Network Topology Vulnerability/Cost Tradeoff: Model, Application, and Computational Complexity

ARON LASZKA
Vanderbilt University
ASSANE GUEYE
University of Maryland, College Park

Abstract

Technological networks (e.g. telephone and sensor networks, Internet) have provided modern society with increased efficiency, but have also exposed us to the risks posed by their vulnerability to attacks. Mitigating these risks involves designing robust network topologies in situations where resources are economically constrained. In this paper, we consider the vulnerability of network topologies from an economic viewpoint and propose security metrics, which are necessary for assessing the efficiency of our solutions. We define the vulnerability of a network as the potential loss in connectivity due to the actions of a strategic adversary. To derive vulnerability metrics, we revisit our recently introduced network blocking game models, which provide a framework for quantifying network topology vulnerability in adversarial environments. We assume that the network operator takes both security and economic goals into consideration. To model these goals, we generalize previous models by introducing usage costs and budget constraints for the operator. We study two natural constraint formulations, the maximum and the expected cost constraints, and derive the feasible vulnerability/cost region. Since the proposed metrics are based on game-theoretic models, computing them can be challenging. To elucidate these challenges, we provide a thorough complexity analysis for solving the proposed games.

Keywords Network topology robustness, robustness metrics, game theory, blocking games, computational complexity

1 Introduction

The security of networks and systems continues to grow in importance as new threats are emerging every day and attackers are becoming more and more sophisticated. Consequently, achieving perfect security is in practice technically impossible and/or economically impractical. Thus, the priority should be in designing good security solutions that are not only technically viable, but also economically cost effective. These, on the other hand, require defining security

metrics, which are necessary to assess networks' overall level of security and to quantify the set of feasible security/cost operating points and the corresponding Pareto optimal frontier representing the best achievable security/cost tradeoff.

In this paper, we propose a framework for deriving such metrics and finding the feasible vulnerability/cost tradeoff region. First, we revisit our recently introduced network blocking game (NBG) models. These models provide a framework for deriving metrics for the vulnerability of network topologies in adversarial environments. Second, we introduce costs and budget constraints and combine them with the proposed metrics to draw the optimal[1] vulnerability/cost tradeoff curve. Since the proposed metrics are derived from game-theoretic models, understanding the complexity of solving such games is of key importance. In this paper, we thoroughly analyze the complexity of computing such metrics.

The vulnerability of a network is defined as the potential loss in connectivity due to the action of a strategic adversary who tries to disrupt the network connectivity by attacking some resources. One of the main challenges to finding vulnerability/robustness metrics for network topologies resides in quantifying the robustness of a network in the presence of such a strategic attacker, who might exploit the structure of the network topology to design harmful attacks. This is to be distinguished from the *complementary* and more conventional reliability analysis, where failures result from random events such as natural disasters, human errors, etc.

Quantifying the robustness or, equivalently, the vulnerability of topologies has been extensively studied [11, 12, 14, 16, 22, 23, 37, 38, 39]; however, the simultaneous and strategic decision making of the defender and the adversary, which is key to the security of information systems, has received only little attention. For a discussion of previously proposed robustness metrics, see Section 6. To study strategic decision making, game-theoretic models have been gaining a lot of interest in the security community. In game-theoretic approaches, the security problem is modeled as a game and the equilibria are analyzed to predict each player's action.

Recently, *network blocking games* (NBGs) have been introduced and applied to the analysis of the robustness of network topologies in adversarial environments [18, 19, 20, 28, 27]. An NBG takes the *communication model* and the *topology* of a network as inputs, and casts the strategic interactions between an adversary and a defender, called the network operator, as a two-player game. The operator chooses a set of network resources (e.g., links and nodes) as the communication infrastructure, while the adversary targets a resource to disrupt the communication. The communication model defines the type of "connectivity" that the network operator is trying to achieve, the set of resources she can choose from, and the payoffs (operator's loss and attacker's gain) of the game. The Nash equilibrium strategies are then used to predict the attacker's most likely actions; and the adversary's equilibrium payoff[2] serves as a metric

[1] We use "optimal" in the sense that the defender chooses a best response to the attacker's strategy.

[2] It has been shown that the attacker's payoff is the same in every equilibrium of a net-

for the vulnerability (i.e., inverse robustness) of the network. This metric has a number of interesting properties that are discussed in this paper by using illustrative examples. Furthermore, for the communications models considered here, the metrics correspond to well-known graph-theoretic notions.

As our metrics are derived from game-theoretic models, computing them requires solving the games. With respect to the complexity of computing a Nash equilibrium, NBG models present two challenges. First, at least one player's strategy set (and, hence, the payoff matrix) is only *implicitly* defined, and the actual strategy set needs to be computed from the input of the game, i.e., from the communication model and the network topology. Second, even though checking whether a given action is a feasible strategy can be done efficiently in most NBG models, computing a player's complete strategy set is inherently difficult. In fact, in most cases, the payoff matrix is exponential in size. As a consequence, solving network blocking games can be expected to be harder than solving games for which the payoff matrix is "explicitly given". For such explicit games, computing a NE has been shown to be PPAD-complete (*Polynomial Parity Arguments on Directed graphs*), a class of problems that are believed to be hard, but not necessarily NP-hard [6]. In this paper, we show that computing a Nash equilibrium of a network blocking game is NP-hard in general.

Interestingly though, in the series of NBG papers cited above, new algorithms have been developed to *efficiently* compute a Nash equilibrium in a number of communication models: All-to-All (e.g., Ethernet) networks with constant [19] and linear loss [28], All-to-One (e.g., access and sensor) networks [27], and Supply-Demand networks [18]. These algorithms are mostly based on the theory of network flows and, for some models, on the minimization of submodular functions. More precisely, the problem of finding a Nash equilibrium is cast as a network flow problem (or a submodular function minimization problem), which enables bypassing the computation of the payoff matrix.

In previous NBG models [19, 20, 28, 27], it is assumed that – when there is no attack – the operator is indifferent to which strategy she is using. When there is an attack, the operator is only interested in minimizing her expected loss due to attacks and remains indifferent among the strategies that achieve this minimum loss. Implicitly, the assumption is that network resources can be used at zero cost. This assumption is however not realistic. In practice, network elements have positive usage costs (e.g., operation and maintenance costs, protection costs, quality of service), and these costs may be non-uniform. Consequently, a strategy that achieves the minimum loss can have very high cost and thus be undesirable to the operator. Furthermore, network operators do not have an unlimited budget, which could allow them to use any combination of network resources. In sum, in addition to security, there are other economic and technical goals that network operators have to take into consideration. Often, security and economic goals conflict with each other implying the necessity for the operator to find a balance between them.

work blocking game; thus, it suffices to find a single equilibrium in order to characterize the robustness of a network.

In [18], a usage cost model as well as a budget constraint have been introduced for the particular case of Supply-Demand (S-D) networks. This budget constraint means that the network operator can use a set of network elements (links) only if its associated cost does not exceed a given budget. In the present paper, we extend the budget constraint idea to network blocking games in general. We introduce a unit cost for each network resource and use these unit costs to define a cumulative cost for a set of network resources. We also assume that the operator has a fix budget to operate the network. We integrate the costs and budget into the game by defining two constraint formulations: *maximum* and *expected* cost. In the maximum cost formulation, the constraint is applied to the operator's pure strategies, while in the expected cost formulation, it is applied to her mixed strategies. We then define a NBG for the given budget limit and use the equilibrium payoff as the metric for vulnerability corresponding to that budget.

The budget limit can be considered as the network operator's security investment to reduce vulnerability. When there is no investment, the system is expected to have maximum vulnerability. On the other hand, if the operator has an infinite budget, then a (clever) investment can reduce the vulnerability to a minimum value. By letting the budget vary between its minimum and its maximum values, we obtain the vulnerability/cost tradeoff curve. Notice that in this game model, an equilibrium means that the defender chooses a best response to the attacker strategy. As a consequence, each achieved vulnerability (hence the obtained tradeoff) can be considered as "optimal" in the best response sense.

Since computing the vulnerability metric (i.e. solving the game) is in general NP-hard for the model without constraints, one can readily conclude that it remains NP-hard for the constrained model (the absence of constraint is equivalent to an infinite budget). Hence, the interesting question is: "what happens to the complexity of the models cited above, for which there exist efficient algorithms in the unconstrained case?" We show that for these models, the maximum cost constraint leads to NP-hard problems while the expected cost constraint formulation leads to games that can be solved efficiently.

This article is a synthesis and extension of the authors' previously published conference papers [25] and [26]. It builds upon the study in [18], but considers a more general setting and presents many additional results. [18] is the first study to introduce the idea of a budget limit and usage costs in the context of a NBG. However, it considers only the special case of Supply-Demand networks and (what we call here) the maximum cost constraint. Furthermore, it does not provide a complexity analysis. In the present paper, we consider the general definition of NBGs, introduce a second cost constraint, and provide a thorough complexity analysis.

The main contributions of this paper are the following. We show that solving a blocking game is generally NP-hard (Theorem 1). We generalize the network blocking game model by introducing usage costs to network links and a budget limit for the operator and consider two constraint formulations: the maximum cost constraint (MCC) and the expected cost constraint (ECC). We analyze the complexity of solving the constrained game in the previously proposed mod-

els, which can be solved efficiently in the unconstrained case. We show that the problem of determining the equilibrium payoff is NP-hard under the MCC (Theorem 2) and, for the ECC model, we show how to solve the game in polynomial time given a linear characterization of the operator's mixed strategy space (Theorem 3). We provide complete proofs for all our results, and we provide the formulas of our proposed metric and discuss it in the case of some known communication models and classic graph topologies. Finally, we apply our tradeoff analysis to two real-life network topologies.

This paper can be viewed as composed of two main parts. In the first part, assuming that the network operator only worries about security, we revisit the previously introduced NBG models and discuss our proposed vulnerability metric (Section 2). We also prove the NP-completeness of computing such a metric in the general case (Section 3). In the second part of the paper, assuming that the operator has additional economic goals, we present our cost model, derive the vulnerability/cost tradeoff, and apply the tradeoff analysis to two real-life network topologies in Section 4. We also reconsider the computational complexity in this case (Section 5). We briefly discuss related work on the vulnerability of network topologies in Section 6. Finally, we provide concluding remarks and discuss some future work in Section 7. For ease of reading, all proofs have been moved to the appendix.

Notational Conventions

We use lower case bold letters (e.g., $\boldsymbol{\alpha}$) and upper case bold letters (e.g., $\boldsymbol{S}$) to denote column vectors and matrices, respectively. We use the prime sign ($'$) to denote transposition, and subindices (e.g., α_T) to refer to elements of vectors. For the presentation and the analysis of our model, we make use of a number of symbols. For quick reference to these symbols, we list them in Table 1.

2 Unconstrained Network Blocking Games

In this section, we summarize the previous work on network blocking games. Since these models do not consider a budget constraint, we will refer to them as *unconstrained network blocking games* whenever the distinction is important. We first discuss three examples of *communication models*. Then, we present the game model and discuss the characterization of its Nash equilibria. Finally, we discuss the properties of our proposed vulnerability metrics by using illustrative examples.

As it was stated earlier, network blocking games are defined by the communication model and the topology of the network. The topology of the network is represented by a connected simple graph $G = (V, E)$, where V is the set of nodes and E is the set of links. The edges can be undirected or directed depending on the communication models (as we will see later). The network operator wants to guarantee *"some" connectivity* between the nodes of the network. For this, she selects a collection $T \subseteq E$ of the links as the communication infrastructure.

Table 1: List of Symbols

Symbol	Description
$G = (V, E)$	graph representing the network topology
$\mathcal{T}$	set of feasible collections for the operator
$\lambda(T, e)$	usage of link e in collection T
θ^*	vulnerability of the network
$\boldsymbol{\mu}$	attack costs for the adversary
w_e	unit usage cost of link e
$w(T)$	cumulative usage cost of collection T
$w(\boldsymbol{\alpha})$	expected usage cost of strategy $\boldsymbol{\alpha}$
b	operator's budget
All-to-One communication model	
r	designated node
Supply-Demand communication model	
$s(v)$	supply at node v
$d(v)$	demand at node v

The type of *connectivity* and the set of feasible collections (denoted by $\mathcal{T}$) are determined by the communication model (see the next subsection for examples of communication models).

In this paper, we only consider failures that are due to the actions of a malicious and strategic adversary. Assuming that the operator chooses collection T for her communication and that a given link e in the network fails, if $e \notin T$, then the communication is not affected at all. If, on the other hand, $e \in T$, then e can no longer be used: the operator incurs some *usage loss*, which is how much she would transmit using the link if it were intact. For a given T and e, we let $\lambda(T, e)$ denote this usage loss (or zero if $e \notin T$). Notice that all results presented in this paper also hold if the attacker is allowed to attack nodes as well[3], but for ease of presentation, we restrict our analysis to link attacks.

2.1 Communication Models

The communication model defines the type of "connectivity" that the network operator is trying to achieve, the set of feasible collections $\mathcal{T}$ which she can use for that, and the usage losses $\lambda(T, e)$ for the network elements. Next, we introduce three examples of a communication model. Note that, in these three communication models, the set of feasible subsets $\mathcal{T}$ is only implicitly defined. Furthermore, the size of the set $\mathcal{T}$ is in general exponential for each model; hence, there exists no efficient algorithm to list all elements of $\mathcal{T}$.

[3]The results for both node and edge attacks can be derived using node splitting.

2.1.1 Supply-Demand Model

In a Supply-Demand (S-D) network [18], the operator wants to carry a fixed amount of goods from a nonempty set $S \subseteq V$ of "source" nodes to a nonempty set $D \subseteq V$ of "destination" nodes using the network links. We assume that $S \cap D = \emptyset$ and that network links are directed. With each node $u \in S$, we associate a nonnegative number $s(u)$, the "supply" at u, and with each node $u \in D$, we associate a nonnegative number $d(u)$, the "demand" at u. We consider *uncapacitated* networks, where each link can carry an unlimited amount of goods[4]. We also assume that links carry only *integer* amounts of goods and that the total amount of goods to be carried from S to D is also a given positive integer.

To transport the goods, the network operator chooses a collection of links that forms a *feasible (integer) flow*. A feasible flow $T \in \mathcal{T}$ is a function that assigns to each link e the amount of goods $T(e)$ (≥ 0) it carries, such that the *conservation of flow* property is satisfied at each node. Hence, the set of collections $\mathcal{T}$ is equal to the set of all feasible flows.

The usage (loss) $\lambda(T, e)$ is defined to be the *amount of goods $T(e)$ that flow T assigns to link e*. This is how much the operator will lose if she uses feasible flow $T \in \mathcal{T}$ and link e fails.

2.1.2 All-to-One Model

In an *All-to-One* network [27], the primary goal of the network operator is to enable all nodes to communicate with a designated node r. This models sensor and access networks, where all nodes are trying to reach a gateway or data collection node (or, alternatively, a set of nodes, which can be modeled by a designated super-node).

To get all nodes connected to r, the network operator chooses a collection of links T that forms a spanning tree. Hence, the set of feasible collections $\mathcal{T}$ is the set of all spanning trees. In practice, a spanning tree can be implemented, for example, as the next-hop forwarding table entries for r, which are stored at the individual nodes of the network.

Let the network be connected using a spanning tree T. Then, if a given link $e \in E$ fails, some nodes might no longer be able to communicate with r and can be considered lost for the network operator. Thus, we define the usage (loss) $\lambda(T, e)$ as *the number of nodes that are disconnected from r* when the operator chooses T to connect the network and link e fails.

2.1.3 All-to-All Model

In an *All-to-All* network [19, 28], the goal of the network operator is to enable each node to communicate with every other node, using the minimum number of links. For example, this is the case for bridged Ethernet LANs, where every

[4]The analysis of *capacitated* network follows from the study in this paper, but it is not considered in this paper due to space limitation.

node should be able to "logically" communicate with every other node, but the topology has to be loop-free. Assuming that links are undirected, spanning trees are the subgraph structures that (looplessly) connect all nodes with the minimum number of links. Hence, the network operator selects a spanning tree as communication infrastructure, and the set of feasible collections $\mathcal{T}$ corresponds to the set of all spanning trees.

Let the network be connected using a spanning tree T and assume that link e fails. If link e does not belong to T, then the network remains connected and the operator does not lose any connectivity. If, on the other hand, $e \in T$, the network is cut into two separate components that are unable to communicate. Now, if e is a link connecting a leaf to the rest of the spanning tree, only that leaf gets disconnected and all the other nodes can still reach each other. In this case, the operator loses some connectivity, but the loss can be considered *minor*. If, on the other hand, the removal of link e cuts the network into two components of comparable size, then connections between many pairs of nodes are now missing, and the loss to the operator is considerably *larger*. To capture this phenomenon, the usage (loss) $\lambda(T, e)$ is defined as *the size of the smaller connected component of* $G(V, T \setminus e)$, where $G(V, T \setminus e)$ is the subgraph containing only the links in $T \setminus e$.

2.2 Game Model

Given a communication model and the topology of a network, we define a two-player game between the network operator and a strategic attacker as follows. The network operator wants to guarantee "some" connectivity by choosing a *feasible* collection of links in the network (i.e., her strategy space is the set $\mathcal{T}$ of feasible collections). The type of connectivity and the set of feasible collections are defined by the communication model, as previously discussed. At the same time, a strategic and malicious adversary is trying to disrupt the communication by attacking a link (i.e., her strategy space is the set E of links in the network). We assume that to successfully attack a link e, the adversary has to spend some effort which is quantified by a cost of attack μ_e. The players' payoffs are defined as follows: when the operator picks collection T and the attacker targets link e, the operator loses $\lambda(T, e)$ (as defined above), and the attacker gets a net reward of $\lambda(T, e) - \mu_e$. Intuitively, if the attack costs are too high, the adversary will probably not launch an attack. To capture this, we assume that the attacker has the option not to launch an attack, which results in zero loss for the operator and zero gain for the attacker.

We consider mixed strategy Nash equilibria, where the network operator chooses a distribution (denoted by $\boldsymbol{\alpha}$) over the set $\mathcal{T}$, and the attacker chooses a distribution (denoted by $\boldsymbol{\beta}$) over the set E or the option of not attacking. We assume that the operator's goal is to minimize her *expected loss*, while the attacker's objective is to maximize her *expected net reward*. Formally, the operator

chooses $\boldsymbol{\alpha}$ to minimize $L(\boldsymbol{\alpha}, \boldsymbol{\beta})$ defined as

$$L(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{T \in \mathcal{T}} \sum_{e \in E} \alpha_T \beta_e \lambda(T, e) \ , \tag{1}$$

while the attacker chooses $\boldsymbol{\beta}$ to maximize $R(\boldsymbol{\alpha}, \boldsymbol{\beta})$ defined as

$$R(\boldsymbol{\alpha}, \boldsymbol{\beta}) = L(\boldsymbol{\alpha}, \boldsymbol{\beta}) - \sum_{e \in E} \beta_e \mu_e \tag{2}$$

or not attacking if the maximum is negative.

2.3 Equilibrium Characterization

Here, we recall the notions of polyhedra and blockers, and discuss how they can be used to characterize the Nash equilibria of the game (see [17, Chap. 4] for more details).

Let $\boldsymbol{\Lambda}$ be the adversary's payoff matrix. We let the rows of $\boldsymbol{\Lambda}$ be denoted by $\boldsymbol{\lambda}_T$, $T \in \mathcal{T}$, where the entries of each row vector $\boldsymbol{\lambda}_T \in \mathbb{R}^{|E|}_{\geq 0}$ are given by $\lambda(T, e)$, $e \in E$. We define the polyhedron $P_{\boldsymbol{\Lambda}}$ associated with $\boldsymbol{\Lambda}$ as the vector sum of the convex hull of the row vectors $\boldsymbol{\lambda}_T$, $T \in \mathcal{T}$ and the nonnegative orthant. In other words, a vector $\boldsymbol{x} \in \mathbb{R}^{|E|}_{\geq 0}$ is an element of $P_{\boldsymbol{\Lambda}}$ iff it can be expressed as the sum of a nonnegative vector and a convex linear combination of the rows of $\boldsymbol{\Lambda}$. This polyhedron can also be expressed as

$$P_{\boldsymbol{\Lambda}} = \left\{ \boldsymbol{x} \in \mathbb{R}^{|E|}_{\geq 0} \ \middle| \ \exists \boldsymbol{\alpha} \in \mathbb{R}^{|\mathcal{T}|}_{\geq 0} \left(\boldsymbol{\Lambda}' \boldsymbol{\alpha} \leq \boldsymbol{x} \ \wedge \ \boldsymbol{\alpha}' \mathbf{1} \geq 1 \right) \right\} \ . \tag{3}$$

Next, the *blocker* $bl(P_{\boldsymbol{\Lambda}})$ of $P_{\boldsymbol{\Lambda}}$ is the polyhedron defined as

$$bl(P_{\boldsymbol{\Lambda}}) := \left\{ \boldsymbol{y} \in \mathbb{R}^{|E|}_{\geq 0} \ \middle| \ \forall \boldsymbol{x} \in P_{\boldsymbol{\Lambda}} \left(\boldsymbol{y}' \boldsymbol{x} \geq 1 \right) \right\} \ . \tag{4}$$

In other words, a vector $\boldsymbol{y} \in \mathbb{R}^{|E|}_{\geq 0}$ is an element of $bl(P_{\boldsymbol{\Lambda}})$ iff its product with each element of $P_{\boldsymbol{\Lambda}}$ is at least 1. Since the blocker $bl(P_{\boldsymbol{\Lambda}})$ is also a polyhedron, its set of vertices (i.e., extreme points) is well-defined. For each vertex $\boldsymbol{\omega} \in \mathbb{R}^{|E|}_{\geq 0}$ of the blocker, we define the quantity

$$\theta(\boldsymbol{\omega}) := \frac{1}{\sum_{e \in E} \omega_e} \left(1 - \sum_{e \in E} \omega_e \mu_e \right) \ . \tag{5}$$

A vertex of the blocker is called *critical* if it maximizes the quantity $\theta(\boldsymbol{\omega})$, i.e., $\theta(\boldsymbol{\omega}) = \max_{\tilde{\omega}} \theta(\tilde{\boldsymbol{\omega}})$. Finally, we let $\tilde{\theta}$ denote the maximum quantity.

Since the attacker has the option to not attack and get a payoff of zero, it is not hard to show that there does not exist an equilibrium in which the attacker receives a negative expected payoff. In [17], it has been shown that in every Nash equilibrium where the attacker launches an attack, her strategy corresponds to a critical vertex or a convex combination of critical vertices. Her equilibrium

Table 2: Vulnerability Metrics of Some Communication Models

Communication Model	Vulnerability Metric θ^*
S-D (uncapacitated)	$\max_{U\subseteq V:\ d(\bar{U})-s(\bar{U})\geq 1}\left(\frac{\left(d(\bar{U})-s(\bar{U})\right)-\boldsymbol{\mu}(\delta(U))}{\lvert\delta(U)\rvert}\right)$
All-to-One	$\max_{U\subseteq V\setminus\{r\}}\left(\frac{\lvert U\rvert-\boldsymbol{\mu}(\delta(U))}{\lvert\delta(U)\rvert}\right)$
All-to-All (linear loss, $\boldsymbol{\mu}=0$)	$\leq \max_{U\subset V:\ 0<\lvert U\rvert\leq\frac{\lvert V\rvert}{2}}\left(\frac{\lvert U\rvert}{\delta(U)}\right)$
All-to-All (constant loss) [19]	$\max_{\mathcal{U}\ \text{partition}}\left(\frac{\lvert\mathcal{U}\rvert-1-\boldsymbol{\mu}(\delta(\mathcal{U}))}{\lvert\delta(\mathcal{U})\rvert}\right)$

Notations: For a set $U \subset V$, $\delta(U)$ is the set of links connecting nodes in U and nodes in $\bar{U}$, where $\bar{U} = V \setminus U$. $\mathcal{U} = \{U_1, U_2, \ldots, U_{|\mathcal{U}|}\}$ is a partition of the graph into connected components and $\delta(\mathcal{U})$ is the set of edges whose end nodes belong to different components.

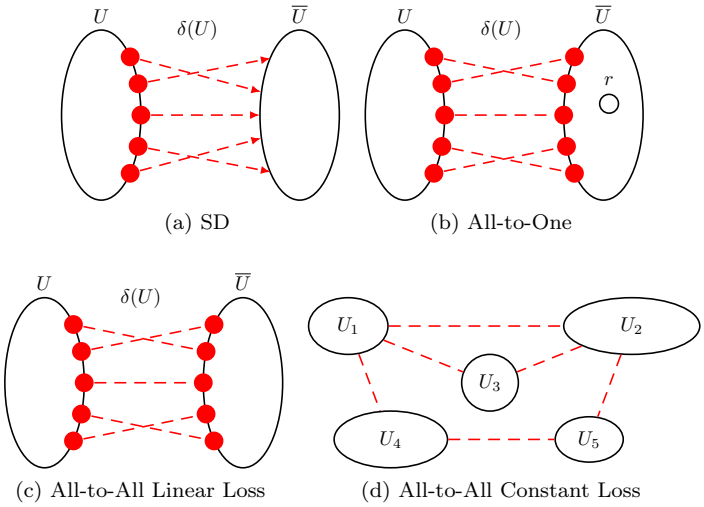
payoff is also shown to be the same in all equilibria and can be written as $\theta^* = \max(0, \tilde{\theta})$, where $\tilde{\theta} = \max_{\boldsymbol{\omega}} \{\theta(\boldsymbol{\omega})\}$, where $\boldsymbol{\omega}$ is a critical vertex of the blocker $bl(P_{\boldsymbol{\Lambda}})$. As a consequence, if this blocker can be "efficiently" characterized, then an efficient algorithm can be derived to solve the maximization problem and, hence, the game.

2.4 Vulnerability Metric

In the analysis of the general NBG [17, Chap. 4], it has been shown that the attacker's equilibrium payoff θ^* is a property of (i.e., solely determined by) the topology of the network, the communication model, and the attack costs $\boldsymbol{\mu}$. Furthermore, this unique equilibrium payoff reflects both the network operator's expected loss due to attack as well as the attacker's willingness to attack. For a given $\boldsymbol{\mu}$, a low θ^* indicates that operating the network has low expected loss due to attack, that is, the network is robust against attacks. If, on the other hand, θ^* is high, then the expected loss is also high, and the network can be considered vulnerable. As such, θ^* has been proposed [19] as a measure of network topology vulnerability (i.e., inverse robustness) in an adversarial environment.

Another property of θ^* is that, when $\boldsymbol{\mu} = \mathbf{0}$ (the case of the most powerful attacker), it can be related to well-known graph-theory notions. Table 2 gives the formulas of the vulnerability metric for the communication models introduced in Subection 2.1. We also provide the formula for the All-to-All model with constant loss introduced in [19].

Supply-Demand Model For the S-D model, the vulnerability metric θ^* (shown in the first row of Table 2 and illustrated in Figure 1a) can be read



(a) SD (b) All-to-One

(c) All-to-All Linear Loss (d) All-to-All Constant Loss

Figure 1: Illustration of the vulnerability metrics: (a) S-D, (b) All-to-One, (c) All-to-All Linear Loss, (d) All-to-All Constant Loss.

as follow: For the set of nodes U, $d(U) - s(D)$ is the total excess demand in $\bar{U}$. This excess demand has to be produced from sources in U and carried over the edges $\delta(U)$ going from U to $\bar{U}$ (i.e. the edge-cut induced by U). The total cost of attacking those edges is $\boldsymbol{\mu}(\delta(U))$. Hence, if the attacker were able to attack all edges in $\delta(U)$, the net attack reward would be $d(\bar{U}) - s(\bar{U}) - \boldsymbol{\mu}(\delta(U))$ (the term in the numerator). However, since the attacker can target only one link, her payoff is given by the expected net reward $\frac{d(\bar{U})-s(\bar{U})}{|\delta(U)|} - \frac{\boldsymbol{\mu}(\delta(U))}{|\delta(U)|}$ when she targets links in $\delta(U)$ with uniform probability. The Nash equilibrium theorem tells us that the vulnerability metric θ^* corresponds to the maximum possible value of this expected net reward, taken over all subset of nodes U. To design the optimal attack strategy that achieves θ^*, the attacker first chooses a set of nodes U that maximizes this expected net reward and then targets the edges of $\delta(U)$ with uniform probability.

This vulnerability metric (i.e. the maximum possible expected payoff) can be interpreted as the attacker's willingness to attack. If it is negative (when the attack costs $\boldsymbol{\mu}$ are large), the attacker will not launch an attack (she will instead choose the no-attack strategy). The larger it is, the more attractive the network is for the attacker. Also, notice that its first term $\frac{d(\bar{U})-s(\bar{U})}{|\delta(U)|}$ corresponds to the expected non-satisfied demand due the attacker's action (i.e. the defender's loss due to the attack). For a fixed $\boldsymbol{\mu}$, a larger value for this term indicates a more vulnerable network. In sum, the vulnerability metric θ^* reflects both the attacker's willingness to attack as well as the defender's expected loss.

All-to-One Model For the All-to-One model (shown in the second row of Table 2 and illustrated in Figure 1b), θ^* can be interpreted in the same way. Here, U is a set of nodes that does not include the designated node r. When edges in $\delta(U)$ are attacked, all nodes in U are disconnected from r and hence are lost for the defender. The total net attack reward for targeting those edges is given by $|U|-\boldsymbol{\mu}(\delta(U))$ and the corresponding expected net reward is $\frac{|U|-\boldsymbol{\mu}(\delta(U))}{|\delta(U)|}$. The vulnerability metric θ^* is the maximum of such a quantity over all U. When designing her attack, the attacker chooses a set U that maximize this quantity and uniformly targets the edges in $\delta(U)$. It is noteworthy that in this All-to-One model, when $\boldsymbol{\mu} = 0$, θ^* is equal the inverse of the *persistence* of the graph of the network, a metric that has previously been proposed in [10] to quantify graph robustness (although in a non-game-theoretic framework).

All-to-All Models The vulnerability metric for the All-to-All model with linear loss is shown in the third row of Table 2 and is illustrated in Figure 1c. It can be interpreted as an All-to-One where the source is required to belong to the largest connected component. In this case, θ^* is tightly upper bounded by the *Cheeger constant* [7] (also called the *edge-expansion*) of the graph when $\boldsymbol{\mu} = 0$.

Finally, the vulnerability metric for the All-to-All model with constant loss [19] (Figure 1d) can be read as follow: $\mathcal{U} = \{U_1, U_2, \ldots, U_{|\mathcal{U}|}\}$ is a partition of the graph into connected components, $\delta(\mathcal{U})$ is the set of edges whose end nodes belong to different components, and $|\mathcal{U}|$ is the number of connected components. By uniformly attacking the edges in $\delta(\mathcal{U})$, the attacker gets an expected net attack reward equal to $\frac{|\mathcal{U}|-1-\boldsymbol{\mu}(\delta(\mathcal{U}))}{|\delta(\mathcal{U})|}$ and the vulnerability metric θ^* is equal to its maximum possible value, taken over all partitions $\mathcal{U}$. To design her attack, the attacker chooses a partition $\mathcal{U}$ that achieves θ^* and uniformly targets the edges in $\delta(\mathcal{U})$. In this case, when $\boldsymbol{\mu} = 0$, θ^* can be related the *spanning tree packing (STP) number of the graph*, i.e. the number edge-disjoint spanning trees ($STP = \lceil \frac{1}{\theta^*} \rceil$) [35].

Comparison with Other Metrics To contrast our proposed metric with conventional ones, consider the example S-D network shown in Figure 2. We

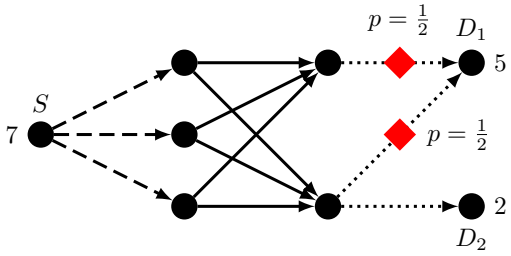


Figure 2: Example of an S-D network.

will compare θ^* with the metrics (i.e. attacker payoff) obtained from (1) a totally random attacker who uniformly targets all links in the network, (2) a min-cut attacker who first computes an edge-min-cut of the graph and then uniformly targets them, and (3) a risk-averse attacker who tries to get the maximum deterministic attack reward. In this comparison, we assume that there is no attack cost (i.e. $\boldsymbol{\mu} = 0$). The network contains one source S with a supply of $s(S) = 7$ of goods and two destinations D_1 and D_2 with respective demands $d(D_1) = 5$ and $d(D_2) = 2$.

For this network, the equilibrium strategy for the attacker is to uniformly target the two links marked with the diamonds, and the corresponding expected reward (i.e. the vulnerability metric θ^*) is equal to $\frac{7}{2}$. A totally random attacker targets each link with probability $\frac{1}{12}$ and gets an expected reward of $\frac{7}{12}$, which is much less than θ^*. Indeed, such a naive strategy "wastes" a lot of attack effort by targeting links that do not carry goods in the defender's equilibrium strategy. A more clever strategy, which leads to a very commonly used metric (i.e. the connectivity), is that of a min-cut attacker. In the figure, the two min-cuts of the graph are marked by the dashed and dotted lines, respectively. The attacker can choose either one of the two cuts and picks a link to attack with probability $\frac{1}{3}$. The corresponding expected attack reward is $\frac{7}{3} < \theta^*$. It can be shown that in general, the reward obtained by a min-cut attacker is always less than or equal to θ^*. Notice that the total traffic carried through the min-cut is larger than the total traffic carried over the links targeted by the NE attacker (proposed in this paper). However, the min-cut contains more links and leads to a smaller expected reward. Finally, the risk-averse attacker targets the only link going to destination D_2 and receives a payoff of $2 < \theta^*$. Indeed, in general, a risk-averse attacker will always get a smaller payoff.

This simple example shows that a NE attacker (as proposed in this paper) is more sophisticated because she uses complete knowledge about the network to design her attack strategy. In reality, full knowledge of the graph of the network is not always available: the attacker (and sometimes even the defender) might have only partial knowledge about the graph. Despite this limitation, the vulnerability metric proposed here can be considered as a (worst-case) benchmark, when the attacker has complete information and both the attacker and the defender are fully rational.

2.5 Example: Vulnerability Values of Classic Graphs

We use our proposed metrics to compute the vulnerability of some classic graph topologies. We have assumed that the attack costs $\boldsymbol{\mu} = \mathbf{0}$, which corresponds to the most powerful attacker in our model. Table 3 shows the vulnerability for the complete graph, the wheel, the ring, the star, and the path topologies. We have computed the vulnerabilities for the All-to-One and All-to-All (linear loss) communication models. For the S-D communication model, the metric depends on where the sources S and the destinations D are located in the network. We leave such discussion for interested readers. We also provide the formulas for the All-to-All model with constant loss introduced in [19]. Recall that for this

Table 3: Vulnerability Values of Some Common Graphs

Graph	Communication Model		
	All-to-One	All-to-All (linear)	All-to-All (constant)
Complete	1	$\frac{2}{n}$	$\frac{2}{n}$
Ring	$\frac{n-1}{2}$	$\frac{n}{4}$	$\frac{n-1}{n}$
Wheel	$\begin{cases} 1, & \text{if hub } = r \\ \frac{n-1}{3}, & \text{if hub } \neq r \end{cases}$	$\begin{cases} \frac{n}{n+4}, & \text{if } n \text{ is even} \\ \frac{n+1}{n+5}, & \text{if } n \text{ is odd} \end{cases}$	$\frac{1}{2}$
Star	$\begin{cases} 1, & \text{if hub} = r \\ n-1, & \text{if hub } \neq r \end{cases}$	1	1
Path	$\begin{cases} n-1, & \text{if } r \text{ is a leaf} \\ n_r, & \text{if } r \text{ is not leaf} \end{cases}$	$\frac{n}{2}$	1

Notation: n denotes the number of nodes in the graph, r is the designated node of the All-to-One model, n_r is the size of the largest subgraph connected to r.

model, if the attacked link belongs to spanning tree chosen by the defender, then she loses a constant amount which is normalized to 1. As a consequence, the vulnerability $\theta^* \in [0, 1]$.

As a metric for robustness, understanding the computational complexity of calculating θ^* is of primal importance. In the next section, we discuss the complexity of computing a Nash equilibrium in the unconstrained NBG model.

3 Computational Complexity of the Unconstrained Game

In this section, we show that solving a NBG is generally NP-hard. Recall that computing a Nash equilibrium in general two-player games has been shown to be PPAD-complete. Zero-sum, two-player games, on the other hand, can be cast as linear programs and, hence, can be solved in polynomial time using linear programming tools. In all these cases, the input of the computational problem is assumed to be the payoff matrix. For NBG models, however, only an *implicit* description of the payoff matrix is given. In addition, the payoff matrix is potentially exponential in size, which makes NBG models even more challenging to deal with.

The following theorem shows that, indeed, computing a NE for a general blocking game is NP-hard. We prove this by reducing a well-known NP-hard problem, the *Knapsack Problem* (KP), to the problem of computing the at-

tacker's equilibrium payoff, which we formalize as the *Equilibrium Problem* (EP). KP and EP are formally defined as follows.

Definition 1 (Knapsack Problem [KP])**.** Given N items, where item $i = 1, \ldots, N$ has weight c_i and value v_i, a capacity C, and a value V, is there a subset S whose sum weight is at most C (i.e., $\sum_{i \in S} c_i \leq C$), and whose sum value is at least V (i.e., $\sum_{i \in S} v_i \geq V$)?

Definition 2 (Equilibrium Problem [EP])**.** Given a set of elements E, a polynomial-time computable function $I_{T \in \mathcal{T}}$ for testing $T \in \mathcal{T}$ (i.e., the feasibility of collections), a polynomial-time computable usage function $\lambda(T, e)$, a vector of attack costs $\boldsymbol{\mu} \in \mathbb{R}_{\geq 0}^{|E|}$, and a threshold payoff value p, is the adversary's equilibrium payoff less than or equal to p?

The above formulation of EP allows us to easily show the computational complexity of all the problems relevant to NBGs. First, if the adversary's equilibrium payoff can be efficiently computed, then EP can also be solved efficiently. Conversely, if EP is NP-hard, then computing the adversary's equilibrium payoff is also necessarily NP-hard. Second, for similar reasons, we also have that computing a mixed-strategy equilibrium $(\boldsymbol{\alpha}, \boldsymbol{\beta})$ of the game is at least as hard as EP.

The following theorem shows that EP is NP-hard.

Theorem 1. *The Knapsack Problem is polynomial-time reducible to the Equilibrium Problem.*

The proof of the theorem can be found in Appendix A.

Thus, solving a NBG is generally NP-hard. Interestingly, however, efficient algorithms have been derived to compute a NE for the models discussed in Subsection 2.1 (the All-to-All with linear loss, the All-to-One, and the Supply-Demand communication models). It is our conjecture that there exists a class of blocking games (defined by the loss function $\lambda(T, e)$) that can be solved efficiently even when the payoff matrix is exponential in size (indeed the solution has to somehow bypass using the payoff matrix).

4 Budget Contraints

In the unconstrained NBG model, the operator is only interested in minimizing her expected loss due to attacks, without taking any other economic or technical concerns into consideration. More formally, the unconstrained model assumes that – when there is no attack – the operator is indifferent to which strategy she is using. In practice, however, network operators also have to take economic goals and constraints into consideration. Since network elements and, hence, feasible collections can have varying costs, these economic considerations can affect the operator's strategic choices. Furthermore, as a more robust strategy can entail higher costs, economic and security goals can conflict with each other.

Hence, the operator has to balance between security and cost, which will affect her choice of strategy.

In this section, we show how economic factors can be taken into consideration using the framework described in the first part of the paper. We first introduce costs for the network resources and discuss how they can be interpreted. These costs can be integrated in the game in several ways. In this paper, we use them to formulate budget constraints on the operator. More precisely, we assume that the operator has a fixed budget b and she can use a strategy only if its associated cost fits into this budget. We then define a game for each budget limit b and derive a vulnerability metric $\theta^*(b)$ that is parametrized by b. Finally, by varying b in the range of all possible budgets, we derive the tradeoff curve.

4.1 Cost Model

A network link is often associated with some cost, which models the amount of effort (money, energy, time, etc.) that the operator needs to spend to use the link (e.g. setup, maintenance, protection). Furthermore, links usually have different costs. In [18], a usage cost model was introduced and discussed for the particular case of the S-D communication model. Here, we extend this cost model to network blocking games in general. Recall that $\lambda(T, e)$ quantifies the usage of link e when the operator employs collection T. This usage can model, for example, the amount of traffic on link e or the number of active paths between nodes that traverse link e. We assume that each link e has some *unit cost* w_e, so that using the link costs $w_e\lambda(T, e)$ to the operator if she employs collection T.

This unit cost can model multiple economic and technical factors. For example, the operator might have to lease or pay for some of the network elements in order to use them. If we let w_e be the unit usage price for a link, then $\lambda(T, e)w_e$ is the total cost of using link e when the operator selects collection T. Using a network element might also require energy consumption, which in turn can require expenditure from the operator. In this case, w_e is the unit energy cost and $\lambda(T, e)w_e$ is the total energy cost associated with e and T. Another interpretation is that w_e is the unit protection cost, so that $w_e\lambda(T, e)$ is the total cost of protecting against a loss equal to $\lambda(T, e)$. Network links can also have negative effects on the quality of the traffic that goes through them, e.g., introduce different delay or jitter. If we let w_e, be the delay (or jitter) of that link, $\lambda(T, e)w_e$ is the total amount of delay (or jitter) experienced by all traffic that goes through that link.

Henceforth, we will assume that, for each link e, these economic and technical factors have been added together, and a constant w_e is given. Furthermore, we will refer to all of these factors together as *costs* (hence w_e is called the unit cost of link e).

Based on the above definition, the cumulative cost of using the resources in

a collection T is given by

$$w(T) := \sum_{e \in E} \lambda(T, e) w_e \ . \tag{6}$$

If the network operator randomly chooses a collection T according to some probability distribution $\boldsymbol{\alpha}$ (as, e.g., in the *mixed strategy* of the game introduced in the previous section), then we can define the operator's *expected cost* as

$$w(\boldsymbol{\alpha}) := \sum_{T \in \mathcal{T}} \alpha_T w(T) = \sum_{e \in E} w_e \sum_{T \in \mathcal{T}} \alpha_T \lambda(T, e) \ . \tag{7}$$

The operator can take these costs into account in several ways. In this paper, they are used to formulate budget constraints, which we discuss in the next subsection.

4.2 Budget Constraints

We assume that the operator has a fixed *budget* $b \in \mathbb{R}_{\geq 0}$.Therefore, her objective is to minimize her expected loss (see Equation (1)) by choosing the most secure strategy that satisfies her budget constraint. This budget constraint can be formulated in multiple ways. Next, we introduce and study two straightforward formulations, the maximum and the expected (or average) cost budget constraints.

4.2.1 Maximum Cost Budget Constraint

In the *maximum cost constraint* (MCC), we require that for a given budget b, the operator only uses collections whose cumulative cost (see Equation (6)) are less than or equal to b. Formally, the pure strategy set of the operator is restricted to

$$\mathcal{T}^{(b)} = \{T \in \mathcal{T} \mid w(T) \leq b\} \ . \tag{8}$$

The maximum cost constraint is best-suited for budget limits that are determined by the amount of preallocated resources available. In this case, the cost of a link can be the amount of resources needed (e.g., energy consumption) to operate the link and the budget limit can be the amount of resources available (e.g., amount of power available).

4.2.2 Expected Cost Budget Constraint

The maximum cost constraint misses to capture certain situations. For instance, when the amount of allocated resources can be modified during operation, e.g., resources can be leased, the budget limit should apply to the average or, equivalently, the expected cost of a strategy during continuous periods of operation. Thus, in our second budget constraint formulation, which we will refer to as the *expected cost constraint* (ECC), we only require the expected (or average) cost of the operator to not exceed the budget limit.

Under the *expected cost constraint* with a budget limit b, the operator can employ a mixed strategy only if its expected cost (see Equation (7)) is less than or equal to b. Formally, the set of mixed strategies available to the operator is

$$\mathcal{A}^{(b)} = \left\{ \boldsymbol{\alpha} \in [0,1]^{|\mathcal{T}|} \;\middle|\; w(\boldsymbol{\alpha}) \leq b \;\wedge\; \mathbf{1}'\boldsymbol{\alpha} = 1 \right\} . \tag{9}$$

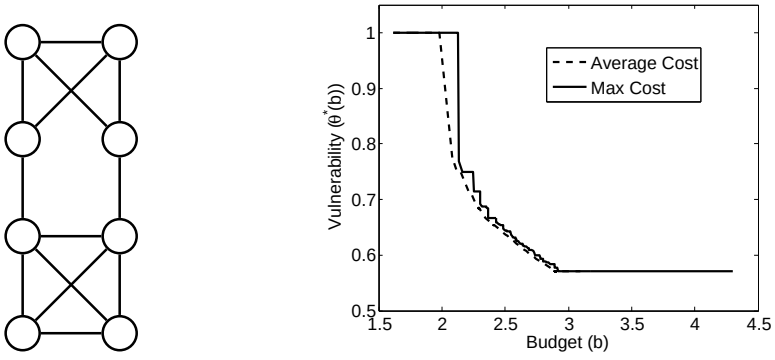
Note that the above formulation generalizes the classic notion of mixed strategies in game theory, where the set of mixed strategies is always the set of *all* distributions over the set of pure strategies. Here, a mixed strategy is chosen from a predefined subset of distributions.

4.3 Constrained Game

Having defined the set of available strategies (pure for MCC and mixed for ECC), we can now setup the constrained game in a similar way to the unconstrained game presented in Subsection 2.2. We are interested in mixed strategy Nash equilibria, where the operator picks a distribution $\boldsymbol{\alpha}$ over $\mathcal{T}^{(b)}$ (for MCC) or from the set $\mathcal{A}^{(b)}$ (for ECC), while the attacker chooses a distribution $\boldsymbol{\beta}$ over the set of links. The attacker's Nash equilibrium payoff is denoted $\theta^*(b)$ for a game with budget limit b.

Using the same interpretation as in Subsection 2.2, the attacker's NE payoff $\theta^*(b)$ can be used to quantify the vulnerability (i.e., inverse robustness) of the network when the operator's budget is b. By varying b, one can draw the Pareto frontier between the region of achievable vulnerability/budget points and the region of unachievable ones, as was done in [18] for the particular case of S-D networks with the maximum cost constraint.

4.4 Vulnerability/Cost Tradeoff



(a) Network topology

(b) Vulnerability / budget tradeoff curve

Figure 3: Example of vulnerability / budget tradeoff in the All-to-All communication model.

In this section, we illustrate the vulnerability/cost tradeoff using the All-to-All communication model on the topology depicted in Figure 3a. The link costs $\boldsymbol{w}_e$ are randomly chosen between 0 and 0.6, which makes the average cost of a spanning tree equal to 2.1. For each value of b, a game is played with the defender's strategy set given by Equation (8) for the maximum cost constraint (MCC) and by Equation (9) for the expected (or average) cost constraint (ECC). In all games, the attacker's strategy set is the set of all links and the cost of attack is $\boldsymbol{\mu} = \mathbf{0}$. Figure 3b shows the vulnerability $\theta^*(b)$ as a function of the budget b for both the MCC and the ECC. Observe that the two curves are very close to each other, but vulnerability for the MCC is always at least as high as for the ECC.

Once the tradeoff curve is determined, the next question is finding the optimal operating point on this frontier. The optimal operating point depends on the specific operator's preferences with respect to the vulnerability and the budget. These preferences can be quantified by a utility function $U(\theta^*, b)$. In general, the optimal operating point is determined by solving a 2-dimensional optimization problem which, in this case, can be reduced to a one-dimensional optimization, and can be written as $b^* = \text{argmax}\{U(\theta^*(b), b) : \ (\theta^*(b), b) \text{ is feasible}\}$.

4.5 Application Example

In this subsection, we apply the tradeoff analysis presented above to two example topologies. We consider an All-to-All communication model with linear loss (Subsection 2.1.3), where the goal of the network operator is to enable each node to communicate with every other node.

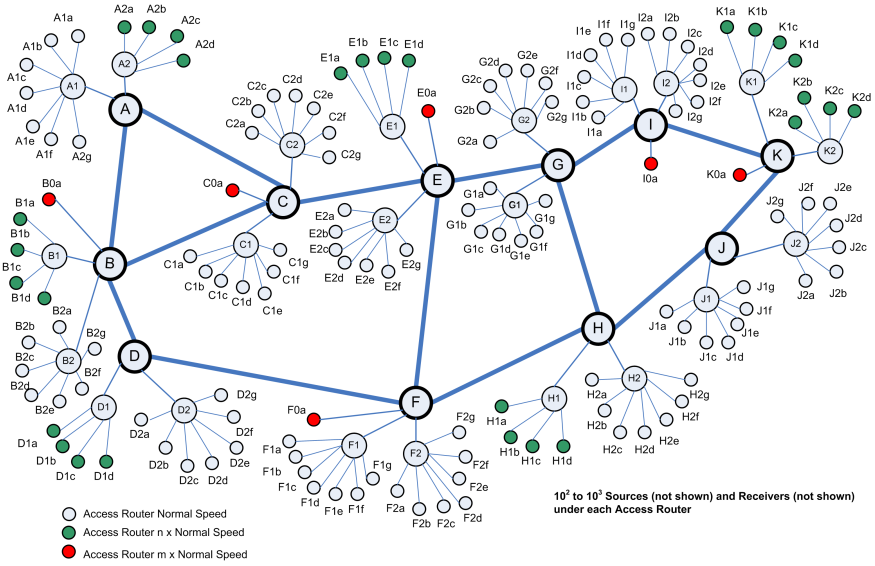


Figure 4: Topology of the Abilene network.

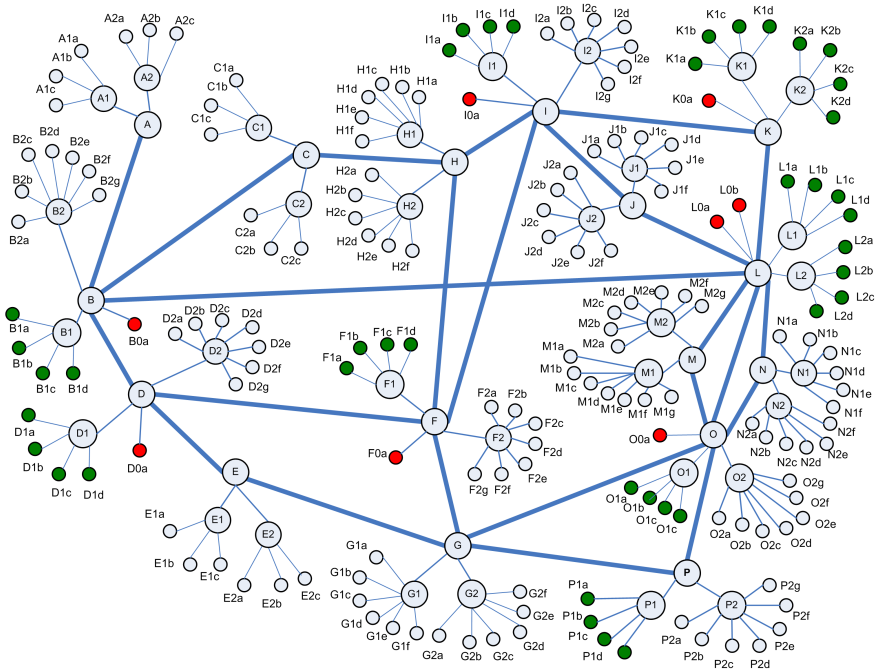
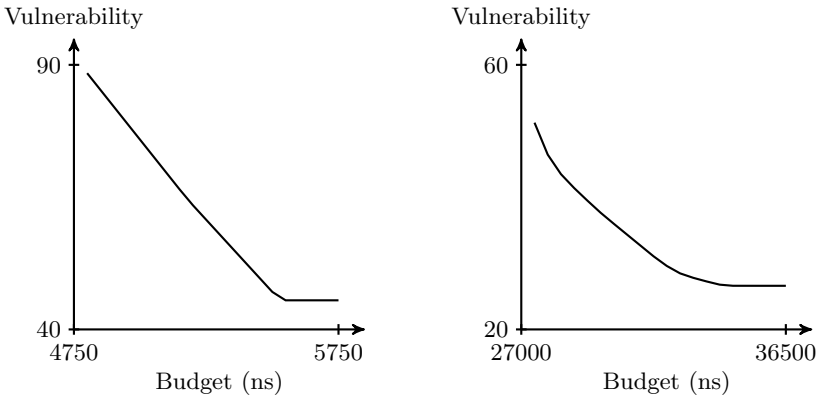


Figure 5: Topology of the ISP3 network.

The first topology (shown in Figure 4) is based on the Abilene network, which was created by the Internet2 community and connects regional network aggregation points to provide advanced network capabilities to over 230 Internet2 university, corporate, and affiliate member institutions in the US. It contains 11 backbone routers, 22 point of presence routers, and 139 access routers [31] (see Figure 4). The second one (shown in Figure 5), which we will call ISP3 [32], is the topology of a modern US Internet service provider (ISP)[5]. Due to space limitation, the topology is shown in Appendix E. It contains 16 Backbone Routers (A-P), 32 Point of Presence Routers (A1-P2) and 170 Access Routers (A1a-P2g). Table 4 summarizes the key properties of these topologies.

The unit cost of a link is assumed to be equal to its propagation delay. In other words, the network operator's goal is to choose links that form a robust spanning tree while trying to keep the total propagation delay low. The links' propagation delays are given in Table 5 (we list the thirty largest delay values for each network). We have only considered the expected cost constraint (ECC) 4.2.2. In both cases, solving the ECC game took less than a second on an average desktop computer. Recall that for the maximum cost constraint (MCC) model, finding a a Nash equilibrium is NP-hard. Figure 6 shows the expected vulnerability/cost tradeoff curves for the two networks.

[5]We cannot reveal the ISP's identity because of disclosure agreement.



(a) Abilene network (b) ISP3 network

Figure 6: Cost / vulnerability tradeoff curves for the example networks.

Table 4: Statistics of the Example Networks

	Abilene	ISP3
Number of nodes	172	218
Number of links	175	226

5 Computational Complexity of the Constrained Game

In this section, we discuss the complexity of solving budget-constrained network blocking games. However, since solving an unconstrained NBG is in general NP-hard (as shown in Theorem 1), we readily have that solving a NBG under a budget constraint[6] is also NP-hard generally. Consequently, we focus our discussion on the communication models introduced in Subsection 2.1, for which efficient algorithms exist to compute the NE payoff in the unconstrained game, and discuss computational complexity under the MCC and ECC.

5.1 NP-Hardness of the Maximum Cost Constraint

We first show that the maximum cost constraint formulation leads to NP-hard problems. More specifically, we show that computing the equilibrium payoff of a network blocking game is NP-hard under a maximum cost budget constraint in the Supply-Demand, All-to-One, and All-to-All communication models, which were previously shown to be efficiently solvable without a budget constraint.

[6]The unconstrained game is the special case of $b \to \infty$.

Theorem 2. *Computing the NE payoff of an NBG under a maximum cost budget constraint is NP-hard in the (a) S-D communication model, the (b) All-to-All communication model, and the (c) All-to-One communication model.*

The proof of the theorem can be found in Appendix B. We show NP-hardness by reducing a well-known NP-hard problem, the *Partition Problem* (*PP*) [30], to the problem of deciding whether the equilibrium payoff in a given network under a maximum cost constraint is at most a certain value. We refer to the latter problem as the *Equilibrium Problem with Maximum Cost Constraint* (EPMAX).

For each communication model, we show how an instance of *EPMAX* (i.e., a network, a budget limit, and a payoff value) can be constructed in polynomial time from an instance of *PP*. We then show that *PP* is true (i.e., it has a solution A, B) if and only if the constructed *EPMAX* is true. Consequently, the *Equilibrium Problem with Maximum Cost Constraint* has to be at least as hard as the *Partition Problem*. Since the proof techniques follow the same lines for all models, we only give a detailed proof for the S-D model.

5.2 Efficient Algorithms for the Expected Cost Constraint

In this section, we show how the expected cost constrained game can be solved efficiently for the models introduced in Subsection 2.1. Recall that, in Subsection 2.3, we provided a derivation of the attacker's Nash equilibrium payoff in the unconstrained game model using the theory of blocking pairs of polyhedra. In this section, we use a similar derivation to show how polynomial-time algorithms can be devised to solve the game under an expected cost constraint.

First, by following the detailed analytical steps presented in [17, Chap. 4] for the unconstrained game, we can show that the attacker's equilibrium strategies correspond to the vertices of the blocker $bl(P_{\mathbf{\Lambda}})$ in the constrained game as well. In the constrained game, the definition of the polyhedron $P_{\mathbf{\Lambda}}$ in Equation (3) includes an additional linear inequality, which corresponds to the budget constraint. Since the expected cost $w(\boldsymbol{\alpha})$ defined in Equation (9) can also be expressed as $w(\boldsymbol{\alpha}) = \boldsymbol{w}'\mathbf{\Lambda}'\boldsymbol{\alpha}$, the constrained polyhedron can be written as

$$P_{\mathbf{\Lambda}} = \left\{ \boldsymbol{x} \in \mathbb{R}^{|E|}_{\geq 0} \;\middle|\; \exists \boldsymbol{\alpha} \in \mathbb{R}^{|\mathcal{T}|}_{\geq 0} \left(\mathbf{\Lambda}'\boldsymbol{\alpha} \leq \boldsymbol{x} \;\wedge\; \boldsymbol{\alpha}'\mathbf{1} \geq 1 \;\wedge\; \boldsymbol{w}'\mathbf{\Lambda}'\boldsymbol{\alpha} \leq b \right) \right\}. \quad (10)$$

Notice that the above definition of $P_{\mathbf{\Lambda}}$ involves the matrix $\mathbf{\Lambda}$, which is generally exponential in size. As a consequence, this definition of $P_{\mathbf{\Lambda}}$ cannot be used directly to solve the game efficiently.

To derive a polynomial-time solution for the ECC model, we first characterize the blocker $bl(P_{\mathbf{\Lambda}})$ of $P_{\mathbf{\Lambda}}$ using a set of linear equations whose cardinality is polynomial in the size of the network. We do so by showing that if a polynomial-size characterization exists for the unconstrained polyhedron, then there also exists one for the blocker of the expected cost constrained polyhedron. We then show how one can use linear programming tools to efficiently compute the equilibrium payoff based on a polynomial-size characterization of the blocker. Finally, we provide a characterization for each of the models discussed in Subsection 2.1.

Assume that the polyhedron $P_{\mathbf{\Lambda}}$ of the unconstrained game has a polynomial-size linear characterization

$$P_{\mathbf{\Lambda}} = \{\boldsymbol{x} \mid \exists \boldsymbol{f} \, (\boldsymbol{S}\boldsymbol{f} \leq \boldsymbol{x} \wedge \boldsymbol{C}\boldsymbol{f} \geq \boldsymbol{d})\} \ , \tag{11}$$

where $\boldsymbol{f} \in \mathbb{R}^k_{\geq 0}$ is a vector of polynomial length (i.e., k is a polynomial function of the network size), while vector $\boldsymbol{d}$ and matrices $\boldsymbol{S}$ and $\boldsymbol{C}$ are all constants of polynomial size. Then, the polyhedron associated with the expected cost constrained game is given by

$$P_{\mathbf{\Lambda}} = \{\boldsymbol{x} \mid \exists \boldsymbol{f} \, (\boldsymbol{S}\boldsymbol{f} \leq \boldsymbol{x} \wedge \boldsymbol{C}\boldsymbol{f} \geq \boldsymbol{d} \wedge \boldsymbol{w}'\boldsymbol{S}\boldsymbol{f} \leq b)\} \ . \tag{12}$$

The following theorem gives a polynomial-size characterization of the blocker in the expected cost constrained game.

Theorem 3. *The blocker of the polyhedron defined as*

$$P_{\mathbf{\Lambda}} = \{\boldsymbol{x} \mid \exists \boldsymbol{f} \, (\boldsymbol{S}\boldsymbol{f} \leq \boldsymbol{x} \wedge \boldsymbol{C}\boldsymbol{f} \geq \boldsymbol{d} \wedge \boldsymbol{w}'\boldsymbol{S}\boldsymbol{f} \leq b)\} \ , \tag{13}$$

where $\boldsymbol{f} \in \mathbb{R}^k_{\geq 0}$, $\boldsymbol{S} \in \mathbb{R}^{|E|\times k}_{\geq 0}$, $\boldsymbol{C} \in \mathbb{R}^{l\times k}_{\geq 0}$, *and* $\boldsymbol{d} \in \mathbb{R}^l_{\geq 0}$, *can be characterized as*

$$bl(P_{\mathbf{\Lambda}}) = \left\{\boldsymbol{y} \mid \exists K, \boldsymbol{g}, \boldsymbol{h} \left(\boldsymbol{g} \leq \boldsymbol{y} \wedge \boldsymbol{C}'\boldsymbol{h} \leq \boldsymbol{S}'\boldsymbol{w}K + \boldsymbol{S}'\boldsymbol{g} \wedge \boldsymbol{d}'\boldsymbol{h} - bK \geq 1\right)\right\} \ , \tag{14}$$

where $K \in \mathbb{R}_{\geq 0}$, $\boldsymbol{g} \in \mathbb{R}^{|E|}_{\geq 0}$, *and* $\boldsymbol{h} \in \mathbb{R}^l_{\geq 0}$.

The proof of the theorem can be found in Appendix C.

Recall that our goal is to efficiently compute the equilibrium payoff $\theta^* = \max\{\tilde{\theta}, 0\} = \max\{\max_{\boldsymbol{y}\in bl(P_{\mathbf{\Lambda}})} \theta(\boldsymbol{y}), 0\}$ of the game, which we use as a metric for topology vulnerability. The most straightforward solution would be to formulate this as a maximization problem subject to the set of linear constraints given by the above characterization of $bl(P_{\mathbf{\Lambda}})$. Unfortunately, the desired objective function $\theta = \frac{1}{\mathbf{1}'\boldsymbol{y}} (1 - \boldsymbol{\mu}'\boldsymbol{y})$ cannot be expressed as a linear function in $\boldsymbol{y}$ because of the division by $\mathbf{1}'\boldsymbol{y}$. Thus, to formulate the problem as a linear program, we "scale" our variables. We introduce a new variable ϕ, which is equal to $\frac{1}{\mathbf{1}'\boldsymbol{y}}$, and we divide the original variables and constants by $\mathbf{1}'\boldsymbol{y}$. The resulting linear program is:

$$\text{Maximize } \ \phi - \boldsymbol{\mu}'\boldsymbol{\beta} \tag{15}$$

subject to

$$\mathbf{1}'\boldsymbol{\beta} = 1 \tag{16}$$

$$\boldsymbol{g} \leq \boldsymbol{\beta} \tag{17}$$

$$\boldsymbol{C}'\boldsymbol{h} \leq \boldsymbol{S}'\boldsymbol{w}K + \boldsymbol{S}'\boldsymbol{g} \tag{18}$$

$$\boldsymbol{d}'\boldsymbol{h} - bK \geq \phi \ , \tag{19}$$

where $K, \phi \in \mathbb{R}_{\geq 0}$, $\boldsymbol{\beta}, \boldsymbol{g} \in \mathbb{R}^{|E|}_{\geq 0}$, and $\boldsymbol{h} \in \mathbb{R}^l_{\geq 0}$. Note that we let the scaled version of $\boldsymbol{y}$ be denoted by $\boldsymbol{\beta}$, as an optimal solution can be shown to be an equilibrium strategy for the adversary.

We now apply the above results to the Supply-Demand, All-to-One, and All-to-All communication models. Since the analysis follows the same lines for all models, we provide details only for the Supply-Demand model. For the All-to-One and All-to-All models, we describe the main points without providing details.

5.2.1 Characterizing the Blocker for the Supply-Demand Communication Model

In the case of the Supply-Demand model, we begin by finding a polynomial-size characterization of the unconstrained polyhedron $P_{\mathbf{\Lambda}}$. First, consider the restricted unconstrained polyhedron

$$\hat{P}_{\mathbf{\Lambda}} = \left\{ \boldsymbol{x} \in \mathbb{R}^{|E|}_{\geq 0} \;\middle|\; \exists \boldsymbol{\alpha} \in \mathbb{R}^{|\mathcal{T}|}_{\geq 0} \left(\mathbf{\Lambda}' \boldsymbol{\alpha} = \boldsymbol{x} \;\wedge\; \boldsymbol{\alpha}' \mathbf{1} = 1 \right) \right\} . \tag{20}$$

By comparing the above formula with definition of $P_{\mathbf{\Lambda}}$ (see Equation (3)), we can see two differences. First, $\hat{P}_{\mathbf{\Lambda}}$ uses convex linear combinations of the rows, while $P_{\mathbf{\Lambda}}$ uses any linear combination. Second, $\hat{P}_{\mathbf{\Lambda}}$ consists of the vectors $\boldsymbol{x}$ that are equal to a combination, while $P_{\mathbf{\Lambda}}$ consists of any vector $\boldsymbol{x}$ that is greater than or equal to a combination. From these differences, it follows readily that every element of $P_{\mathbf{\Lambda}}$ is the sum of an element of $\hat{P}_{\mathbf{\Lambda}}$ and a non-negative vector. Thus, if we have a polynomial-size characterization for $\hat{P}_{\mathbf{\Lambda}}$, we readily have one for $P_{\mathbf{\Lambda}}$. The following theorem characterizes $\hat{P}_{\mathbf{\Lambda}}$.

Theorem 4. *In the Supply-Demand communication model, the following hold.*

1. *For every convex linear combination $\boldsymbol{\alpha}$, the function $f^* : E \mapsto \mathbb{R}_{\geq 0}$ defined as $f^*(e) = \sum_{T \in \mathcal{T}} \alpha_T \lambda(T, e)$ is a feasible real-valued network flow.*
2. *For every feasible real-valued network flow $f^* : E \mapsto \mathbb{R}_{\geq 0}$, there is a convex linear combination $\boldsymbol{\alpha}$ such that, for every edge e, the flow value $f^*(e)$ is equal to $\sum_{T \in \mathcal{T}} \alpha_T \lambda(T, e)$.*

The proof of the theorem can be found in Appendix D.
Using the above theorem, we can characterize $P_{\mathbf{\Lambda}}$ as

$$P_{\mathbf{\Lambda}} = \left\{ \boldsymbol{x} \in \mathbb{R}^{|E|}_{\geq 0} \;\middle|\; \exists \text{ real-valued network flow } f \left(\forall e \in E : \; f(e) \leq x_e \right) \right\} , \tag{21}$$

which can formally be written as

$$P_{\mathbf{\Lambda}} = \left\{ \boldsymbol{x} \in \mathbb{R}^{|E|}_{\geq 0} \;\middle|\; \exists f : E \mapsto \mathbb{R}_{\geq 0} \Bigg(\forall e \in E : f(e) \leq x_e \wedge \forall v \in V : \sum_{(u,v) \in E} f(u,v) - \sum_{(v,w) \in E} f(v,w) = s(v) - d(v) \Bigg) \right\} . \tag{22}$$

Then, from Theorem 3, we have that the constrained blocker has the following polynomial-size characterization:

$$bl(P_{\mathbf{\Lambda}}) = \left\{ \boldsymbol{y} \in \mathbb{R}^{|E|}_{\geq 0} \;\middle|\; \exists \pi : V \mapsto \mathbb{R}, K \in \mathbb{R}_{\geq 0} \Big(\sum_{v \in V} \pi(v)(s(v) - d(v)) - bK \geq 1 \right.$$
$$\left. \wedge \; \forall e = (u,v) \in E : \pi(u) - \pi(v) \leq y_e + w_e K \Big) \right\}. \tag{23}$$

5.2.2 Characterizing the Blocker for the All-to-One Communication Model

In [27], it was shown that the polyhedron in the All-to-One model can be characterized using a set of multi-source flows. More specifically, for each element of the polyhedron, there exists a single-commodity flow that is dominated by the element and has the following properties: the designated node r is a sink with a demand of $N-1$, and every other node is a source with a supply of 1. Formally,

$$P_{\Lambda} = \left\{ \boldsymbol{x} \in \mathbb{R}^{|E|}_{\geq 0} \;\middle|\; \exists f : E \mapsto \mathbb{R}_{\geq 0} \Big(\forall e \in E : f(e) \leq x_e \right.$$
$$\left. \wedge \;\; \forall v \in V \setminus \{r\} : \sum_{(v,w) \in E} f(v,w) - \sum_{(u,v) \in E} f(u,v) \geq 1 \Big) \right\}. \tag{24}$$

Then, from Theorem 3, we have that the expected cost constrained blocker has the following polynomial-size characterization:

$$bl(P_{\mathbf{\Lambda}}) = \left\{ \boldsymbol{y} \in \mathbb{R}^{|E|}_{\geq 0} \;\middle|\; \exists \pi : V \setminus \{r\} \mapsto \mathbb{R}_{\geq 0}, K \in \mathbb{R}_{\geq 0} \Big(\sum_{v \in V} \pi(v) - bK \geq 1 \right.$$
$$\left. \wedge \;\; \forall e = (u,v) \in E : \pi(u) - \pi(v) \leq y_e + w_e K \Big) \right\}, \tag{25}$$

where $\pi(r) \equiv 0$ by definition to simplify the equation.

5.2.3 Characterizing the Blocker for the All-to-All Communication Model

In [28], it was shown that the polyhedron in the All-to-All model be characterized using a set of multi-commodity flows. In this characterization, there exists a dominated multi-commodity flow for each element of the polyhedron. In each flow, there are $|V|$ commodities corresponding to the nodes of the network. For

each commodity, the corresponding node is a sink, while all the other nodes are sources with a uniform (but variable) production. Finally, the total amount of transported flow has to be at least 1. Formally,

$$\hat{P_\Lambda} = \Bigg\{ \boldsymbol{x} \in \mathbb{R}_{\geq 0}^{|E|} \;\Bigg|\; \exists f : V \times E \mapsto \mathbb{R}_{\geq 0}, \boldsymbol{\alpha} \in \mathbb{R}_{\geq 0}^{|V|} \Bigg(\sum_{r \in V} \alpha_r \geq 1 \\ \wedge \forall r \in V, v \in V \setminus \{r\} : \sum_{\{u,v\} \in E} f_r(v,u) - f_r(u,v) \geq \alpha_r \\ \wedge \forall \{u,v\} = e \in E : x_e \geq \sum_{r \in V} f_r(u,v) + f_r(v,u) \Bigg) \Bigg\} . \quad (26)$$

Then, from Theorem 3, we have that the expected cost constrained blocker has the following polynomial-size characterization:

$$bl(P_{\boldsymbol{\Lambda}}) = \Bigg\{ \boldsymbol{y} \in \mathbb{R}_{\geq 0}^{|E|} \;\Bigg|\; \exists \pi : V \times V \mapsto \mathbb{R}_{\geq 0}, K \in \mathbb{R}_{\geq 0} \Bigg(\\ \forall r \in V : \sum_{v \in V} \pi_r(v) - bK \geq 1 \wedge \\ \forall r \in V, e = (u,v) \in E : |\pi_r(u) - \pi_r(v)| \leq y_e + w_e K \Bigg) \Bigg\} , \quad (27)$$

where $\pi_r(r) \equiv 0$ by definition to simplify the equation.

6 Related Work on Network Topology Robustness

In this section, we discuss some of the related work on assessing and quantifying the robustness of network topologies. First, we provide a brief overview of results from complex-network theory, which are concerned with the robustness of certain classes of networks. Then, we discuss some graph-theory based metrics, such as connectivity and toughness. Finally, we give examples of metrics which are derived by making strategic assumptions about the adversaries.

6.1 Robustness of Complex Networks

The question of network topology robustness against random faults and non-random attacks has attracted considerable interest from the complex-networks community. Albert, Jeong, and Barabási use simulations to study the error-tolerance of two classes of networks: scale-free networks and exponential networks [1]. Scale-free networks, such as the Internet or social networks, have degree distributions which decay according to power-laws. On the other hand, the degree distributions of exponential networks, such as the Erdős-Rényi (E-R)

random graph model [13] or the Watts-Strogatz small-world model [40], decay exponentially. The authors find that scale-free networks display an unexpected degree of robustness against random faults, that is, against the removal of a random subset of the nodes. However, these networks are very vulnerable to non-random attacks, which remove the highest-degree nodes. On the other hand, exponential networks are less robust against random faults, but more resilient to attacks.

Callaway et al. study percolation on graphs with general degree distribution [5]. Percolation models on random graphs can represent processes that remove network nodes in a random or targeted manner, which can model the failure of internet routers or power transmission lines. Assuming general degree distribution, the authors give exact solutions for a variety of cases, including uniform site percolation (i.e., random node removal), uniform bond percolation (i.e., uniform edge removal), and models in which probabilities depend on node degree. These exact solutions can be used to predict the behavior of networked systems under quite general types of breakdowns and interference, and they confirm previous results on the resilience of scale-free networks to random-faults and non-random attacks.

Cohen et al. also study the robustness of scale-free graphs against random faults and intentional attacks using percolation theory [8, 9]. In the case of random faults, the authors establish a general condition for the critical fraction of nodes that need to be removed before the network disintegrates. By applying their analysis to the physical structure of the Internet, they find that it is impressively robust, with a critical fraction above 99%. In the case of intentional attacks, they confirm that scale-free graphs are not as robust as against random faults, since the critical fraction is much lower. Furthermore, they show that the disruptive effects of intentional attacks become relevant even before the critical threshold is reached, as the average distance between nodes in the largest cluster is substantially higher near the threshold.

Bollobás and Riordan consider the robustness of linearized chord diagram (LCD) graphs, a more rigorously defined version of the Barabási-Albert model [4]. Their findings can be summarized as follow: Against random node removal, LCD graphs are much more robust than E-R graphs with the same number of edges. However, against malicious attacks, which remove nodes with higher degrees, LCD graphs are more vulnerable than E-R graphs.

Moreira et al. examine the robustness of scale-free networks against a range of attacks from random faults to intentional attacks [34]. In their model, it is assumed that the probability that an edge remains intact depends on the degrees of the adjacent nodes. Then, by varying the level of dependence, one can interpolate between random faults and non-random attacks. The authors show that, in their model, the critical percolation threshold, at which connectivity is lost, depends on both the degree distribution and the randomness level of the failures. Consequently, network robustness can be controlled through adjusting the topological bias in the failure process.

Paul et al. consider the problem of maximizing the robustness of networks while keeping their costs constant [36]. Their study provides network design

guidelines which maximize robustness against both random failures and non-random attacks while keeping the average degree of the network constant. They find optimal parameters for scale-free networks (i.e., networks having degree distributions with a single power-law regime), networks having degree distributions with two power-law regimes, and networks having degree distributions with two peaks. For these network models, they show that the optimal network design is the one in which all nodes except one have the same degree and one node has a very large degree.

While the work of Paul et al. provides guidelines for creating robust networks from scratch, Beygelzimer et al. consider the problem of improving the robustness of an existing network without substantial modifications [3]. In this work, robustness is measured as either the size of the largest connected component or the shortest path length between pairs of nodes after an attack (i.e., deleting the nodes with the highest degrees) or random faults (i.e., removing a random subset of nodes). The authors present empirical results showing how robustness is affected by various strategies for rewiring and creating edges, such as random addition and preferential rewiring. Based on these results, they conclude that a modest alteration of an initially scale-free network can substantially improve its robustness against attacks, and they identify the most effective modification strategies.

In contrast to our work, these studies are mostly concerned with the robustness of certain classes of networks, not with specific given topologies. Also contrary to our work, their measures of robustness are not primarily motivated by usage models, such as the All-to-One or S-D models in our work. Finally, they do not consider the strategic interaction between the defender and the attacker, which we have captured in this paper by using a game-theoretic model.

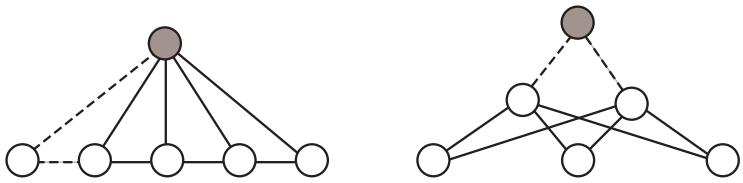
6.2 Graph-Theoretic Metrics

Connectivity The vertex-connectivity (or edge-connectivity) of a graph measures the minimum number of vertices (or edges) that have to be removed in order to disconnect the graph [15]. More formally, a graph is said to be k-vertex-connected (or k-edge-connected) if it remains connected whenever fewer than k vertices (or edges) are removed, and vertex-connectivity (or edge-connectivity) of a graph is defined as the largest k for which it is k-vertex-connected (or k-edge-connected). Connectivity has many appealing theoretical properties, such as being closely related to the number of independent paths between vertices (see Menger's theorem [15]). Furthermore, the problem of computing the connectivity of a graph can be solved in polynomial time.

Connectivity is very widely used as a measure for the robustness of network topologies. For example, in the case of wireless sensor networks, vertex- and edge-connectivity are undoubtedly the most prevalent metrics [29, 41, 33, 24, 42, 21]. For another example in support of using connectivity, see [12].

Unfortunately, as a metric for robustness against strategic attacks, connectivity suffers from a number of weaknesses. Firs, connectivity is only concerned

with the size of smallest disconnecting attack. In practice, however, the maximum attack-size that should be anticipated – in terms of the number of vertices (or edges) that the adversary can remove from the network – may be difficult to estimate. Second, connectivity is only concerned with whether an attack disconnects a network or not. In other words, connectivity does not take into account how disintegrated the network becomes as a result of an attack. In practice, however, an operator needs to care about how much functionality is retained by the network after an attack. Finally, similarly to the studies discussed in the previous subsection, connectivity does not build on a communication model (i.e., type of connectivity to be achieved). As a consequence, it performs suboptimally for specific cases, such as the ones shown in the examples of Figure 2 and Figure 7 below.



(a) (b)

Figure 7: Illustration of connectivity not characterizing the robustness of sensor network topologies well. The edge-connectivity of both graphs is 2, and thus, they are equally robust in terms of edge-connectivity. However, when the two dashed edges are removed, only a single vertex is separated from the sink in graph (a), while all of the vertices are separated from the sink in graph (b).

In this example, each of the two graphs represents a sensor network, in which the operator's objective is to transfer measurement data from the sensor nodes to the sink node, represented by the shaded vertex. Both of these graphs have an edge-connectivity of two, and therefore, they are supposed to be equally robust. However, by removing two edges, we can separate at most one vertex from the sink in graph (a), while we can separate all vertices in graph (b) (the dashed edges represent such attacks in the figure). In other words, a strategic attack removing two edges can only slightly affect graph (a) but can almost completely disable graph (b). Hence, we can hardly say that the networks are equally robust.

Using our proposed metric, the vulnerability of graph (a) is $\theta^* = 1$, and the targeted links are the five links connecting the sink to the five sensor nodes, each with probability $\frac{1}{5}$. For graph (b), $\theta^* = \frac{5}{2} > 1$, with the targeted links being the two links connecting the sink to its two neighbors, each attacked with probability $\frac{1}{2}$. In other words, sensor network (b) is more vulnerable than sensor network (a), which is more intuitive than what is suggested using connectivity.

Toughness Graph *toughness* is another well-known topology robustness metric with several theoretical results [2]. The toughness of a graph measures the minimum of the ratio between the number of vertices removed and the number of components in the resulting graph. Unfortunately, the problem of computing the toughness of a graph is NP-hard. Hence, it is not very well-suited for general practical use, especially when one is concerned with large graphs.

Strength Graph *strength* is a metric that is very similar to graph toughness. The strength of a graph measures the minimum of the ratio between the number of edges removed and the increase in the number of components in the resulting graph [10]. Intuitively, one can think of strength as the "edge-attack version" of toughness. However, unlike toughness, the strength of a graph can be computed in polynomial time. Compared to connectivity, the advantage of graph strength as a robustness metric is that it considers attacks of various sizes due to the fact that the minimum is taken over all possible edge removal attacks.

Summary While the above metrics are appealing because of their simplicity and/or analytical tractability, it is very hard to argue about how well they capture the notion of robustness. The main reason for this is that they approach the problem of quantifying robustness from a purely graph-theoretic perspective; hence, the corresponding attacker models have to be derived from the metrics and – as a results – usually assume that the strategic nature of the adversary is very limited. In general, they suffer from the same weaknesses as connectivity.

6.3 Attacker-Model-Driven Studies

In this section, we list examples of previous work that approach the problem of studying robustness by starting with an attacker model and "defining" robustness with respect to this model. However, none of these studies consider the simultaneous and strategic decision making of the defender and the attacker, on which the game-theoretic framework of our network blocking games model is built. Rather, they assume that the attacker will chose from a set of elementary strategies, such as removing the nodes with the highest betweenness, and will not anticipate the defender's response.

Dall'Asta et al. study the robustness of networks against various strategies that remove the most central nodes in the network [11]. These strategies are based on ranking the nodes using degree, strength, outreach, distance strength, topological betweenness, and weighted betweenness (for the definitions of these metrics, see [11]). To quantify the damage sustained by a network after an attack removed some nodes, the authors introduce three metrics: the ratio between the total node strength of the damaged network's largest component and that of the intact network, the ratio of total node outreach, and the ratio of total node distance strength. Their study shows that complex networks are more fragile than expected form the analysis of topological quantities when the traffic characteristics are taken into account.

Estrada studies the property of graphs being both sparse and highly connected, which is known as "good expansion" (GE) [14]. Using spectral graph theory, the author introduces a new metric for measuring the good expansion of networks, and classifies 51 real-world networks as being GE or non-GE. By comparing the networks based on their robustness against intentional attacks against nodes, the author argues that being GE and having uniform degree distribution makes networks robust.

Holme et al. study the resilience of complex networks to attacks targeting nodes and edges [23]. They evaluate several existing network models using attack strategies that are based on removing nodes in descending order of either their degree or their betweenness centrality. Their study shows that the Erdős-Rényi random graph model is the most robust of the evaluated models.

7 Conclusions & Future Work

In this paper, we have considered the problem of finding metrics for the vulnerability (or robustness) of network topologies in adversarial environments. We have proposed a metric derived from our previously introduced network blocking game (NBG) models and studied its computational complexity. We have also discussed the properties of the metric in several examples.

In previous NBG models, the network operator was assumed to be interested only in security. We have generalized the models by considering a situation where, in addition to security, the network operator takes other economic and/or technical goals into consideration. We have modeled these additional goals as budget constraints on the operator and have studied two constraint formulations: the maximum and the expected cost constraints. We have shown that the maximum cost formulation leads to NP-hard problems and proposed efficient solutions for the expected cost formulation. Using these formulations, we have derived the optimal vulnerability/cost tradeoff curve and have applied our tradeoff analysis to two real-life network topologies.

Several future directions are being considered as follow ups of this work. First, we conjecture that there exists a class of games that, despite an exponential-size payoff matrix, can be solved efficiently. Early results indicate that for such a class, the equilibrium payoff satisfies certain properties such as submodularity and subadditivity. Second, in this paper, we have assumed that the adversary can attack only one link at a time. In general, an attacker could have the capability to attack two or more network resources in one shot. We plan to study such multiple elements attack scenarios in the future. Another natural extension of this study is the case where several communication models are run on top of the same underlying network resources. Finally, in this study, we have only considered failures that are due to the actions of a malicious attacker. However, in reality, failures can also be due to random events. A more complete study should concurrently consider both types of failures.

Acknowledgments

This paper has been partially supported by NIST Cooperative Agreement No. 70NANB13H012 with the University of Maryland while one of the authors was a guest researcher in the NIST Applied and Computational Mathematics Division. The authors would like to thank Dr. Kevin Mills (NIST) for providing data for the Abilene and the ISP3 networks.

References

[1] Réka Albert, Hawoong Jeong, and Albert-László Barabási. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, 2000.

[2] Douglas Bauer, Hajo Broersma, and Edward Schmeichel. Toughness in graphs - A survey. *Graphs and Combinatorics*, 22(1):1–35, April 2006.

[3] Alina Beygelzimer, Geoffrey Grinstein, Ralph Linsker, and Irina Rish. Improving network robustness by edge modification. *Physica A: Statistical Mechanics and its Applications*, 357(3):593–612, 2005.

[4] Béla Bollobás and Oliver Riordan. Robustness and vulnerability of scale-free random graphs. *Internet Mathematics*, 1(1):1–35, 2003.

[5] Duncan S Callaway, Mark EJ Newman, Steven H Strogatz, and Duncan J Watts. Network robustness and fragility: Percolation on random graphs. *Physical Review Letters*, 85(25):5468–5471, Dec 2000.

[6] Xi Chen and Xiaotie Deng. Settling the complexity of two-player nash equilibrium. In *Proceedings of the 47th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 261–272, October 2006.

[7] Fan Chung. Laplacians and the Cheeger inequality for directed graphs. *Annals of Combinatorics*, 9(1):1–19, 2005.

[8] Reuven Cohen, Keren Erez, Daniel Ben-Avraham, and Shlomo Havlin. Resilience of the internet to random breakdowns. *Physical Review Letters*, 85(21):4626–4628, Nov 2000.

[9] Reuven Cohen, Keren Erez, Daniel Ben-Avraham, and Shlomo Havlin. Breakdown of the internet under intentional attack. *Physical Review Letters*, 86(16):3682–3685, Apr 2001.

[10] William H. Cunningham. Optimal attack and reinforcement of a network. *Journal of the ACM*, 32(3):549–561, 1985.

[11] Luca Dall'Asta, Alain Barrat, Marc Barthélemy, and Alessandro Vespignani. Vulnerability of weighted networks. *Journal of Statistical Mechanics*, 2006(04):P04006, 2006.

[12] Anthony H. Dekker and Bernard D. Colbert. Network robustness and graph topology. In *Proceedings of the 27th Australasian Conference on Computer Science - Volume 26*, ACSC '04, pages 359–368, Darlinghurst, Australia, Australia, 2004. Australian Computer Society, Inc.

[13] P Erdős and A Rényi. On random graphs I. *Publicationes Mathematicae*, 6:290–297, 1959.

[14] Ernesto Estrada. Network robustness to targeted attacks. The interplay of expansibility and degree distribution. *European Physical Journal B*, 52(4):563–574, 2006.

[15] Jonathan L Gross and Jay Yellen. *Graph theory and its applications*. CRC press, 2005.

[16] Tony H. Grubesic, Timothy C. Matisziw, Alan T. Murray, and Diane Snediker. Comparative approaches for assessing network vulnerability. *International Regional Science Review*, 31(1):88–112, 2008.

[17] Assane Gueye. *A Game Theoretical Approach to Communication Security*. PhD thesis, EECS Department, University of California, Berkeley, Mar 2011.

[18] Assane Gueye and Vladimir Marbukh. A game-theoretic framework for network security vulnerability assessment and mitigation. In *Proceedings of 3rd Conference on Decision and Game Theory for Security (GameSec)*. Springer, November 2012.

[19] Assane Gueye, Jean C. Walrand, and Venkat Anantharam. Design of network topology in an adversarial environment. In *Proceedings of 1st Conference on Decision and Game Theory for Security (GameSec)*, 2010.

[20] Assane Gueye, Jean C. Walrand, and Venkat Anantharam. A network topology design game: How to choose communication links in an adversarial environment? In *Proceedings of 2nd International ICST Conference on Game Theory for Networks (GameNets)*, 2011.

[21] Xiaofeng Han, Xiang Cao, Errol L. Lloyd, and Chien-Chung Shen. Fault-tolerant relay node placement in heterogeneous wireless sensor networks. In *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM)*, pages 1667–1675, May 2007.

[22] Hans J. Herrmann, Christian M. Schneider, André A. Moreira, José S. Andrade Jr., and Shlomo Havlin. Onion-like network topology enhances robustness against malicious attacks. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(01):P01027, 2011.

[23] Petter Holme, Beom Jun Kim, Chang No Yoon, and Seung Kee Han. Attack vulnerability of complex networks. *Physical Review E*, 65(5):056109, 2002.

[24] Abhishek Kashyap, Samir Khuller, and Mark Shayman. Relay placement for higher order connectivity in wireless sensor networks. In *Proceedings of the 25th IEEE International Conference on Computer Communications (INFOCOM)*, pages 1–12, April 2006.

[25] Aron Laszka and Assane Gueye. Quantifying All-to-One network topology robustness under budget constraints. In *Proceedings of Workshop on Pricing and Incentives in Networks and Systems (W-PIN+NetEcon)*. ACM, June 2013.

[26] Aron Laszka and Assane Gueye. Quantifying network topology robustness under budget constraints: General model and computational complexity. In *Proceedings of 4th Conference on Decision and Game Theory for Security (GameSec)*, pages 154–174, November 2013.

[27] Aron Laszka, Dávid Szeszlér, and Levente Buttyán. Game-theoretic robustness of many-to-one networks. In *Proceedings of 3rd International ICST Conference on Game Theory for Networks (GameNets)*, 2012.

[28] Aron Laszka, Dávid Szeszlér, and Levente Buttyán. Linear loss function for the network blocking game: An efficient model for measuring network robustness and link criticality. In *Proceedings of 3rd Conference on Decision and Game Theory for Security (GameSec)*, 2012.

[29] Ji Li, Lachlan L. H. Andrew, Chuan H. Foh, Moshe Zukerman, and Hsiao-Hwa Chen. Connectivity, coverage and placement in wireless sensor networks. *Sensors*, 9(10):7664–7693, 2009.

[30] Stephan Mertens. The easiest hard problem: Number partitioning. *Computational Complexity and Statistical Physics*, 125(2):125–139, 2006.

[31] Kevin L. Mills, J. Filliber, D-Y. Cho, and Edward J. Schwartz. Predicting macroscopic dynamics in large distributed systems. In *Proceedings of the ASME 2011 Conference on Pressure Vessels & Piping*, July 2011.

[32] Kevin L. Mills, Edward J. Schwartz, and Jian Yuan. How to model a TCP/IP network using only 20 parameters. In *Proceedings of the 2010 Winter Simulation Conference (WSC)*, pages 849–860, December 2010.

[33] Satyajayant Misra, Seung D. Hong, Guoliang Xue, and Jian Tang. Constrained relay node placement in wireless sensor networks to meet connectivity and survivability requirements. In *Proceedings of the 27th IEEE International Conference on Computer Communications (INFOCOM)*, pages 281–285, April 2008.

[34] Andre A Moreira, José S Andrade Jr, Hans J Herrmann, and Joseph O Indekeu. How to make a fragile network robust and vice versa. *Physical Review Letters*, 102(1):018701, Jan 2009.

[35] Edgar M. Palmer. On the spanning tree packing number of a graph: A survey. *Discrete Mathematics*, 230(1):13–21, 2001.

[36] G Paul, T Tanizawa, S Havlin, and HE Stanley. Optimization of robustness of complex networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 38(2):187–191, 2004.

[37] Gerald Paul, Sameet Sreenivasan, Shlomo Havlin, and H Eugene Stanley. Optimization of network robustness to random breakdowns. *Physica A: Statistical Mechanics and its Applications*, 370(2):854–862, 2006.

[38] Christian M. Schneider, André A. Moreira, José S. Andrade, Shlomo Havlin, and Hans J. Herrmann. Mitigation of malicious attacks on networks. *Proceedings of the National Academy of Sciences*, 108(10):3838–3841, 2011.

[39] Toshi Tanizawa, Gerald Paul, Reuven Cohen, Shlomo Havlin, and H Eugene Stanley. Optimization of network robustness to waves of targeted and random attacks. *Physical Review E*, 71(4):047101, 2005.

[40] Duncan J Watts and Steven H Strogatz. Collective dynamics of small-worldnetworks. *Nature*, 393(6684):440–442, 1998.

[41] Mohamed Younis and Kemal Akkaya. Strategies and techniques for node placement in wireless sensor networks: A survey. *Ad Hoc Networks*, 6(4):621–655, 2008.

[42] Weiyi Zhang, Guoliang Xue, and Satyajayant Misra. Fault-tolerant relay node placement in wireless sensor networks: Problems and algorithms. In *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM)*, pages 1649–1657, May 2007.

A Proof of Theorem 1

Proof. Given an instance $(\boldsymbol{c}, \boldsymbol{v}, C, V)$ of the Knapsack Problem, we construct an instance $(E, I_{T\in\mathcal{T}}, \lambda(T,e), p)$ of the Equilibrium Problem as follows.

- Let the set of elements be $E = \{1, \ldots, N\}$,
- let the feasibility testing function be $I_{T\in\mathcal{T}} = \begin{cases} \text{true} & \text{if } \sum_{i\in T} c_i \leq C \\ \text{false} & \text{otherwise,} \end{cases}$
- let the usage function be $\lambda(T,e) = \frac{1}{\sum_{i\in T} v_i}$,
- let the adversary's attack costs be $\boldsymbol{\mu} = \mathbf{0}$,
- let the threshold payoff value be $p = \frac{1}{V}$.

First, observe that we define the function $\lambda(T, e)$ such that its value does not depend on e. Consequently, the payoff of the game does not depend on the adversary's strategy, it only depends on the operator's strategy. To simplify our proof, we will let $\lambda(T)$ denote $\lambda(T, e)$ for any e.

It is easy to see that both $I_{T\in\mathcal{T}}$ and $\lambda(T)$ can be computed in polynomial time, as they only require computing the sum of a given set and then comparing it with a constant or calculating its reciprocal. Furthermore, every step of the reduction can also be carried out in time and space that is polynomial in the size of the Knapsack Problem instance. Hence, the reduction itself can be done in polynomial time.

We claim that the given instance of KP is true if and only if the above instance of EP is true. To prove this, we have to show that there exists a subset $S \subseteq \{1, \ldots, N\}$ whose sum weight is at most W and whose sum value is at least V if and only if the adversary's equilibrium payoff in the above game is less than or equal to p.

First, assume that there exists a subset S satisfying the constraints of the Knapsack Problem. Then, we have to show that the adversary's equilibrium payoff is at most p. Let $\boldsymbol{\alpha}^*$ be the mixed strategy that uses only subset S. Formally, let $\alpha^*_S = 1$ and, for every other subset $U \neq S$, let $\alpha^*_U = 0$. If the operator uses this strategy, her loss is

$$\lambda(S) = \frac{1}{\sum_{i\in S} v_i} = \frac{1}{V} = p \ , \tag{28}$$

regardless of the strategy of the adversary. Therefore, the operator's equilibrium loss and, hence, the adversary's equilibrium payoff have to be at most p.

Second, assume that there does not exist a subset satisfying the constraints of the Knapsack Problem. In this case, we have to show that the adversary's equilibrium payoff is greater than p. Since no subset satisfies the constraints of KP, we have that, for every $T \in \mathcal{T}$, $\sum_{i\in T} v_i < V$ and, hence,

$$\lambda(T) = \frac{1}{\sum_{i\in T} v_i} > \frac{1}{V} = p \ . \tag{29}$$

Consequently, the expected loss for any operator strategy $\boldsymbol{\alpha}$ is

$$\sum_{T\in\mathcal{T}} \alpha_T \underbrace{\lambda(T)}_{>p} > p \ . \tag{30}$$

Therefore, the adversary's equilibrium payoff has to be greater than p. □

B Proof of Theorem 2

In this appendix section, we provide a prof of Theorem 2, which states that computing the NE payoff of an NBG under a maximum cost budget constraint is NP-hard in the S-D, the All-to-All, and the All-to-One communication models.

Since the proof techniques follow the same lines for all communication models, we only give a detailed proof for the S-D model. For the All-to-One and All-to-All models, we describe the main points of the proofs without providing the details.

Proof. We show NP-hardness by reducing a well-known NP-hard problem, the *Partition Problem* (*PP*) [30], to the problem of deciding whether the equilibrium payoff in a given network under a maximum cost constraint is at most a certain value. We refer to the latter problem as the *Equilibrium Problem with Maximum Cost Constraint* (EPMAX). These computational problems are defined formally as follows.

Definition 3 (Partition Problem [PP])**.** Given a multiset of positive integers $\{x_1, \ldots, x_n\}$, is there a partitioning of the multiset into two disjoint subsets A and B such that $\sum_{x \in A} x = \sum_{x \in B} x$?

Definition 4 (Equilibrium Problem with Maximum Cost Constraint [EPMAX])**.** Given a communication model, a network G, a budget limit b, and a threshold payoff value p, is the adversary's equilibrium payoff less than or equal to p?

For each communication model, we show how an instance of *EPMAX* (i.e., a network, a budget limit, and a payoff value) can be constructed in polynomial time from an instance of *PP*. We then show that *PP* is true (i.e., it has a solution A, B) if and only if the constructed *EPMAX* is true. Consequently, the *Equilibrium Problem with Maximum Cost Constraint* has to be at least as hard as the *Partition Problem*.

To simplify the notations in our proofs, we define the *expected loss* of an edge $e \in E$ in a given operator strategy $\boldsymbol{\alpha}$ as

$$L(e) = \sum_{T \in \mathcal{T}} \alpha_T \lambda(T, e) \ . \tag{31}$$

From the definitions of the player's payoffs, it follows readily that the adversary's expected payoff is $L(e) - \mu_e$ and the operator's loss is $L(e)$ if the adversary uses the pure strategy e.

B.1 Proof of Theorem 2 for the Supply-Demand Communication Model

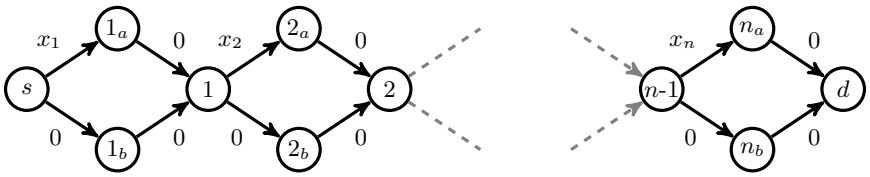


Figure 8: Illustration for the proof of Theorem 2 for the S-D model. Numbers along edges indicate unit costs.

Given an instance of *PP*, we build an instance of *EPMAX* as follows.

- Let the topology of the network be the following (see Figure 8 for an illustration): There is one source node, denoted by s, one sink node, denoted by d, and $3n-1$ other nodes, which are denoted by 1_a, 1_b, 1, 2_a, 2_b, 2, ..., n_a, and n_b.

  Node s is connected to nodes 1_a and 1_b with edges having unit costs of x_1 and 0, respectively. Nodes i_a and i_b, for $i < n$, are connected to node i with edges having zero unit cost. Node i is connected to node $(i+1)_a$ and $(i+1)_b$ with edges having unit costs of x_{i+1} and 0, respectively. Finally, nodes n_a and n_b are connected to node d with edges having zero unit cost.

- Let the capacity of the links and the amount of goods to be moved from s to d be 1.

- Let the operator's budget be $b = \frac{1}{2}\sum_{i=1}^{n} x_i$.

- Let the threshold payoff value be $p = \frac{1}{2}$.

We claim that the adversary's equilibrium payoff in the above network is greater than p if and only if *PP* does not have a solution.

First, we assume that the multiset $\{x_1, \ldots, x_n\}$ can be partitioned into two subsets A and B of equal sum, i.e., we assume that *PP* has a solution. In this case, we have to show that the equilibrium payoff is at most $\frac{1}{2}$. First, notice that, since the total amount of goods to be moved from s to d is 1 and the amount of flow on each edge is either 0 or 1, the set of feasible integer flows is equal to the set of directed s-d paths.

We now show that there exist two disjoint paths (or, equivalently, flows) that satisfy the operator's budget constraint. The first path (or, equivalently, the first set of links with positive flow values) consists of the edges $(i-1, i_a)$ and (i_a, i) for each $x_i \in A$, and of the edges $(i-1, i_b)$ and (i_b, i) for each $x_i \notin A$. The second path consists of the remaining edges. In other words, the first flow takes the "path above" whenever $x_i \in A$ and the "path below" whenever $x_i \notin A$, while the second flow does the contrary. By our assumption, the costs of the two flows are equal and given by $\sum_{x_i \in A} x_i = \sum_{x_i \in B} x_i = \frac{1}{2}\sum_i x_i = b$; thus, they both satisfy the maximum cost budget constraint. By assigning a probability of $\frac{1}{2}$ to each flow, we obtain an operator strategy for which the expected loss of every edge is at most to $\frac{1}{2}$. If the operator employs this strategy, then the adversary's payoff for every pure and, consequently, every mixed strategy is at most $\frac{1}{2}$. Therefore, the adversary's equilibrium payoff has to be at most $\frac{1}{2}$.

Second, we assume that the multiset $\{x_1, \ldots, x_n\}$ cannot be partitioned into two subsets of equal sum, that is, we assume that *PP* does not have a solution. In this case, we have to show that the adversary's equilibrium payoff is greater than $\frac{1}{2}$. If the equilibrium payoff were at most $\frac{1}{2}$, then there would exist an operator strategy $\boldsymbol{\alpha}$ in which the expected loss of every edge is at most $\frac{1}{2}$. We show that no such strategy can exist using contradiction.

First, suppose that the contrary holds, i.e., that there exists a strategy in which the expected loss of every edge is at most $\frac{1}{2}$. Because of the maximum

cost budget constraint, the cost of every pure strategy is less than or equal to $b = \frac{1}{2}\sum_i x_i$. Moreover, we can show that this inequality has to be *strict*. Every pure strategy is an *s-d* path, and if the cost of a path I were equal to b, then there would exist a subset of links $I \subsetneq \{1, 2, \ldots, n\}$ such that $\sum_{i \in I} x_i = b$. By letting $A = \{x_i \mid i \in I\}$ and $B = \{x_i \mid i \notin I\}$, we would get a solution for *PP*, which would contradict the assumption that the set cannot be partitioned. Thus, the cost of every pure strategy is strictly less than b and, as a consequence, the expected cost of every mixed strategy is also strictly less than b. Formally, we have

$$\sum_{e \in E} L(e) w_e < b = \frac{1}{2} \sum_{i=1}^{n} x_i = \sum_{e \in E} \frac{1}{2} w_e \ . \tag{32}$$

Now, observe that the expected loss $L(e)$ of an edge e in the S-D model is equal to the expected amount of flow on that edge. Since the total amount of goods to be moved is 1 and each pair of "above" and "below" edges is an *s-d* cut, the sum of the flows on any pair of above and below edges is equal 1. Thus, for every pair of above and below edges e_a and e_b, we have $L(e_a) + L(e_b) \geq 1 = \frac{1}{2} + \frac{1}{2}$. By combining this with the supposition that the expected loss of every edge is at most $\frac{1}{2}$, we have that

$$\forall e \in E : \ L(e) = \frac{1}{2} \ , \tag{33}$$

which implies that

$$\sum_{e \in E} L(e) w_e = \sum_{e \in E} \frac{1}{2} w_e \ . \tag{34}$$

However, this leads to a contradiction with Equation (32), which proves that there exists no operator strategy for which the expected loss of every edge is at most $\frac{1}{2}$. Therefore, if *PP* does not have a solution, then the equilibrium payoff is greater than $\frac{1}{2}$.

B.2 Proof of Theorem 2 for the All-to-One model

For the All-to-One communication model, we construct an instance of *EPMAX* from an instance of *PP* as follows.

- Let the network topology be the following (see Figure 9 for an illustration): There is a designated node r, which is connected to $2n$ nodes (denoted by 1_a, 1_b, 2_a, 2_b, …, n_a and n_b) in the form of a large star rooted at r. Furthermore, there are n "outer" nodes, which are denoted by 1, 2, …, n. Node i is connected to nodes i_a and i_b with edges having unit costs of x_i and 0. The edges connecting nodes i_a and i_b to r both have zero unit cost.
- Let the operator's budget be $b = \frac{1}{2}\sum_{i=1}^{n} x_i$.
- Let the threshold payoff value be $p = \frac{3}{2}$.

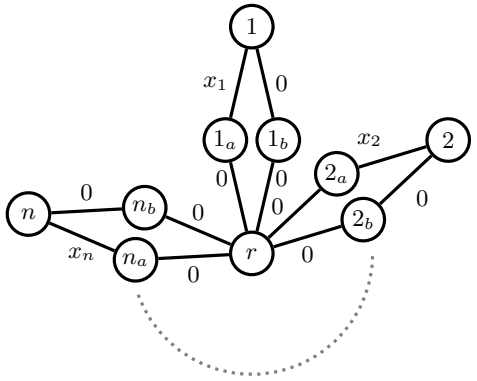


Figure 9: Illustration for the proof of Theorem 2 for the All-to-One model.

We claim that the adversary's equilibrium payoff in the above network is greater than $\frac{3}{2}$ iff *PP* does not have a solution.

We first assume that *PP* has a solution (A, B), and use it to derive an operator strategy for which the expected loss of every edge is at most $\frac{3}{2}$. For that, we give a randomized algorithm for choosing pure strategies, and use the distribution of its output as the operator's mixed strategy. The algorithm for choosing pure strategies (i.e., spanning trees) is the following. First, select all the edges connected to r (i.e., use them with a probability of 1), which form a star network. Then, choose either A or B with equal probability. Finally, connect each outer node i to the star as follows: if x_i belongs to the chosen set, then use the edge that has cost x_i; otherwise, use the other edge. Notice that this algorithm randomly chooses one out of two spanning trees (with probabilities $\frac{1}{2}$ and $\frac{1}{2}$).

We now show that the expected loss of every edge is at most $\frac{3}{2}$. First, each outer edge e is used with probability $\frac{1}{2}$, and its removal cuts off at most 1 node. Hence, we have $L(e) = \frac{1}{2}$ for these outer edges. Second, each inner edge e is used with probability 1, and the number of nodes cut off by its removal is 1 with probability $\frac{1}{2}$ (when the corresponding outer node is not connected to r through it) and 2 with probability $\frac{1}{2}$ (when the corresponding outer node is connected through it). Hence, we have $L(e) = \frac{3}{2}$ for inner edges. Therefore, we have that $L(e) \leq \frac{3}{2}$ for every edge e, which proves that *EPMAX* is true if *PP* has a solution.

Now, assume that *PP* does not have a solution. Then, we have that the cost of every pure strategy is strictly less than b, which implies

$$\sum_{e \in E_{\text{outer}}} w_e L(e) < b = \frac{1}{2} \sum_i x_i = \sum_{e \in E_{\text{outer}}} \frac{1}{2} w_e \ , \tag{35}$$

where E_{outer} is the set of outer edges. For a pair of edges $e_a = (i_a, r)$ and $e_b = (i_b, r)$, we can easily show that $L(e_a) + L(e_b) = 3$. If there were an operator strategy in which the expected loss of every edge was at most $\frac{3}{2}$, it

would follow that $L(e_a) = L(e_b) = \frac{3}{2}$, which would imply that expected loss of every outer edge is at least $\frac{1}{2}$. However, this would lead to a contradiction with Equation (35); thus, no such strategy can exist. Therefore, if *PP* does not have a solution, then *EPMAX* is not true.

B.3 Proof of Theorem 2 for the All-to-All model

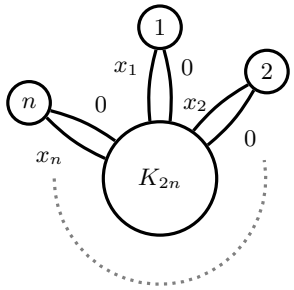


Figure 10: Illustration for the proof of Theorem 2 for the All-to-All model.

For the All-to-All communication model, we construct an instance of *EPMAX* from an instance of *PP* as follows.

- Let the network topology be the following (see Figure 10 for an illustration): There is a large clique that consists of $2n$ nodes, and there are n "outer" nodes, to which we refer as node 1, node 2, ..., node n. Each node i, where $i = 1, \ldots, n$, is connected to two distinct nodes of the clique with edges having unit costs of x_i and 0, such that every node in the clique is connected to exactly one outer node. Finally, edges between two nodes inside the clique have zero unit cost.
- Let the operator's budget be $b = \frac{1}{2} \sum_{i=1}^{n} x_i$.
- Let the threshold payoff value be $p = \frac{1}{2}$.

We claim that the adversary's equilibrium payoff in the above network is greater than $\frac{1}{2}$ iff *PP* does not have a solution.

We first assume that *PP* has a solution (A, B), and use it to derive an operator strategy in which the expected loss of every edge is at most $\frac{1}{2}$. For that, we give a randomized algorithm for selecting pure strategies, and use the distribution of its output as the operator's mixed strategy. The algorithm for selecting pure strategies (i.e., spanning trees) is the following. First, choose uniformly at random a star subgraph of the K_{2n} clique. Second, choose either set A or set B with equal probability (that is, choose them at random with probabilities $\frac{1}{2}$ and $\frac{1}{2}$). Finally, connect each outer node i to the star with exactly one edge: if x_i belongs to the chosen set, then use the edge that has cost x_i; otherwise, use the other edge.

We now show that the expected loss of every link is at most $\frac{1}{2}$ if the operator uses this randomized algorithm as her mixed strategy. First, each outer edge e is selected with probability $\frac{1}{2}$, and its removal cuts off one node if it has been selected. Hence, we have $L(e) = \frac{1}{2}$ for these outer links. Second, each link e inside the clique is used with probability $\frac{1}{n}$ (the probability that a randomly chosen star subgraph contains it), and its removal cuts off at most two nodes. Hence, we have $L(e) \leq \frac{2}{n}$ for these inner links[7], which implies that $L(e) \leq \frac{1}{2}$. Therefore, if *PP* is true, then so is *EPMAX*.

Next, we assume that *PP* does not have a solution, and use the same argument as before to show that the cost of every pure strategy and, hence, the expected cost of every mixed strategy is strictly less than b. Formally, we have

$$\sum_{e \in E_{\text{outer}}} w_e L(e) < b = \frac{1}{2} \sum_i x_i = \sum_{e \in E_{\text{outer}}} \frac{1}{2} w_e \ , \tag{36}$$

where E_{outer} is the set of outer links. Now, consider an arbitrary pair of edges e_a and e_b which connect the same outer node to the clique. It can be shown easily that $L(e_a) + L(e_b) \geq 1$. If there were an operator strategy in which the expected loss of every edge were at most $\frac{1}{2}$, then it would follow that $\forall e \in E_{\text{outer}} : \ L(e) = \frac{1}{2}$. However, this would lead to a contradiction with Equation (36); thus, no such strategy can exist. Therefore, if *PP* is not true, then neither is *EPMAX*.

□

C Proof of Theorem 3

Proof. We prove Equation (14) in two steps:

- Right-hand side (RHS) of Equation (14) $\subseteq bl(P_{\boldsymbol{\Lambda}})$: We have to show that every element of the RHS of (14) is an element of the blocker $bl(P_{\boldsymbol{\Lambda}})$. Let $\tilde{\boldsymbol{y}}$ be an arbitrary element of the RHS, that is, a vector which satisfies the constraints of the RHS with some $\tilde{\boldsymbol{g}}$, $\tilde{\boldsymbol{h}}$, and $\tilde{K}$. To prove that $\tilde{\boldsymbol{y}} \in bl(P_{\boldsymbol{\Lambda}})$, we show that $\tilde{\boldsymbol{y}}'\boldsymbol{x} \geq 1$ for every $\boldsymbol{x} \in P_{\boldsymbol{\Lambda}}$. To this end, we formulate the following linear programming problem and show that its value is greater than or equal to 1:

  $$\text{Minimize } \tilde{\boldsymbol{y}}'\boldsymbol{x} \tag{37}$$

  subject to

  $$\boldsymbol{w}'\boldsymbol{S}\boldsymbol{f} \leq b \tag{38}$$
  $$\boldsymbol{S}\boldsymbol{f} \leq \boldsymbol{x} \tag{39}$$
  $$\boldsymbol{C}\boldsymbol{f} \geq \boldsymbol{d} \ , \tag{40}$$

  where $\boldsymbol{f} \in \mathbb{R}^k_{\geq 0}$ and $\boldsymbol{x} \in \mathbb{R}^{|E|}_{\geq 0}$.

  Observe that the constraints of the linear program correspond to the characterization of $P_{\boldsymbol{\Lambda}}$. Consequently, the above linear program's set of feasible

[7] Note that we can assume $n \geq 4$ for the reduction.

solutions projected to $\boldsymbol{x}$ is actually $P_{\boldsymbol{\Lambda}}$. Therefore, it suffices to show that the value of the linear program is at least 1. To see this, consider the dual linear program:

$$\text{Maximize } \boldsymbol{d}'\boldsymbol{h} - bK \tag{41}$$

subject to

$$\boldsymbol{g} \leq \tilde{\boldsymbol{y}} \tag{42}$$

$$\boldsymbol{C}'\boldsymbol{h} \leq \boldsymbol{S}'\boldsymbol{w}K + \boldsymbol{S}'\boldsymbol{g} \ , \tag{43}$$

where $K \in \mathbb{R}_{\geq 0}$, $\boldsymbol{g} \in \mathbb{R}_{\geq 0}^{|E|}$, and $\boldsymbol{h} \in \mathbb{R}_{\geq 0}^{l}$.

Since $\tilde{\boldsymbol{y}}$ satisfies the constraints of the RHS of Equation (14) with $\tilde{K}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{h}}$, we have that $(\tilde{K}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{h}})$ is a feasible solution. Furthermore, we also have that the objective function for this solution is at least 1. Thus, the value of the dual program has to be at least 1. From linear programming duality, it follows readily that the value of the primal program is also at least 1, which proves that $\tilde{\boldsymbol{y}}$ blocks every element of the polyhedron P_{Λ}.

- $bl(P_{\boldsymbol{\Lambda}}) \subseteq$ RHS of Equation (14): We have to show that every $\tilde{\boldsymbol{y}} \in bl(P_{\boldsymbol{\Lambda}})$ satisfies the constraints of the RHS. To see this, first consider the linear program from the first part of the proof. Since $\tilde{\boldsymbol{y}}$ blocks every $\boldsymbol{x} \in P_{\boldsymbol{\Lambda}}$, we have that the value of the linear program and its dual is at least 1. Now, consider an optimal solution $(\tilde{K}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{h}})$ of the dual linear program. Since the value of the dual linear program is at least 1, we have that $1 \leq \boldsymbol{d}'\tilde{\boldsymbol{h}} - b\tilde{K}$. Furthermore, we also have $\tilde{\boldsymbol{g}} \leq \tilde{\boldsymbol{y}}$ and $\boldsymbol{C}'\tilde{\boldsymbol{h}} \leq \boldsymbol{S}'\boldsymbol{w}\tilde{K} + \boldsymbol{S}'\tilde{\boldsymbol{g}}$ from the constraints of the linear program. Thus, $\tilde{\boldsymbol{y}}$ satisfies the constraints of the RHS of Equation (14) with $\tilde{K}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{h}}$.

□

D Proof of Theorem 4

Proof. We prove the two cases separately.

1. We have to show that f^* satisfies the flow conservation constraints. Recall that, in the S-D model, a pure strategy $T \in \mathcal{T}$ is actually an integer flow. In this proof, we will use the conventional notation for network flows, and represent each pure strategy by a function $f : E \mapsto \mathbb{R}_{\geq 0}$, where $f(u, v)$ is the flow along edge (u, v). Hence, for a given edge $e = (u, v)$, $\sum_{T \in \mathcal{T}} \alpha_T \lambda(T, e)$ can be written as $\sum_{f \in \mathcal{T}} \alpha_T f(u, v)$. Then, for each $v \in V$,

we have

$$\sum_{(v,w)\in E} f^*(v,w) - \sum_{(u,v)\in E} f^*(u,v) \tag{44}$$

$$= \sum_{(v,w)\in E} \sum_{f\in\mathcal{T}} \alpha_f f(v,w) - \sum_{(u,v)\in E} \sum_{f\in\mathcal{T}} \alpha_f f(u,v) \tag{45}$$

$$= \sum_{f\in\mathcal{T}} \alpha_f \left(\sum_{(v,w)\in E} f(v,w) - \sum_{(u,v)\in E} f(u,v) \right) \tag{46}$$

$$= \sum_{f\in\mathcal{T}} \alpha_f \left(s(v) - d(v)\right) \tag{47}$$

$$= s(v) - d(v)\ . \tag{48}$$

Note that, to get Equation (47), we used the fact that each pure strategy f has to satisfy the flow conservation constraints, and to get Equation (48), we used the fact that $\boldsymbol{\alpha}$ is a convex linear combination.

2. We have to show the existence of a convex linear combination $\boldsymbol{\alpha}$. Our proof is constructive, and it is based on the following greedy algorithm:

   1: Let $i = 1$ and $f_1^* = f^*$.

   2: Let $F_i \subseteq E$ be the subset of edges to which f_i^* assigns a positive flow value.

   3: Find a feasible integer flow f_i of the original S-D network that uses only the edges belonging to F_i.

   4: Let $\alpha_{f_i} = \min_{e\in E} f_i^*(e)/f_i(e)$.

   5: For each $e \in E$, let $f_{i+1}^*(e) = f_i^*(e) - \alpha_{f_i} f_i(e)$.

   6: If the amount of flow transported by f_i^* is greater than zero, then $i = i+1$ and continue from Step 2. Otherwise, let the probability α_f of every other flow f be 0 and finish.[8]

   Before proving the correctness of the algorithm, we have to introduce one more notation: for $i \geq 1$, let α_i denote $\sum_{j=1}^{i} \alpha_{f_j}$, and let $\alpha_0 = 0$.

   First, we show that every f_{i+1}^* is a feasible flow given that the supply and demand values are scaled down by $(1-\alpha_i)$. More precisely, we show that the net outgoing flow assigned to a node v by f_{i+1}^* is $(1-\alpha_i)(s(v)-d(v))$. Obviously, this holds for f_1^*. Now, we assume that it holds for i, and show

[8] In practice, we do not have to actually assign values to unused flows, which would of course require an exponential number of steps.

that it then has to hold for $i+1$ as well. For each $v \in V$, we have

$$\sum_{(v,w)\in E} f^*_{i+1}(v,w) - \sum_{(u,v)\in E} f^*_{i+1}(u,v) \tag{49}$$

$$= \sum_{(v,w)\in E} (f^*_i(v,w) - \alpha_{f_i} f_i(v,w)) - \sum_{(u,v)\in E} (f^*_i(u,v) - \alpha_{f_i} f_i(u,v)) \tag{50}$$

$$= \sum_{(v,w)\in E} f^*_i(v,w) - \sum_{(u,v)\in E} f^*_i(u,v) - \alpha_{f_i} \left(\sum_{(v,w)\in E} f_i(v,w) - \sum_{(u,v)\in E} f_i(u,v) \right) \tag{51}$$

$$=(1-\alpha_{i-1})(s(v)-d(v)) - \alpha_{f_i}(s(v)-d(v)) \tag{52}$$

$$=(1-\alpha_i)(s(v)-d(v)) \ . \tag{53}$$

Since f^*_i is a feasible flow (with scaled down supply and demand values), there is a path to a source from every sink and a path to a sink from every source that only consist of edges on which there is a positive amount of flow. Consequently, Step 3 can be executed in each iteration.

Second, in every iteration, the flow decreases to zero on at least one edge (edges for which $f^*_i(e)/f_i(e)$ is minimal). Thus, the algorithm terminates after at most E iterations. Furthermore, as each step runs in polynomial time, the whole algorithm runs in polynomial time as well.

Third, after the algorithm has terminated, $\sum_{f\in\mathcal{T}} \alpha_f = 1$ as the net outgoing flow of any source v is $s(v) - d(v)$ before the first iteration, 0 after the last iteration, and it is decreased by $\alpha_{f_i}(s(v)-d(v))$ in each iteration i.

Finally, we have

$$\sum_{f\in\mathcal{T}} \alpha_f f(e) = \sum_i \alpha_{f_i} f_i(e) = f^*(e) \ . \tag{54}$$

□

E Link Propagation Delays of the Abilene and ISP3 Networks

Table 5: Link Propagation Delays (ms) of Each Network (20 Largest Values)

Abilene		ISP3	
0.032	0.016	0.075	0.014
0.032	0.016	0.033	0.014
0.026	0.011	0.033	0.012
0.026	0.011	0.033	0.012
0.025	0.011	0.033	0.012
0.025	0.011	0.025	0.012
0.02	0.009	0.025	0.01
0.02	0.009	0.023	0.01
0.02	0.008	0.022	0.008
0.02	0.008	0.022	0.008
0.017	0.004	0.022	0.008
0.017	0.004	0.021	0.008
0.016	0.004	0.021	0.007
0.016	0.004	0.019	0.007