a

defender of level k' defends against a set of attackers $\mathcal{A}$ that may have different levels of sophistication k, with $0 \le k < k'$. A solution to OptAtt for a level k attacker $A \in \mathcal{A}$ can therefore be used to inform the beliefs of the level k' defenders, with $k+1 \le k' \le K$. To do so, we derive the paths that a level k attacker A takes from the OptAtt solution variables u_{ij}, which are converted to inform a level k' defender's belief parameters. A level k' defender believes the conditional probability attacker A chooses edge $(i,j) \in E$ after reaching node i is

$$\psi_D^{k'}(i,j,A) = u_{ij}, \ \ \forall (i,j) \in E. \tag{8}$$

Although we do not explicitly model the level k in the attacker variables in OptAtt, the level of the attacker is implicitly retained when setting the values of $\psi_D^{k'}(i,j,A)$.

4.2. Defender's Problem

We now consider the defender's problem. The defender maximizes the probability that an attack is prevented by selecting controls from set M. Each control $m \in M$ has cost $\kappa_D(m)$ subject to defender budget β_D. Control m interdicts a set of edges, decreasing the reliability of all edges in the set. The subset $M_{ij} \subseteq M$ contains all controls that interdict edge (i, j). This approach is slightly different than that of the canonical maximum-reliability interdiction problem, where the defender directly interdicts edges on the network. We treat the budget and costs as beliefs, because the attacker may not know their values. We assume a level 0 defender does not choose any controls.

The defender believes that the conditional probability that attacker $A \in \mathcal{A}$ attempts an attack is $\theta_D(A)$. This is assumed to be determined exogenously, possibly from expert opinion or a risk assessment. The defender faces multiple attackers with different levels k' with $0 \le k' \le k-1$, which are captured in the values of $\psi_D^k(i,j,A)$. Each attack follows a fixed path p with a given probability that depends on the defender's level. The defender has belief probabilities $\tilde{\delta}_D(i,j,A)$ and $\delta_D(i,j,A)$ that reflect the reliability of edge (i, j) for attacker A if the edge is or is not interdicted, respectively. We assume that the realizations of these belief probabilities are independent of each other. We further assume that the probabilities $\delta_D(i,j,A)$ and $\tilde{\delta}_D(i,j,A)$ are independent between edges.

To derive the OptDef formulation, note that a level k defender's portfolio interdicts a set of edges $S \subseteq E$. Furthermore, let $P(i)$ be the set of paths leading from the source node 1 to node $i > 1$. Then, the conditional probability of a successful attack is

$$\sum_{A\in\mathcal{A}} \theta_D(A) \sum_{p\in P(n)} \psi_D^k(p,A) \prod_{(i,j)\in p\setminus S} \delta_D(i,j,A) \prod_{(i,j)\in p\cap S} \tilde{\delta}_D(i,j,A). \tag{9}$$

The maximum probability path from any node $i \in N$ to the sink node n does not depend on the path used to reach i. Therefore, we can rewrite (9) as

$$\sum_{A\in\mathcal{A}} \theta_D(A) \sum_{p\in P(n)} \prod_{(i,j)\in p} \psi_D^k(i,j,A) \left(\delta_D(i,j,A)(1-x_{ij}) + \tilde{\delta}_D(i,j,A)x_{ij}\right). \tag{10}$$

The defender believes that the probability that attacker A attempts to traverse edge $(j, \ell) \in E$ is $y_{ij}(A)$ if the edge is not interdicted or $\tilde{y}_{j\ell}(A)$ if it is interdicted, which are decision variables in the defender's model. We define $y_{j\ell}(A)$ and $\tilde{y}_{j\ell}(A)$ recursively as

$$\begin{aligned} y_{j\ell}(A) &= (1-x_{j\ell})\psi_D^k(j,\ell,A) \sum_{(i,j)\in E_j^-} \left(\delta_D(i,j,A)y_{ij}(A) + \tilde{\delta}_D(i,j,A)\tilde{y}_{ij}(A)\right), \\ \tilde{y}_{j\ell}(A) &= x_{j\ell}\psi_D^k(j,\ell,A) \sum_{(i,j)\in E_j^-} \left(\delta_D(i,j,A)y_{ij}(A) + \tilde{\delta}_D(i,j,A)\tilde{y}_{ij}(A)\right). \end{aligned}$$

Gathering these equations, we introduce the OptDef formulation:

$$z_D^k = 1 - \min \sum_{A\in\mathcal{A}} \theta_D(A) \sum_{(i,n)\in E_n^-} \left(\delta_D(i,n,A)y_{in}(A) + \tilde{\delta}_D(i,n,A)\tilde{y}_{in}(A)\right) \tag{11}$$

$$\text{s.t. } y_{1i}(A) + \tilde{y}_{1i}(A) \ge \psi_D^k(1,i,A), \qquad \forall A \in \mathcal{A}, (1,i) \in E_1^+, \tag{12}$$

$$y_{j\ell}(A) + \tilde{y}_{j\ell}(A) \ge \psi_D^k(j,\ell,A) \sum_{(i,j)\in E_j^-} \left(\delta_D(i,j,A)y_{ij}(A) + \tilde{\delta}_D(i,j,A)\tilde{y}_{ij}(A)\right), \qquad \forall A \in \mathcal{A}, j \in N \setminus \{1,n\}, (j,\ell) \in E_j^+, \tag{13}$$

$$\tilde{y}_{ij}(A) \le \sum_{m\in M_{ij}} w_m, \qquad \forall A \in \mathcal{A}, (i,j) \in E, \tag{14}$$

$$\sum_{m\in M} \kappa_D(m) w_m \le \beta_D, \tag{15}$$

$$y_{ij}(A), \tilde{y}_{ij}(A) \ge 0, \qquad \forall A \in \mathcal{A}, (i,j) \in E, \tag{16}$$

$$w_m \in \{0,1\}, \qquad \forall m \in M. \tag{17}$$

A level k defender maximizes the conditional probability that an attack is prevented (11), which is equivalent to minimizing the conditional probability that an attack succeeds, and which occurs when an attacker traverses the network. Constraint set (12) enforces that each attacker attempts an attack. Constraint set (13) serves as a flow balance equation, determining the probability that an attack from each attacker reaches each edge. Specifically, this is the probability of an attack reaching the edge's starting node multiplied by the conditional probability of the attack then progressing on that edge. Constraint set (14) allows edges to be interdicted only if a control is chosen that interdicts that edge. Constraint (15) allows the defender to choose only as many controls as their budget allows. The last two sets of constraints, (16) and (17), ensure that the appropriate variables are nonnegative and binary.

A solution to the level k defender's problem informs the level $k+1$ attackers' beliefs regarding the defender's edge reliabilities. In particular, Ω_A^k is defined as the set $\{m \in M : w_m = 1\}$, and for each $(i,j) \in E$,

$$\delta_A(i,j|\Omega_A^k) = \begin{cases} \delta_D(i,j,A) & \text{if } x_{ij} = 0, \\ \tilde{\delta}_D(i,j,A) & \text{if } x_{ij} = 1. \end{cases} \tag{18}$$

5. Approximation of the Defender's Problem

It may be computationally difficult to find an optimal solution to OptDef for large-scale problem instances. However, we show that there exists an approximation algorithm that provides a solution with a $1-1/e$ performance guarantee. To do so, we show that OptDef is equivalent to a nonnegative, submodular maximization problem with a knapsack constraint. A heuristic can find solutions that are at least $(1-1/e)$ of the optimal solution in polynomial time for this problem (Sviridenko 2004). A similar result exists for the related problem of maximizing a submodular function subject to a cardinality constraint (Nemhauser et al. 1978). Later, in Section 7, we show that this approximation algorithm often identifies solutions whose values are extremely close to the optimal solution value in practice. This approximation algorithm may be used to decrease the time required to find an optimal solution to OptDef by providing a warm start for the mixed integer programming solver.

We adapt the heuristic from Khuller et al. (1999) to OptDef. We begin by selecting the maximum value set of controls, $S_1 \subset M$, with $|S_1| \le 2$. Next, we enumerate all sets of controls with a cardinality of three, $\mathcal{S}_2$, that satisfy the budgetary constraint, that is, we have $\sum_{m\in S_2} \kappa_D(m) \le \beta_D$ for each set of controls $S_2 \in \mathcal{S}_2$ with $|S_2| = 3$. Then, we greedily complete each of these sets until the budget or available controls are exhausted; that is, for a submodular function f on a set M, we add an element $m^* \in M \setminus S_2$ if it satisfies the budget constraint (i.e., $\kappa_D(m^*) + \sum_{m\in S_2} \kappa_D(m) \le \beta_D$) that satisfies

$$m^* \in \arg\max_{m\in M\setminus S_2} \frac{f(S_2 \cup \{m\}) - f(S_2)}{\kappa_D(m)}. \tag{19}$$

Then, $S_2 \leftarrow S_2 \cup m^*$, and the process is repeated until no new elements can be selected. Let the maximum value set completed this way be S_3. If $f(S_1) \ge f(S_3)$, then the algorithm returns S_1. Otherwise, it returns S_3.

Theorem 1 demonstrates that the conditional probability of a successful attack, the complement of the OptDef objective function value, is a supermodular function given a set of interdicted edges.

Theorem 1. *Let S be a set of interdicted edges $(i,j) \in E$, and let*

$$0 \le \tilde{\delta}_D(i,j,A) \le \delta_D(i,j,A), \ \forall (i,j) \in E, A \in \mathcal{A}.$$

Then, for any path $p \in P$ and attacker $A \in \mathcal{A}$,

$$g(S,p,A) = \prod_{(i,j)\in p\setminus S} \delta_D(i,j,A) \prod_{(i,j)\in p\cap S} \tilde{\delta}_D(i,j,A)$$

is a nonincreasing, supermodular function in S.

Proof. We begin by showing that $g(S,p,A)$ is nonincreasing in S. First, note that $0 \le g(S,p,A) \le 1$, because $g(S,p,A)$ is a product of numbers whose values are between zero and one. For sets of edges $S_1 \subseteq S_2 \subseteq E$, we have

$$\begin{aligned} g(S_2,p,A) &= \prod_{(i,j)\in p\setminus S_2} \delta_D(i,j,A) \prod_{(i,j)\in p\cap S_2} \tilde{\delta}_D(i,j,A) \\ &= \prod_{(i,j)\in p\cap(S_2\setminus S_1)} \frac{\delta_D(i,j,A)\tilde{\delta}_D(i,j,A)}{\delta_D(i,j,A)\tilde{\delta}_D(i,j,A)} \\ &\quad \prod_{(i,j)\in p\setminus S_2} \delta_D(i,j,A) \prod_{(i,j)\in p\cap S_2} \tilde{\delta}_D(i,j,A) \\ &= \prod_{(i,j)\in p\cap(S_2\setminus S_1)} \frac{\tilde{\delta}_D(i,j,A)}{\delta_D(i,j,A)} \prod_{(i,j)\in p\setminus S_1} \delta_D(i,j,A) \\ &\quad \prod_{(i,j)\in p\cap S_1} \tilde{\delta}_D(i,j,A) \\ &= \left(\prod_{(i,j)\in p\cap(S_2\setminus S_1)} \frac{\tilde{\delta}_D(i,j,A)}{\delta_D(i,j,A)} \right) g(S_1,p,A). \end{aligned} \tag{20}$$

Because $0 \le \tilde{\delta}_D(i,j,A) \le \delta_D(i,j,A)$, $\forall (i,j) \in E$, (20) shows that $g(S,p,A)$ is nonincreasing in S.

We now prove that $g(S,p,A)$ is supermodular by showing for all $(i_1,j_1),(i_2,j_2) \in E$ that

$$g(S,p,A) + g(S \cup \{(i_1,j_1),(i_2,j_2)\},p,A) - g(S \cup \{(i_1,j_1)\},p,A) - g(S \cup \{(i_2,j_2)\},p,A) \ge 0. \quad (21)$$

We use (20) to algebraically simplify the left-hand side of (21):

$$\begin{aligned} &g(S,p,A) + g(S \cup \{(i_1,j_1),(i_2,j_2)\},p,A) - g(S \cup \{(i_1,j_1)\},p,A) - g(S \cup \{(i_2,j_2)\},p,A) \\ &= g(S,p,A)\left(1 + \frac{\tilde{\delta}_D(i_1,j_1,A)\tilde{\delta}_D(i_2,j_2,A)}{\delta_D(i_1,j_1,A)\delta_D(i_2,j_2,A)} - \frac{\tilde{\delta}_D(i_1,j_1,A)}{\delta_D(i_1,j_1,A)} - \frac{\tilde{\delta}_D(i_2,j_2,A)}{\delta_D(i_2,j_2,A)}\right) \\ &= g(S,p,A)\left(1 - \frac{\tilde{\delta}_D(i_2,j_2,A)}{\delta_D(i_2,j_2,A)}\right)\left(1 - \frac{\tilde{\delta}_D(i_1,j_1,A)}{\delta_D(i_1,j_1,A)}\right). \end{aligned} \quad (22)$$

Because $0 \le \tilde{\delta}_D(i,j,A) \le \delta_D(i,j,A)$, $\forall (i,j) \in E$, we have the following three inequalities:

$$g(S,p,A) \ge 0,$$
$$1 - \frac{\tilde{\delta}_D(i_2,j_2,A)}{\delta_D(i_2,j_2,A)} \ge 0,$$
$$1 - \frac{\tilde{\delta}_D(i_1,j_1,A)}{\delta_D(i_1,j_1,A)} \ge 0.$$

Combining these three inequalities with (22) yields (21). Therefore, $g(S,p,A)$ is a nonincreasing supermodular function. □

The consequence of Theorem 1 is that the OptDef objective function is a nonnegative, submodular function. Next, we prove that OptDef is a nonnegative, submodular maximization problem subject to a knapsack constraint.

Theorem 2. *OptDef is a nonnegative, submodular maximization problem subject to a knapsack constraint.*

Proof. We begin by reformulating (10), which is equivalent to the complement of the OptDef objective function the objective (11) and Constraints (12) and (13). Then the probability that an attack is prevented, using the definition of $g(S,p,A)$ from Theorem 1, is

$$1 - \sum_{A \in \mathcal{A}} \theta_D(A) \sum_{p \in P(n)} g(S,p,A) \prod_{(i,j) \in p} \psi_D(i,j,A), \quad (23)$$

where S is the set of edges interdicted by the defender's portfolio. Note that $\prod_{(i,j) \in p} \psi_D(i,j,A)$ is a constant in this formulation. The nonnegative weighted sum of supermodular functions is also supermodular, yielding a function in (23) that is submodular after multiplying by –1 and subtracting the term from one. The resulting value is between zero and one, because it represents a probability.

The remaining constraints of OptDef, Constraints (14) and (15), simply define a knapsack constraint on the set of controls. Because S is nondecreasing in the controls, this is equivalent to a knapsack constraint on S. Hence, we have the desired result. □

There are two implications of Theorem 2. The first is that the algorithm introduced earlier in this section from Khuller et al. (1999) identifies solutions with a guaranteed $1 - 1/e$ approximation ratio. The second implication is that a more computationally efficient greedy algorithm (also introduced by Khuller et al. 1999) can be used to identify solutions with an approximation ratio of $1 - 1/\sqrt{e}$. The greedy algorithm starts with an empty set of controls and greedily completes this set according to (19) until the budget or available controls are exhausted. Then it compares this greedily chosen set with the maximum value single-element set and chooses the one with the higher objective function value.

6. Iterative ARA Algorithm

We formally introduce an iterative algorithm for solving OptAtt and OptDef for the attacker and defender problems across all values of k:

1. Initialization: Provide data.
   - Some sets and beliefs given in Table 1 must be provided, including the level (K) of defender, set of attackers ($\mathcal{A}$), sets of nodes (N) and edges (E), set of controls (M), reliabilities of edges ($\delta_D(i,j,A)$, $\tilde{\delta}_D(i,j,A)$ $\forall (i,j) \in E$, $A \in \mathcal{A}$), defender budget (β_D), costs of controls ($\kappa_D(m)$ $\forall m \in M$), and distribution of attackers ($\theta_D(A)$ $\forall A \in \mathcal{A}$).
2. Identify level 0 attacker and defender solutions.
   - We assume a level 0 attacker A takes a greedily formed path p as described in Section 4.1. This yields $\psi_D^{k'}(i,j,A) = 1$ if $(i,j) \in p$ and $\psi_D^{k'}(i,j,A) = 0$ if $(i,j) \notin p$ for all defenders $k' > 0$ who believe that attacker A is at level 0.

- We assume a level 0 defender does not choose any controls, so that the level 0 defender's solution is $\Omega_A^0 = \emptyset$ with $\delta_A^1(i,j|\Omega_A^0) = \delta_D(i,j,A)$ for each attacker $A \in \mathcal{A}$.

3. For $k = 1, 2, \ldots, K-1$:
   - Solve OptAtt using Dijkstra's algorithm: For each level k attacker A, we solve OptAtt, given by (1)–(7), to obtain the attack path p using Dijkstra's algorithm as described in Section 4.1. Use the values of u_{ij} to set $\psi_D^{k'}(i,j,A)$, $(i,j) \in E$ for each $k < k' \leq K$ using (8).
   - Solve OptDef for a level k defender. We suggest two ways to do this:
     — Solve OptDef to optimality using the mixed integer programming formulation given by (11)–(17).
     — Identify approximate solutions using the approximation algorithm or the greedy algorithm presented in Section 5 to identify solutions within $(1-1/e)$ or $(1-1/\sqrt{e})$ of the optimal solution value, respectively.

Given a solution to OptDef, let $\Omega_A^k = \{m \in M | w_m = 1\}$. Then set the values of $\delta_A^{k+1}(i,j|\Omega_A^k)$ for all level $k+1$ attackers $A \in \mathcal{A}$ using (18).

4. Solve OptDef a final time to obtain the solution for a level K defender.

To illustrate the algorithm, we revisit the example introduced in Section 1.1. Again, we assume the defenders budget allows them to select only a single control. Next, we iterate through the algorithm to find the OptDef solutions with level 1–4 defenders, respectively. For each level k defender from $k = 1$ to $k = 4$, for the sake of simplicity, we assume the defender is defending against a uniform distribution of the attackers of levels less than k. We report the attack success probability for both models to directly compare the OptAtt and OptDef solutions in Table 2.

Table 2. Example OptAtt and OptDef Solutions for Attackers and Defenders of Varying Levels

	Attacker		Defender	
K	Path selected	OptAtt attack success probability	Mitigation selected	OptDef attack success probability
0	Right	0.102	N/A	N/A
1	Right	0.102	m_3	0.074
2	Left	0.09	m_3	0.074
3	Right	0.102	m_1	0.078
4	—	—	m_2	0.069

We first consider level 0 attackers, who choose the right path using a greedy tactic, with an attack success probability of 0.102. A level 0 defender selects no controls. A level 1 attacker optimizes against the level 0 defender and again chooses the right path for an attack success probability of 0.102, because this is the maximum-reliability path with all uninterdicted edges.

We continue to consider each level of attacker and defender up to the desired defender level. Next, we solve OptDef for a level 1 defender, who optimizes against level 0 attackers. The level 1 defender chooses the control that most decreases the reliability of the level 0 attackers' path, which is control m_3. This attains an OptDef attack success probability of 0.074. The level 2 attacker then selects the left path, yielding an attack success probability of 0.09. A level 2 defender optimizes against both level 0 and level 1 attackers, who both take the right path, and again selects m_3, giving an attack success probability of 0.074. Given this, a level 3 attacker takes the right path with an attack success probability of 0.102.

Next, we consider a level 3 defender, who defends against level 0–2 attackers, two of whom take the right path and one of whom takes the left path. A level 3 defender selects m_1 for an OptDef attack success probability of 0.078. The level 4 defender defends against all level 0–3 attackers, two of whom traverse the left path and two of whom traverse the right path. In this case, m_2 becomes the optimal control choice. This achieves a final OptDef attack success probability of 0.069. In this example we see the defender change their control decisions based on what paths they believe the attackers would take.

Note that the ARA algorithm results in a series of solutions to OptDef that correspond to level $0, 1, \ldots, K$ defenders. Therefore, our approach provides decision makers with a suite of solutions instead of a single, "good" solution, which could be advantageous for defensive planning.

7. Computational Results

In this section, we provide computational results for the models and algorithms based on the information security investment case study introduced by DuBois (2020). In the case study, we generate random data according to a specific parameterized structure that we are able to adjust to leverage control over the size and complexity

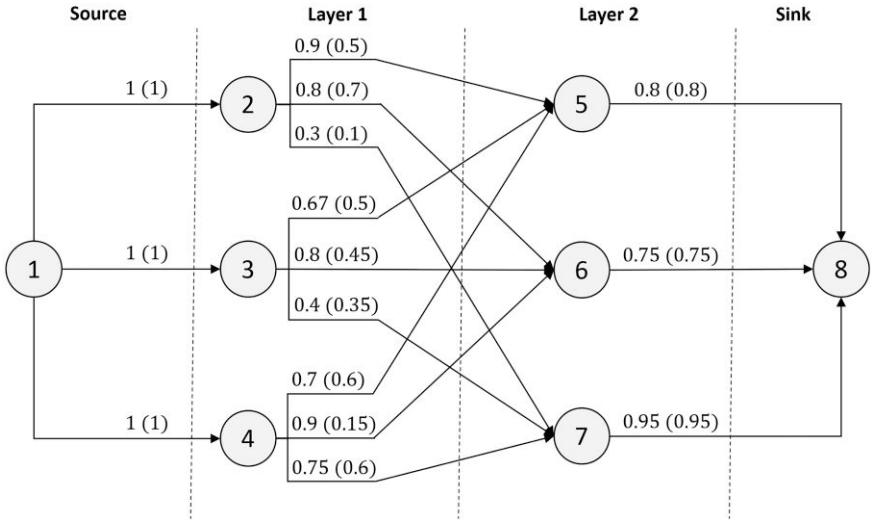
of the problem. We describe how this structure is created as well as the parameters used. Later, we vary the size of the problem instances to assess the performance of the approximation algorithm.

In the case study, each problem instance is given by a network that is organized into a source node, a sink node, and ℓ "layers" of nodes. Each layer of nodes is a set of nodes such that nodes in a layer connect only to nodes in the next layer. For simplicity, we assume that each layer contains the same number of nodes, which results in a structure with many interconnections and paths. Figure 2 illustrates an example of the network under consideration with two layers of three nodes each. A path through the network includes one node from each layer. The source node is connected by noninterdictable edges to each node in the first set, with reliabilities of one. Three edges leave each node in sets $1,\ldots,\ell-1$ to nodes in sets $2,\ldots,\ell$, respectively.

We first assign edges such that each node has both an entering and leaving edge, before randomly assigning the remaining edges. We then assign reliabilities to each of these edges randomly, and each reliability is treated as a constant once it is generated. The uninterdicted reliabilities are uniformly assigned a value between zero and one, that is, $\delta(i,j) \sim U(0,1]$. The interdicted reliabilities are a random (0, 1) proportion of the uninterdicted value, that is, $\tilde{\delta}(i,j) \sim U(0,1] \cdot \delta(i,j)$. Finally, noninterdictible edges connect each node in layer ℓ to the sink node, with reliabilities of $U(0.5,1]$. We choose a number of possible controls and a budget for the defender, and then randomly determine costs for each of these controls using a uniform distribution over some specified cost range. This cost is exactly one for a cardinality budget constraint, and the costs follow a $U[0.5,1.5]$ distribution for a knapsack constraint to maintain an average cost of 1.0 per control in both situations. Each control is then randomly assigned a subset of these edges to interdict. To do so, we first specify a parameter, α, that gives the average proportion of edges covered by any control. If we have a cardinality budget constraint, for each control, we loop through the edges and assign each to the control with a probability of α. If we have a knapsack constraint, we alter the probability of assigning each edge to a control by considering the cost. Let Δ_m be the value attained by subtracting the average cost from the cost of the control m and then dividing by the range of possible costs. In our case, with costs generated from the range $[0.5,1.5]$, this is equivalent to $\Delta_m = \kappa_D(m) - 1$. Then we assign edges to control m with a probability of $\alpha(1+\alpha_2\Delta_m)$, where α_2 represents the amount of effect cost has on mitigation quality.

For our case study, we generated a problem instance for a graph with $\ell = 5$ layers and five nodes per layer. To better visualize the trade-offs between controls, a cardinality constraint was used for the defender's budget, where the defender can choose 4 of 10 controls. Each

Figure 2. Simple Example of Synthetic Network Structure with Two Layers



Note. Example edge reliabilities are shown along each edge for when the edge is not interdicted (interdicted), $\delta_{ij}(i,j,A)(\tilde{\delta}_{ij}(i,j,A))$.

Table 3. Table of Portfolios Selected by the Defender for Differing Levels of k

Level of defender	Controls									
	m_1	m_2	m_3	m_4	m_5	m_6	m_7	m_8	m_9	m_{10}
0	—	—	—	—	—	—	—	—	—	—
1	—	✓	✓	✓	—	—	—	—	—	✓
2	—	✓	✓	✓	—	—	—	—	—	✓
3	—	✓	—	✓	✓	—	—	—	—	✓
4	—	✓	—	✓	✓	—	—	—	—	✓
5	—	✓	—	✓	✓	—	—	—	✓	—
6	—	✓	—	✓	✓	—	—	—	✓	—
7	—	✓	✓	—	✓	—	—	—	✓	—
8	—	✓	✓	—	✓	—	—	—	✓	—
9	—	—	✓	✓	✓	—	—	—	✓	—
10	—	—	✓	✓	✓	—	—	—	✓	—

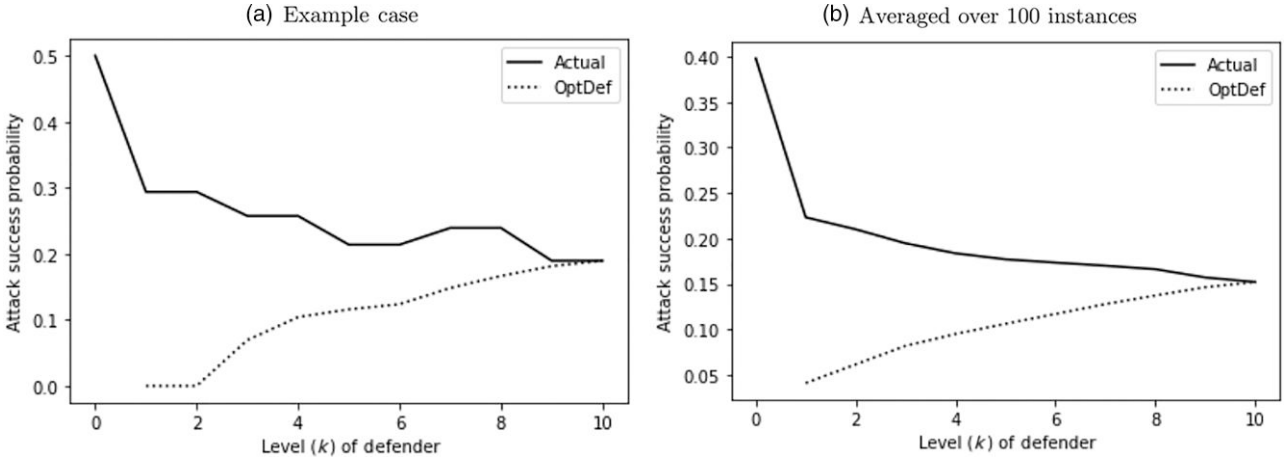
control was randomly generated to cover approximately 15% of the edges ($\alpha = 0.15$). Although a realistic value of the defender's level is 4, we set a maximum defender level of $K = 10$ to study a range of defensive solutions. Each defender level $k \leq K$ assumes a uniform distribution of attacker levels over $0 \leq k' < k$. We refer to DuBois (2020) for additional experiments that consider alternative distributions over the attacker levels.

We solve the case study using the algorithm in Section 6, where for each level of defender we solve OptDef to optimality using Gurobi 9.1.1. This computation was performed on a computer with an Intel(R) Core(TM) i7-8650U central processing unit (CPU) at 1.90 GHz with a 2.11 GHz processor and 16 GB of random access memory, and took 1.36 seconds to run the full algorithm.

The ARA algorithm solves the defender's problem 11 total times across all levels of k, thereby generating various potential defender solutions before arriving at the final solution for the level $K = 10$ defender. Table 3 reports the portfolio of controls chosen by each defender level k from 0 to 10. Note that by assumption, the level 0 defender chooses no controls. Table 3 indicates that some controls are not chosen in any of the defender's solutions, such as m_1, m_6, m_7, and m_8. Other controls work together to provide better protection against different or more of a variety of paths the attackers may take (e.g., controls m_2, m_3, and m_4). Controls m_5 are m_9 are chosen by defenders whose levels are at least 3 and 5, respectively, indicating that some controls only become attractive by more strategic defenders (who believe they face more strategic attackers).

Next, we examine the *attack success probability*, the average probability of a successful attack and the complement to the OptDef objective function value. We report the attack success probability for defenders with levels 0–9 given a uniform distribution of level 0–9 attackers. Figure 3 reports the attack success probability from both the defender's and attackers' points of view. First, it reports the attack success probability corresponding to OptDef (the dotted line), which captures the attack success probability based on the defender's

Figure 3. Two Lines Showing the Average Attack Success of Attackers Levels 0–9 Against Levels $k < 10$ Defenders and the OptDef Value for Level $k < 10$ Defenders (Attack Success of a Uniform Distribution of Attackers Levels 0 to k)



belief regarding the attackers. Second, it reports the actual, retrospective attack success probability for the same set of attackers based on their actual levels of strategic sophistication (the solid line). These values differ because defenders with levels lower than 10 are incapable of perceiving level 9 (or higher) attackers' true levels of strategic sophistication.

Figure 3(a) illustrates the results across a single problem instance, and Figure 3(b) illustrates the results averaged over 100 problem instances of the same size as the example problem. The attack success probability associated with OptDef is increasing in k in both figures. This occurs because more strategic defenders select controls for more strategic attackers who are better able to evade defenses. We observe that the actual attack success probability is higher than that associated with OptDef, because defenders with levels $k < 10$ believe that some attackers are less strategic than they are in actuality, resulting in an inaccurate believed attack success probability. Both lines converge to the same point when the defender's beliefs match reality, which in our example occurs when the defender is at level $k = 10$. Figure 3(a) indicates that the actual attack success probability does not always decrease with k for any particular problem instance. This effect is due to attackers with higher levels of k. Because the defender does not accurately assume the distribution of attackers, it is possible that attackers can find more reliable paths against a sophisticated defender.

Figure 3(b) illustrates the attack success probability associated with OptDef and for the actual values averaged across 100 randomly generated problem instances of the same size. The overall trends in Figure 3(b) are similar to those in Figure 3(a), with Figure 3(b) showing that the average actual attack success probability monotonically decreases with k. This suggests that the solutions provided by OptDef, on average, provide better defenses for more strategic defenders.

Next, we consider the effects of the defender's portfolio of controls and how it performs against attackers with different levels of strategic sophistication. As before, we consider defenders of levels 0 through 9. We study how two subsets of attackers perform against these defenders: low level (levels 0–4) and high level (levels 5–9). We also report all attackers (levels 0–9) for comparison. Note that the case with level 0–9 attackers matches that considered in Figure 3. In each case, the attackers' actual levels follow a discrete uniform distribution as before. Each defender assumes that the levels of the attackers are uniformly distributed across all levels k lower than that of the defender. The level 5 defender believes they are defending against low-level 0–4 attackers, and a level 10 defender believes they are defending against level 0–9 attackers. The defender's beliefs matches the actual attacker levels in both of these cases; however, there is a mismatch between the defender's beliefs and the actual levels of the attackers in all other cases, where the defender either overestimates or underestimates the attackers' levels to varying degrees.

Figure 4. Attacker Success Probabilities Against Defenders of Levels 0 through 9 for Three Uniform Distributions of Attackers: Low Level (0–4), High Level (5–9), and Both (0–9)

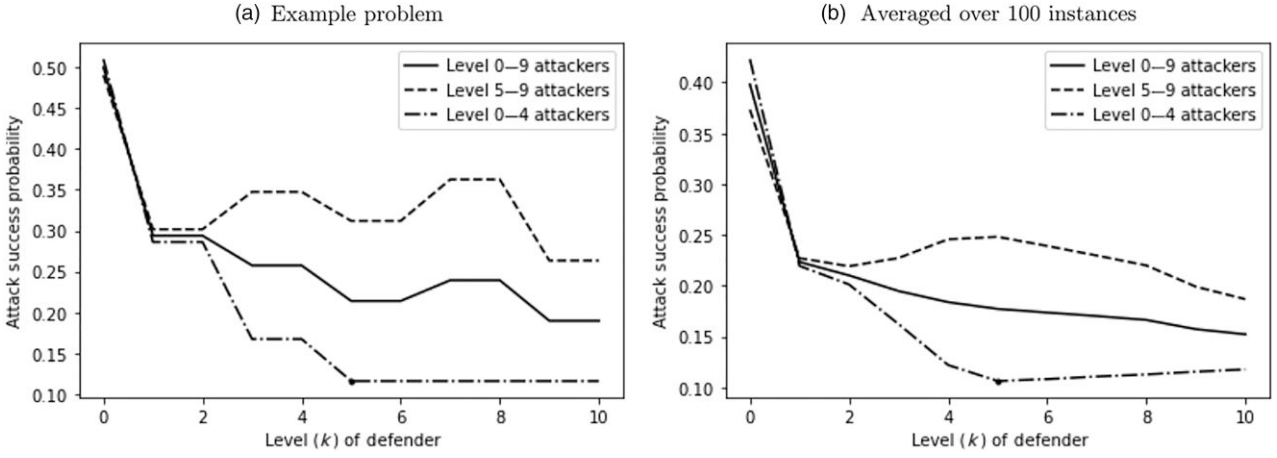


Figure 4 illustrates the actual attack success probability for the defender across the three groups of attackers, with Figure 4(a) illustrating the actual attack success probability for the example problem and Figure 4(b) illustrating the actual attack success probability averaged across 100 problem instances. In both figures, higher level attackers tend to be more successful than lower level attackers across all defender levels, which is expected. We observe that that higher level defenders tend to achieve a lower attack success probability when facing level 0–4 attackers, which suggests that defenses may be more effective against less strategic attackers, even when the defender overestimates the attackers' levels (which occurs for level 6–9 defenders).

In contrast, level 5–9 attackers often achieve high attack success probabilities. In Figure 4(a), the attack success probability of level 5–9 attackers is highest against level 7 and 8 defenders. This is surprising and occurs because the attackers whose levels are higher than the defender's can find paths that the defender does not defend well. This indicates that the defenses chosen by more sophisticated defenders may sometimes perform worse against sophisticated attackers than the defenses chosen by less sophisticated defenders.

Figure 4(b) presents results averaged across 100 randomly generated problem instances. Overall, we see a decreasing trend in the attack success probability across all level 0–9 attackers and low-level 0–4 attackers as the defender's level increases. The decreasing attack success probability is more pronounced for level 0–4 attackers. This occurs because the defender increasingly assumes a more accurate distribution of the attackers as the defender level increases to 5. After level 5, the defender continues to defend well across level 0–4 attackers and mostly improves against level 5–9 attackers. The actual attack success probability against the high-level 5–9 attackers does not monotonically decrease with the defender's level (i.e., it increases between defender levels 2 to 5). This is counterintuitive, and results from the defender's beliefs. A level $k = 5$ defender, for example, assumes they face level 0–4 attackers and therefore optimizes defenses for less strategic attackers. As the defender's level increases to $k = 9$, the defender becomes more effective against the most strategic attackers. Overall, this example suggests that is better for the defender to overestimate the level of the attackers rather than to underestimate the level of the attackers.

To further analyze the effects of the defender misjudging the sophistication of the attackers, we consider a defender of level $k = 10$ who uniformly overestimates or underestimates the attackers' levels by a certain *offset*. As before, we consider a uniform distribution of attackers whose true levels are 0 through 9. Positive offsets correspond to the defender overestimating the attackers' levels, and negative offsets correspond to underestimating the attackers' levels. When overestimating, the defender cannot perceive attackers of a higher level than them, and thus they assume any such attackers are one level below them in OptDef. Likewise, when underestimating, the defender always assumes the attackers have a level of at least 0. A defender with an offset of -3, for example, only perceives level 0 through 6 attackers after capping the perceived attacker levels below at 0.

Figure 5 depicts the actual attack success probability averaged across 100 problem instances as a function of the offset. We observe that both over- and underestimating the distribution of the attackers leads to higher attack success probabilities in comparison with the case when the defender perceives the true attacker levels (shown by the offset of zero), with underestimating the attackers being slightly worse than overestimating. This observation is consistent with Figure 4. This suggests that it is ideal for the defender to "get it right" and correctly model attackers, and that there is a benefit to erring on the side of planning for sophisticated attackers.

We next assess the quality of the solutions identified by the greedy algorithm (see Section 5), which is guaranteed

Figure 5. Attack Success of a Uniform Distribution of Attackers Against Level 10 Defenders Who Believe All Attackers' Levels Are Offset by the Amount Given on the x-Axis

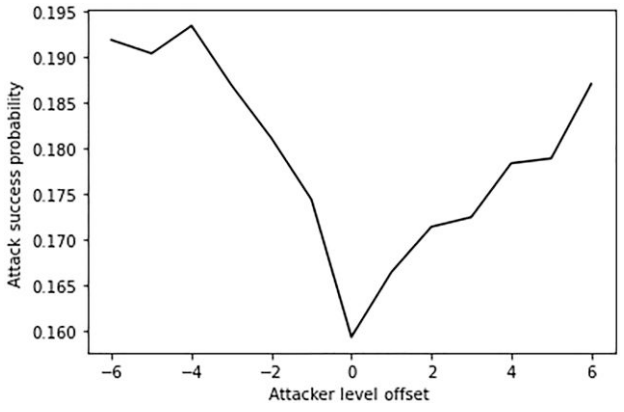


Table 4. Solution Times for Various Instances of the Defender's Problem Subject to a Knapsack Constraint and the Relative Accuracy of the Greedy Algorithm

Layers ℓ	Nodes per layer, $\lvert N_\ell \rvert$	Edges $\lvert E \rvert$	Controls $\lvert M \rvert$	Budget β	OptDef time (s)	Greedy time (s)	Gap z_h/z_D^*
5	15	930	10	5	3.07	0.078	0.983
5	15	930	20	10	2.62	0.236	0.998
10	10	920	10	5	2.47	0.237	0.999
10	10	920	20	10	2.26	0.277	0.999
10	15	2,055	12	6	19.9	0.998	0.997
10	15	2,055	24	12	6.99	1.07	0.999
10	20	3,640	14	7	49.7	2.25	0.998
10	20	3,640	30	15	19.1	3.24	1.000
10	25	5,675	16	8	66.2	2.12	0.999
10	25	5,675	34	17	63.7	6.13	1.000
15	5	360	10	5	0.724	0.075	0.999
15	5	360	20	10	1.16	0.337	1.000
15	10	1,420	12	6	8.40	0.328	0.997
15	10	1,420	24	12	7.16	1.13	1.000
15	15	3,180	14	7	24.0	1.13	0.993
15	15	3,180	30	15	22.2	2.48	1.000
15	20	3,000	16	8	22.1	1.15	1.000
15	20	3,000	34	17	18.8	3.75	1.000
15	20	5,640	16	8	68.1	2.36	1.000
15	20	5,640	34	17	60.3	8.27	1.000
15	25	4,000	18	9	36.7	2.22	0.999
15	25	4,000	40	20	41.6	9.74	1.000
15	25	8,800	18	9	157.6	4.15	1.000
15	25	8,800	40	20	159.1	17.8	1.000
20	10	1,920	14	7	12.0	0.717	1.000
20	10	1,920	30	15	10.3	2.81	1.000
20	15	4,000	16	8	38.7	1.33	1.000
20	15	4,000	34	17	34.2	6.66	1.000
20	15	4,305	16	8	44.8	1.62	1.000
20	15	4,305	34	17	41.8	6.17	1.000
20	20	4,000	18	9	12.1	0.637	1.000
20	20	4,000	40	20	32.2	7.56	1.000
20	20	7,640	18	9	124.6	3.54	1.000
20	20	7,640	40	20	107.3	13.8	1.000
20	25	5,000	20	10	22.7	1.09	1.000
20	25	5,000	44	22	47.6	11.9	1.000
20	25	11,925	20	10	95.1	2.83	1.000
20	25	11,925	44	22	238.3	28.6	1.000
25	10	2,420	16	8	5.96	0.459	1.000
25	10	2,420	34	17	15.0	3.29	1.000

to identify solutions whose objective function values are at least $1 - 1/\sqrt{e}$ of the optimal OptDef solution values. We consider randomly generated instances for a level 10 defender, each using a knapsack constraint for the budget. We compare the greedy and exact algorithms on the same problem instances with a level 10 defender as follows by first using the greedy algorithm to identify near-optimal OptDef solutions for defenders of levels 1–9. Then, we use either an exact algorithm or the greedy algorithm for the level 10 defender's problem instance.

We randomly generated 40 problem instances with varying sizes to evaluate the solution quality of the greedy algorithm. Table 4 reports the parameters used to generate each problem instance as well as the CPU times associated with the exact (OptDef) and greedy algorithms. The CPU time (in seconds) to solve OptDef to optimality includes the setup time of writing the variables and constraints, as well as the solver time. Likewise, the time to implement the greedy algorithm includes the time to read the inputs and execute the algorithm. Table 4 reports the ratio of the greedy solution value (z_h) and the

optimal solution value to OptDef (z_D^*). The results indicate that the greedy solution is within 1.7% of the optimal solution values across all problem instances, and the greedy algorithm identifies the optimal solution in 28 of the 40 problem instances. Gurobi takes considerably longer to load and solve the problem instances than the greedy algorithm for larger problem instances, which provides an incentive for the greedy algorithm's use.

8. Conclusion

In this paper, we introduce new models and algorithms for identifying a portfolio of security controls to deploy when considering multiple adaptive adversaries of varying levels of strategic sophistication. To inform these investments, our approach extends an ARA framework to consider a maximum-reliability path interdiction problem with a single defender and multiple attackers. All players are assumed to be boundedly rational, and we allow for uncertainty regarding system and player information. We present mixed integer programming formulations of the attacker and defender problems, and we introduce an ARA algorithm to iteratively solve the models. We also introduce an approximation algorithm that identifies near-optimal solutions to the defender's problem with a guaranteed $1 - 1/e$ approximation ratio.

The solutions provide insight into investments that construct a layered security defense and that perform well against many adversaries, including some adversaries who are not strategic. One practical benefit of the proposed methodology is that it yields a suite of investment solutions, instead of a single solution, which can aid decision makers. Another benefit of the modeling approach is that it allows for the consideration of nonstrategic attackers, which could be used to model risks arises from nature, although we did not specifically consider the impact of natural disasters (Zhuang and Bier 2007). New vulnerabilities appear regularly, and, therefore, organizations should proactively perform risk assessments to inform defensive investments and decision making on a regular basis. The approach introduced in this paper seeks to help with these decisions.

We illustrate the models and solution techniques on a case study. The framework indicates that the defensive strategies change along with the defender's level of strategic sophistication as well as those of the attackers. The results identify security controls that are effective across a range of adversarial assumptions. The solutions tend to defend better against less strategic attackers than more strategic defenders. Additionally, the results suggest that it may be better for the defender to overestimate rather than underestimate the strategic sophistication of the attackers.

There are several avenues for future research. First, the ARA approach with boundedly rational players can be extended to include other network interdiction models aside from the maximum-reliability path interdiction problem. Second, model extensions could accommodate attackers with less knowledge of the network than the defender, possibly by eliminating portions of the network that the attacker does not know about when solving the attackers' formulation. Third, the ARA approach could be extended to balance the goal of security with system performance depending on the application under consideration.

Acknowledgments

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation. The authors thank the anonymous reviewers, whose suggestions for improvement led to a substantially improved manuscript.

References

Ahuja R, Magnanti T, Orlin J (1993) *Network Flows: Theory, Algorithms, and Applications*, 1st ed. (Prentice-Hall, Hoboken, NJ).

Albert LA, Nikolaev A, Jacobson SH (2023) Homeland security research opportunities. *IISE Trans.* 55(1):22–31.

Banks D, Gallego V, Naveiro R, Ríos Insua D (2020) Adversarial risk analysis: An overview. *WIREs Comput. Statist.* 14(1):e1530.

Bistarelli S, Fioravanti F, Peretti P (2006) Defense trees for economic evaluation of security investments. *Proc. First Internat. Conf. Availability, Reliability Security* (Institute of Electrical and Electronics Engineers, Piscataway, NJ), 416–423.

Caballero WN, Friend M, Blasch E (2021) Adversarial machine learning and adversarial risk analysis in multi-source command and control. *Proc. Signal Processing, Sensor/Information Fusion, Target Recognition XXX*, vol. 11756L (International Society for Optics and Photonics, Bellingham, WA), 98–108.

Camerer CF, Ho TH, Chong JK (2004) A cognitive hierarchy model of games. *Quart. J. Econom.* 119(3):861–898.

Cano J, Pollini A, Falciani L, Turhan U (2016) Modeling current and emerging threats in the airport domain through adversarial risk analysis. *J. Risk Res.* 19(7):894–912.

Cavusoglu H, Raghunathan S, Yue WT (2008) Decision-theoretic and game-theoretic approaches to IT security investment. *J. Management Inform. Systems* 25(2):281–304.

Council of Economic Advisors (2018) The cost of malicious cyber activity to the U.S. economy. Report, The White House, Washington, DC.

DuBois E (2020) Optimizations models with applications to homeland security systems. Unpublished PhD thesis, University of Wisconsin–Madison, Madison.

Fielder A, Panaousis E, Malacaria P, Hankin C, Smeraldi F (2016) Decision support approaches for cyber security investment. *Decision Support Systems* 86(June):13–23.

Hubbard DW, Seiersen R (2016) *How to Measure Anything in Cybersecurity Risk* (John Wiley & Sons, Hoboken, NJ).

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

Joshi C, Aliaga JR, Insua DR (2020) Insider threat modeling: An adversarial risk analysis approach. *IEEE Trans. Inform. Forensics Security* 16:1131–1142.

Khuller S, Moss A, Naor JS (1999) The budgeted maximum coverage problem. *Inform. Processing Lett.* 70(1):39–45.

Knowles W, Prince D, Hutchison D, Disso JFP, Jones K (2015) A survey of cyber security management in industrial control systems. *Internat. J. Critical Infrastructure Protection* 9(June):52–80.

Kruse CS, Frederick B, Jacobson T, Monticone DK (2017) Cybersecurity in healthcare: A systematic review of modern threats and trends. *Tech. Health Care* 25(1):1–10.

Lallie HS, Debattista K, Bal J (2020) A review of attack graph and attack tree visual syntax in cyber security. *Comput. Sci. Rev.* 35(C):100219.

Lee R, Wolpert DH (2012) Game theoretic modeling of pilot behavior during mid-air encounters. Guy TV, Kárný M, Wolpert DH, eds. *Decision Making with Imperfect Decision Makers.* Intelligent Systems Reference Library, vol. 28 (Springer, Berlin), 75–111.

Letchford J, Vorobeychik Y (2013) Optimal interdiction of attack plans. Ito, Jonker, Gini, and Shehory, eds. *Proc. 12th Internat. Conf. Autonomous Agents Multiagent Systems* (International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC).

Morton DP, Pan F, Saeger KJ (2007) Models for nuclear smuggling interdiction. *IIE Trans.* 39(1):3–14.

Nandi AK, Medal HR, Vadlamani S (2016) Interdicting attack graphs to protect organizations from cyber attacks: A bi-level defender–attacker model. *Comput. Oper. Res.* 75(November):118–131.

National Institute of Standards and Technology (2018) Framework for improving critical infrastructure cybersecurity. Report, National Institute of Standards and Technology, Washington, DC.

Nemhauser GL, Wolsey LA, Fisher ML (1978) An analysis of approximations for maximizing submodular set functions—I. *Math. Programming* 14(1):265–294.

Rios J, Rios Insua D (2012) Adversarial risk analysis for counterterrorism modeling. *Risk Anal.* 32(5):894–915.

Rios Insua D, Couce-Vieira A, Rubio JA, Pieters W, Labunets K, Rasines DG (2021) An adversarial risk analysis framework for cybersecurity. *Risk Anal.* 41(1):16–36.

Ross R, Pillitteri V, Dempsey K, Riddle M, Guissanie G (2021) Protecting controlled unclassified information in nonfederal systems and organizations. Special Publication 800-171v2, National Institute of Standards and Technology, Washington, DC.

Rothschild C, McLay L, Guikema S (2012) Adversarial risk analysis with incomplete information: A level-*k* approach. *Risk Anal.* 32(7):1219–1231.

Salmerón J (2012) Deception tactics for network interdiction: A multiobjective approach. *Networks* 60(1):45–58.

Scheibehenne B, Greifeneder R, Todd PM (2010) Can there ever be too many options? A meta-analytic review of choice overload. *J. Consumer Res.* 37(3):409–425.

Schneier B (1999) Attack trees: Modeling security threats. *Dr. Dobbs J.: Software Tools Professional Programmer* 24(12):21–29.

Singhal A, Ou X (2017) Security risk analysis of enterprise networks using probabilistic attack graphs. Wang L, Jajodia S, Singhal A, Singhal A, Ou X, eds. *Network Security Metrics* (Springer, Heidelberg, Germany), 53–73.

Smith JC, Song Y (2020) A survey of network interdiction models and algorithms. *Eur. J. Oper. Res.* 283(3):797–811.

Stahl DO, Wilson PW (1995) On players' models of other players: Theory and experimental evidence. *Games Econom. Behav.* 10(1):218–254.

Stevens R, Dykstra J, Everette WK, Chapman J, Bladow G, Farmer A, Halliday K, Mazurek ML (2020) Compliance cautions: Investigating security issues associated with us digital-security standards. *Network Distributed Systems Security Sympos.*, San Diego, CA.

Sviridenko M (2004) A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.* 32(1):41–43.

Wang S, Banks D (2011) Network routing for insurgency: An adversarial risk analysis framework. *Naval Res. Logist.* 58(6):595–607.

Wang W, Lu Z (2013) Cyber security in the smart grid: Survey and challenges. *Comput. Networks* 57(5):1344–1371.

Wang W, Di Maio F, Zio E (2019) Adversarial risk analysis to allocate optimal defense resources for protecting cyber–physical systems from cyber attacks. *Risk Anal.* 39(12):2766–2785.

Zhang J, Zhuang J, Behlendorf B (2018) Stochastic shortest path network interdiction with a case study of Arizona-Mexico border. *Reliability Engrg. System Safety* 179(November):62–73.

Zheng K, Albert LA (2019a) Interdiction models for delaying adversarial attacks against critical information technology infrastructure. *Naval Res. Logist.* 66(5):411–429.

Zheng K, Albert LA (2019b) A robust approach for mitigating risks in cyber supply chains. *Risk Anal.* 39(9):2076–2092.

Zheng K, Albert LA, Luedtke JR, Towle E (2019) A budgeted maximum multiple coverage model for cybersecurity planning and management. *IISE Trans.* 51(12):1303–1317.

Zhuang J, Bier VM (2007) Balancing terrorism and natural disasters—Defensive strategy with endogenous attacker effort. *Oper. Res.* 55(5):976–991.

Eric DuBois works at CNA and is interested in applications of homeland security. He earned his PhD in industrial engineering in 2020 from the University of Wisconsin–Madison.

Ashley Peper is a PhD student at the University of Wisconsin–Madison. Her research interests are in the area of mathematical optimization with application to security problems.

Laura A. Albert is a professor and the David Gustafson Department Chair of Industrial and Systems Engineering at the University of Wisconsin–Madison. Her research interests are in the areas of optimization and analytics with application to homeland security, emergency response, and public sector problems. She is a fellow of the American Association for the Advancement of Science and a recipient of the Institute of Industrial and Systems Engineers Fellow Award. She is the author of the blog *Punk Rock Operations Research*.