Noname manuscript No.
(will be inserted by the editor)

Securing Interdependent Assets

Yevgeniy Vorobeychik · Joshua Letchford

the date of receipt and acceptance should be inserted later

Abstract Stackelberg security game models have become among the leading practical game theoretic approaches to security, having seen actual deployment in the LAX Airport, the United States Federal Air Marshals Service, and the United States Coast Guard, among others. However, most techniques for computing optimal security policies in Stackelberg games to date do not explicitly account for interdependencies among targets. We introduce a novel framework for computing optimal randomized security policies in networked (interdependent) domains. Our framework rests upon a Stackelberg security game model, within which we explicitly capture the indirect spread of damages due either to malicious attacks or unintended failures. We proceed to specify a particular simple, yet natural model of damage spread based on a graphical representation of asset interdependencies coupled with an independent failure cascade model. For the general model, we present an algorithm based on submodularity of the attacker's decision problem, in combination with local search, to approximate optimal security resource allocation across the assets, and show experimentally that our algorithm is far more scalable than an alternative exact approach, yields nearly optimal results, and offers substantial improvement over a well-known heuristic alternative. We then show that in a particular important special case we can compute optimal security policies exactly and efficiently. We proceed to apply our framework to study comparative network resilience, unifying previously disparate strands of research

Parts of this paper draw from the material previously presented at UAI 2012 (Letchford and Vorobeychik 2012). Specifically, the model of interdependencies presented in Letchford and Vorobeychik (2012) is a highly restricted special case of the model we present in this paper. Sections 4.3, 6, and 7 draw upon Letchford and Vorobeychik (2012), but much of the material in these sections is new.

Y. Vorobeychik
Vanderbilt University 2301 Vanderbilt Place, Nashville, TN 37235-1679
Tel.: 1-615-936-8153
Fax: 1-615-343-6702 E-mail: eug.vorobey@gmail.com

J. Letchford
Sandia National Laboratories
P.O. Box 969, Livermore, CA
E-mail: jletchf@sandia.gov

in the area, and to offer insights into other aspects of the interdependent security problem.

1 Introduction

The revolution in communication and computing technologies has spurred unprecedented growth in connectivity, be it technical, economic, or social. Everyone benefits from an increasingly connected world: we can collect more information and make better decisions about the electric power grid by communicating with an increasingly complex network of sensors and smart devices, can lead a large-scale project with globally dispersed participants from the comfort of an office, and maintain active membership in a community, real or virtual, despite being geographically removed from its epicenter. The benefits are so patent, indeed, that associated risks are often easy to overlook. The risk of connectivity is that local failures can have global consequences. This is now well recognized in cybersecurity, as viruses propagate from system to connected system, often affecting a large fraction of businesses. Melissa virus, for example, affected more than 300 organizations, causing over $80 million in damage (CERT 1999, Rosencrance 2002). As another example, the electric power grid, which is already a complex networks of generators, electric lines, along with businesses and households, is becoming much more so with the increasingly sophisticated sensors and "smart" meters, and with increasing complexity of the grid, security failures can have increasingly severe consequences (Stamp et al 2009, Energy Sector Control Systems Working Group 2011).

Despite the importance of accounting for interdependent risks in security decisions, there are few systematic approaches for empowering a decision maker to do so. The majority of the approaches to guide security investment decisions aspire to do so without explicitly accounting for interdependencies. For example, the standard approach to security risk management in the industry is to consider consequences in terms of asset value, consequence of a threat on that value, and frequency of threat, but either treats assets as independent, or abstracts away the complex interdependencies in a single cost/value measure (Krutz and Vines 2001). Research in IT security management has largely been in line with this framework (Yue and Bagchi 2003, Ulvila and Gaffney 2004, Cavusoglu and Raghunathan 2004, Cavusoglo et al 2004, Ogut et al 2005, Cavusoglu et al 2005, Ogut et al 2008, Cavusoglu et al 2008, 2009, August and Tunca 2011). The strands that explicitly model interdependent risk focus on spillover effects among many organizations or entities, rather than policy to secure interdependent assets (Kunreuther and Heal 2003, Cremonini and Nizovtsev 2006).

Our point of departure is a class of optimization-based game theoretic approaches in security settings referred to as *Stackelberg security games* (Paruchuri et al 2008). These are two-player games in which a *defender* aims to protect a set of targets using a fixed set of limited defense resources, while an *attacker* aims to assail a target that maximizes his expected utility. A central assumption in the literature on Stackelberg security games is that the defender can commit to a probabilistic defense (equivalently, the attacker observes the probabilities with which each target is covered by the defender, but not the actual defense realization). Much of the work on Stackelberg security games focuses on building fast, scalable

algorithms, often in restricted settings (Kiekintveld et al 2009, Jain et al 2010a, Shieh et al 2012). One important such restriction is to assume that targets exhibit *independence*: that is, the defender's utility only depends on which target is attacked and the security configuration at that target. Short of that restriction, one must, in principle, consider all possible combinations of security decisions jointly for all targets, making scalable computation elusive. Many important settings, however, exhibit interdependencies among potential targets of attack. These may be explicit, as in IT and supply chain network security, or implicit, as in defending critical infrastructure (where, for example, successful delivery of transportation services depends on a highly functional energy sector, and vice versa), or in securing complex software systems (with failures at some modules having potential to adversely affect other modules). While in such settings the assumption of independence seems superficially violated, we demonstrate below that under realistic assumptions about the nature of interdependencies, we can nevertheless leverage the highly scalable optimization techniques which assume independence.

In all, we offer the following contributions. First, we introduce a general framework to modeling security decisions for interdependent assets in the presence of both adversarial and non-adversarial threats. Second, we instantiate our general model of interdependencies using a graph in combination with an independent failure cascade model. Third, we present a general heuristic algorithm for computing approximately optimal security policies on networks that leverages submodularity of the attacker's problem in combination with a simple, yet highly effective, local search heuristic. Fourth, we present an important special case of our model which admits a highly scalable algorithm for computing optimal security policies *exactly*. Fifth, we apply our framework to study several applications of interdependent security, using both real networks, as well as stochastic generative network models. One of our most significant experimental contributions is an extensive study of comparative network resilience. This is a field which has had considerable significance in the broad network science literature and, indeed, is at the focus of two disparate strands of literature: the first comparing susceptibility of networks to attacks or random failures *when no defense is present*, and second, studying inoculation strategies on networks to protect from infectious disease spread, allowing for *no targeted attacks*. Our framework is the first that allows us to capture both endogenous defense measures and targeted attacks on networks, allowing us to unify these two strands of research. Our results, thus, provide much insight into both of these areas, offering additional nuance and, at times, contradicting the commonly held intuitions.

1.1 Literature Review

Our work is situated within the rapidly expanding body of literature on security investment and policy. Topically, this literature can be grouped into several streams. The first studies security policies from the perspective of liability considerations (Cavusoglu et al 2008, August and Tunca 2011), considering, for example, alternative ways to allocate burden or damage of security decisions (such as liability for zero-day exploits). The second is focused on the technical capabilities side, aiming to develop better intrusion detection systems (IDS), or IDS that are especially attuned to costs of decisions about classifying threats (Provost and Fawcett

1997, Domingos 1999, Lee et al 2002, Anderson 2008). The third stream, which is most closely connected to our aims, involves approaches to improve security investment decision support. Within this stream there are three general approaches: risk-based, decision theoretic, and game theoretic.

The risk-based approach is perhaps the oldest and seems still to be the principal approach in practice. At the crux of this approach is evaluation of specific security risks facing an organization, perhaps through an associated assessment of vulnerabilities, threats, and consequences (Krutz and Vines 2001, MITRE 2012, Duggan et al 2007, Mounzer et al 2010). While much attention is paid in this literature on risk assessment and understanding threats (e.g., attackers), it offers relatively little quantitative guidance about *mitigation*, aside from the most basic cost-benefit comparison between deploying a particular security measure, and the expected risks and consequences it is meant to ameliorate.

The academic research community, in contrast, has aimed to shift focus on providing specific guidance about security investments, though in many cases this guidance is in very specific security contexts, such as whether or not to deploy a firewall or an IDS, and how to configure it if deployed (Yue and Bagchi 2003, Ulvila and Gaffney 2004, Cavusoglu and Raghunathan 2004, Cavusoglo et al 2004, Cavusoglu et al 2005, 2009). The corresponding approaches are either decision theoretic, modeling threats as unaffected by mitigation policies (Yue and Bagchi 2003, Ulvila and Gaffney 2004, Cavusoglu and Raghunathan 2004), or game theoretic, accounting for the impact of security policies on attackers' incentives (Cavusoglu and Raghunathan 2004, Cavusoglo et al 2004, Cavusoglu et al 2005, 2009).

Game theoretic treatment of security is intimately connected to two simple classical models: inspection games (Avenhaus et al 2002) and colonel Blotto games (Roberson 2006). The most basic variant of an inspection game involves an inspectee (e.g., a tax evader) who can choose to perform an illegal or a legal action, and an inspector, who receives a noisy signal upon which he can inspect (at some cost), or not. One qualitative difference between this generic inspector game and some of the models we described above, as well as our own approach, is that in our case the defender (inspector) acts first, and the attacker (inspectee) acts *after observing the defender's decision* (which may be randomized, in which case the attacker observes the probability distribution). Moreover, here, as in the above references, the defender's and attacker's action spaces are quite simple, and no interdependencies are relevant. Colonel Blotto game, too, is a simultaneous move game, but here two commanders are endowed with armies, and get to place a fraction of their force on each of n battlefields. Whichever side has the most forces on a battlefield wins that battle, and the winner of the game is the commander with the most battle victories. In this game, the decision space of each player is actually rather complex, though complex in a different way from our setting. However, the game is zero-sum (ours is not), and here again no interdependencies are typically modeled.

Insofar as interdependencies in security decisions have been modeled in related literature, this has been done in the context of interdependencies among multiple entities aiming to jointly defend their systems, with the focus on outcomes of strategic interactions, rather than offering a security policy for the *entire interdependent system* (Kunreuther and Heal 2003, Ogut et al 2005, Cremonini and Nizovtsev 2006, Grossklags et al 2008). For example, Kunreuther and Heal (2003) study the problem of interdependencies among players, each deciding whether or

not to invest in better security. There is no attacker in their model, so in that sense its scope is quite different from ours. Moreover, an individual player's decision is binary. What they aim to model are spillovers due to a decision not to secure one's own assets onto others, and they demonstrate that in many cases equilibrium exhibits insufficient security overall.

While most of the work described above considers either exogenously specified risks (e.g., natural disasters or human error), or deliberate attacks that adopt to the security policy, Zhuang and Bier (2007) were the first to consider both in a single comprehensive model as we do, albeit without explicitly modeling interdependent risks.

All the work described so far on game theoretic and decision theoretic approaches to security attempts to characterize decisions by the defender, attacker, or both using closed-form mathematical expressions. In parallel, there has been considerable literature that aspires to *compute* security decisions. One such stream involves numerous variants of *network interdiction* problems. At the high level, all such approaches start with a network flow or shortest path problem, with the goal of choosing an action (such as blocking a subset of nodes or arcs on the network) that most effectively reduces the flow or increases shortest paths (Wood 1993, Cormican et al 1998, Woodruff 2003, Brown et al 2006, 2009, Nehme 2009). Like ours, these efforts all use mathematical programming formulations to compute an optimal interdiction strategy. Unlike our work, however, these efforts are fundamentally restricted to zero-sum games, account for interdependencies using models based on network flow, and in most cases do not include defense against interdiction, which is our focus here. Brown et al (2006) do present a tri-level formulation that attempts to allow one to take countermeasures against being interdicted by an attacker, but this model is extremely difficult to scale, making its practical utility quite limited.

Our point of departure is a class of optimization approaches for security decisions referred to commonly as *Stackelberg security games*. The paper that provided the computational foundations for what has become an active subfield of computational game theory was the work by Conitzer and Sandholm on computing optimal Stackelberg commitment strategies in general finite games (Conitzer and Sandholm 2006). In this paper, Conitzer and Sandholm presented the first algorithm for computing optimal randomized commitment strategies in Stackelberg games. Paruchuri et al (2008) presented the first mixed-integer linear programming formulation for computing a Stackelberg equilibrium in Bayesian Stackelberg games. Kiekintveld et al (2009) introduce an important restricted class of Stackelberg games specifically targeted at security settings; they refer to these as *Stackelberg security games*, and demonstrate that extremely scalable algorithms can be devised for this class of games. Since then, a number of follow-up papers have emerged, studying, for the most part, computational aspects of the problem and aiming to scale the algorithms to larger and larger instances (Letchford and Conitzer 2010, Tsai et al 2010, von Stengel and Zamir 2010, Jain et al 2010a, Korzhyk et al 2010, Jain et al 2011, Conitzer and Korzhyk 2011, Tsai et al 2012), as well as illustrating their actual deployment in the field, such as the LAX airport (Pita et al 2009), Federal Air Marshall Service (Jain et al 2010b), and the US Coast Guard (Shieh et al 2012). Of these approaches, Tsai et al (2012) presents the most similar model to ours. The principal difference is in the game structure and motivation: Tsai et al. model *both* the defender and attacker as agents who aim to influence contagion

of ideas in a simultaneous move game; thus, the two players actually have symmetric roles. In our model, the attacker's goal is to start a failure cascade, but the defender aims to *minimize damages from cascading failures*, not start a cascade of his own.

2 Stackelberg Security Games

At the core of our model lies a *Stackelberg security game*, which consists of two players, the leader (defender) and the follower (attacker), and a set of possible targets. The leader can decide upon a randomized policy of defending the targets, possibly with limited defense resources. The follower (attacker) is assumed to observe the randomized policy of the leader, but not the realized defense actions. Upon observing the leader's strategy, the follower chooses a subset of targets to attack so as to maximize its expected utility. The typical solution concept for these games is a Strong Stackelberg Equilibrium (SSE), in which the leader plays an optimal policy that accounts for an follower's optimal response to it and, moreover, presumes that the follower breaks ties in the leader's favor.[1]

In past work, Stackelberg security game formulations focused on defense policies that were costless, but resource bounded, and security decisions amounted to covering (defending) a set of targets, or not. In numerous settings such models are quite limiting. For example, in cybersecurity, protecting computing nodes could involve configuring anti-virus and/or firewall settings, with stronger settings carrying a benefit of better protection, but at a cost of added inconvenience, lost productivity, as well as possible licensing costs. Indeed, costs on resources may usefully replace resource constraints, since such constraints are often not hard, but rather channel an implicit cost of adding further resources. Thus, our model allows the defender to choose among many *security configurations* for each valued asset, and, additionally, security resources are only available at some cost. Furthermore, while security games as described above naturally entail an attacker, in practice most failures are not at all a deliberate act of sabotage, but are due entirely to inadvertent errors. Thus, we also depart from previous literature on Stackelberg security games by explicitly modeling both attacks and random failures.

To formalize, suppose that the defender can choose from a finite set O of security configurations for each target $t \in T$, with $|T| = n$. A configuration $o \in O$ for target $t \in T$ incurs a cost $c_{o,t}$ to the defender. Let $s = \{o_1, \ldots, o_n\}$ be the (pure strategy) security configuration vector, with $o_t \in O$ denoting the security configuration chosen for target t; we refer to s as the *defense policy*. We denote by q_s the probability that the defender chooses a security configuration vector s. The attacker observes the randomized defense policy vector q, and chooses a subset of at most L targets to attack; let us denote this subset by $A = \{t_1, ..., t_L\}$. We denote the defender's utility function by $U(s, A)$ and the attacker's by $V(s, A)$ where s is the defense policy and A the attacker's response. To capture the distinction between active attacks and "nature", let r be the prior probability of the defender that a failure will happen due to a deliberate attack. If no attack is involved,

[1] The idea that the follower breaks ties in the leader's favor may seem strange in the context of security games. However, note that the leader can make the follower strictly prefer the corresponding action by a slight change in his randomized policy.

any target can fail; the defender's belief that a set of targets B randomly fails (conditional on the event that no attack is involved) is g_B, with $\sum_B g_B = 1$.

3 Modeling Asset Interdependencies

3.1 A General Model

In this section we offer a general model of interdependencies among assets. We then present an important special case that admits a far more scalable approach for computing optimal security policies. Throughout this section we focus on the defender's utilities; attacker is treated identically.

Let w_t be an *intrinsic worth* of a target to the defender, that is, how much loss the defender would suffer if this target were to be compromised with no other target affected (i.e., not accounting for indirect effects). In doing so, we assume that these worths are independent for different targets. Moreover, suppose that when a target t is damaged or compromised (due to a successful attack either on t directly, or on another target which indirectly impacts t), only a fraction α_t of its worth remains. We allow α_t to be a random variable if the impact of an attack is non-deterministic. Let $z(\tilde{A}, s_A; A)$ be the probability that a subset $\tilde{A}$ of targets fails (or are compromised) when a subset of targets A are attacked and the defense configuration for nodes in A is s_A (that is, s_A is the portion of the defense vector s restricted to nodes in A). For example, if a failure of every node t due to an attack is independent of security configuration of other nodes and $\tilde{A} \subseteq A$, $z(\tilde{A}, s_A; A) = \prod_{t \in \tilde{A}} z(o, t)$, where $z(o, t)$ is the probability that node t fails if attacked when its security configuration is o. The defender utility when security configuration is s and the attacker attacks a subset A of targets is

$$U(s, A) = \sum_{\tilde{A} \subseteq A} z(\tilde{A}, s_A; A) E\left[\sum_{t'} \alpha_{t'} w_{t'} |\ s, \tilde{A}\right] = \sum_{\tilde{A} \subseteq A} z(\tilde{A}, s_A; A) \sum_{t'} w_{t'} E\left[\alpha_{t'} |\ s, \tilde{A}\right]. \tag{1}$$

We can think of the term $E\left[\alpha_{t'} |\ s, \tilde{A}\right]$ as the expected damage to target t' when the subset of targets $\tilde{A}$ is successfully compromised by the attacker and the security configuration vector is s.

3.2 Cascading Failures Model

In general, one may use an arbitrary model to compute or estimate the consequences of node failures due to interdependence, $E\left[\alpha_{t'} |\ s, A\right]$. Here, we offer a specific model of interdependence between targets that is simple, natural, and applies across a wide variety of settings.

Let us fix the security policy vector s and the set A of targets that are initially compromised. Suppose that dependencies between targets are represented by a graph (T, E), with T the set of targets (nodes) as above, and E the set of edges (t, t'), where an edge from t to t' (or an undirected edge between them) means that target t' depends on target t and, thus, a successful attack on t may have an impact on t'. Each target has associated with it a worth, w_t, as above, although in the

current context this worth is incurred only if t is affected (e.g., compromised, broken). We model the interdependencies between the nodes as independent cascade contagion, which has previously been used primarily to model diffusion of product adoption and infectious diseases (Kempe et al 2003, Dodds and Watts 2005). The contagion proceeds starting at the attacked nodes $t \in A$, affecting each of their network neighbors t' with probability $p_{t,t'}(s)$, then spreads from each affected t', and so on, recursively. Contagion can only spread once along any network edge, and if a node is affected, it remains affected through the diffusion process (note also that in this model, the node is either affected, or not; we let $\alpha_t = 0$ when a node is affected by an attack and $\alpha_t = 1$ when it is not[2]). An equivalent way to model this process is to start with the network (T, E) and remove each edge (t, t') with probability $(1 - p_{t,t'}(s))$. The entire connected component of each attacked node is then deemed affected. As an important special case, we can use $p_{t,t'}(o_{t'})$ to model the impact of inoculation on the probability of becoming infected, for example, setting it to 0 if $o_{t'}$ is the decision to administer inoculation on node t' and to 1 if $o_{t'}$ is the decision not to inoculate t'.

3.3 Computing Expected Utilities

In principle, our setup allows us to fully decouple computing or estimating expected utilities $U(s, A)$ and $V(s, A)$ of the defender and the attacker respectively, and subsequently computing an optimal defense policy. In general, we can estimate player utilities by simulating cascades starting at every subset of nodes $\tilde{A}$ of size at most L and for every (deterministic) security configuration vector s, with expected utility of defender/attacker estimated as a sample average over K simulated cascades to obtain estimates of $E\left[\alpha_{t'} \middle| s, \tilde{A}\right]$, and applying Equation 1. Clearly, however, even estimating expected utilities for the entire game is an entirely intractable process in our general setup. Consequently, in the fully general case, we would wish to compute or approximate a Stackelberg equilibrium without having to know the full payoff functions of both players. Below, we demonstrate how this can be done using a combination of heuristic and submodular optimization methods. For the moment, however, we introduce a special case which allows us to compute an optimal security policy exactly and efficiently.

3.4 Special Case: Single-Node Attacks and Security-Independent Cascades

The most basic problem with the general setup that we described above is that in order to estimate the defender and attacker utility functions, and ultimately compute optimal security strategies, one needs to perform a set of simulations *for each defense policy vector s and attack strategy A*. Clearly, this becomes intractable even for a modest number of targets. In this section, we introduce several restrictions on the general model that allow both a much more compact representation of the players' payoff functions, and, ultimately, offers an opportunity for highly scalable Stackelberg equilibrium computation.

[2] Note that it is direct to replace these choices by arbitrary different constants

The first restriction is that the attacker can only attack a single target. Note that under this restriction, Equation 1 simplifies to

$$U(s,t) = z(o,t) \sum_{t'} w_{t'} E\left[\alpha_{t'} |\ s,t\right]. \tag{2}$$

Indeed, this restriction has been operational in most related work on computing strong Stackelberg equilibria in the context of security (Kiekintveld et al 2009, Jain et al 2010a, Shieh et al 2012). The second restriction is captured by the following condition on the impact of interdependencies:

Condition 1 *For all t and t', $E\left[\alpha_{t'} |\ s,t\right] = E\left[\alpha_{t'} |\ o_t,t\right]$.*

In words, the probability that a target t' is affected when an initially attacked target t fails only depends on the security configuration at the attacked target t. Below, we use o instead of o_t where t is clear from context.

There are several natural ways to think about Condition 1. The simplest is consider the consequences of attacks as affecting network flows. In this case, removing a node t and its incident edges from a network means that any flow between a pair of other nodes s, r must take a different route and, indeed, it may even be that s and r are now disconnected. Significantly, the utility lost in this case only depends on the security configuration at t. An alternative way to interpret Condition 1 is that security against external threats is not very efficacious once an attacker has found a way into the system. For example, in cybersecurity defense is often focused on external threats, with little attention paid to threats coming from computers internal to the network. Thus, once a computer on a network is compromised, the attacker may find it much easier to compromise others on the same network. This second interpretation gives rise to a very natural restriction on the cascading failure model that satisfies Condition 1: $p_{t,t'}$ do not depend on security configurations at nodes. This restriction is very common, as argued above. There is, however, an important setting in which it is clearly unrealistic: bioterrorism, where inoculation decisions reduce the likelihood of an individual being infected either by the attacker, or by another infected individual.

Under Condition 1, the defender's utility when t is attacked under security configuration o becomes:

$$U(o,t) = z(o,t) \left[w_t E[\alpha_t | o_t, t] + \sum_{t' \neq t} w_{t'} E[\alpha_{t'} | o_t, t] \right].$$

Thus, in this special case, we can represent the game much more compactly, using $U(o,t)$ and $V(o,t)$ to denote the defender's and attacker's utility, respectively, when target t is attacked and the security configuration at that target is o. In a slight abuse of notation, we denote by $q_{o,t}$ the probability that the defender chooses o at target t. Note that given q_s, we can compute $q_{o,t}$ as $q_{o,t} = \sum_s q_s 1(s_t = o)$, where $1(\cdot)$ is an indicator function which is 1 when its argument is true and 0 otherwise. Capturing the natural disasters in this special setting requires us (for algorithmic reasons) to restrict nature to affect a single target at a time. Thus, we will abuse notation again, denoting by g_t the probability that target t randomly fails (conditional on the event that no attack is involved), with $\sum_t g_t = 1$.

3.5 Incorporating Uncertainty about the Network

Applying our framework in real-world networked security settings requires an accurate understanding of the interdependencies. Thus far, we assumed that the actual network over which cascading failures would spread is perfectly known. A natural question is: what if our network model is inaccurate?

Formally, we model the uncertainty about the network as a parameter ϵ which represents the probability of incorrectly estimating the relationship between a pair of targets. Thus, if there is an edge between t and t', we now let this edge be present with probability $1-\epsilon$. On the other hand, if t and t' are not connected in the graph given to us, we propose that they are, in fact, connected with probability ϵ. Thus, when the graph is large, even a small amount noise will cause us to err about a substantial number of edges.[3]

Note that there is a natural way to incorporate this model of uncertainty into our framework. Let us interpret $p_{t,t'}(s)$ as the probability of a cascade from t to t' *conditional on an edge from* t *to* t'. Then, if t and t' are connected, we modify cascade probabilities to be $\hat{p}_{t,t'}(s) = p_{t,t'}(s)(1-\epsilon)$, whereas if they are not connected, the cascade probability is $\hat{p}_{t,t'}(s) = p_{t,t'}(s)\epsilon$.

4 Computing Optimal Randomized Security Configurations

4.1 The General Case: Exact Solution

Previous formulations of Stackelberg games for security involved a fixed collection of defender resources, and in most cases a binary decision to be made for each target: to cover it, or not. To adapt these to our domains of interest, we first modify the well-known multiple linear program (henceforth, multiple-LP) formulation to incorporate an arbitrary set of security configurations, together with their corresponding costs of deployment. In the multiple-LP formulation, each linear program solves for an optimal randomized defense strategy *given that the attacker attacks a fixed subset of targets* $\hat{A}$, with the constraint that $\hat{A}$ is an optimal choice for the attacker. The defender then chooses the best solution from all feasible LPs as his optimal randomized defense configuration. The LP formulation for a representative subset of targets $\hat{A}$ is shown in Equations 3a-3d.

$$\max \quad r\left(\sum_s U(s,\hat{A})q_s^{\hat{A}}\right) + (1-r)\left(\sum_{B,s} g_B U(s,B) q_s^{\hat{A}}\right) - \sum_t \sum_o c_{o,t} q_{o,t}^{\hat{A}}. \tag{3a}$$

s.t.

$$\forall_s \quad q_s^{\hat{A}} \in [0,1] \tag{3b}$$

$$\sum_s q_s^{\hat{A}} = 1 \tag{3c}$$

[3] We assume here that both the defender and attacker share the same uncertainty about the network. An alternative model could consider an attacker that has more (or exact) information about the network. The resulting defender problem would become a Bayesian Stackelberg game.

$$\forall_A \quad \sum_s V(s,A)q_s^{\hat{A}} \leq \sum_s V(s,\hat{A})q_s^{\hat{A}} \tag{3d}$$

The intuition behind the multiple-LP formulation is that in an optimal defense configuration, the attacker must (weakly) prefer to attack *some* subset of targets, and, consequently, one of these LPs must correspond to an optimal defense policy.

4.2 Approximating Security Policy in the General Case

There are two significant problems with the LP formulation for computing optimal defender policies we described above. First, the LP itself becomes intractably large when we have a sufficient number of network nodes and defense configuration options. Perhaps a far more significant problem, however, is that the LP requires us to first compute or estimate the expected utilities for each joint strategy of the defender and attacker based on our model of interdependencies. It is this bottleneck, as much as any other, that renders the exact approach intractable in practice.

In this section, we offer an alternative that takes advantage of the special structure in the independent failure cascades model. This alternative approach allows us to avoid estimating the entire payoff matrix, interleaving optimization and estimation steps instead in a manner analogous to simulation-based game theoretic analysis (Vorobeychik and Wellman 2008). To simplify the problem, we restrict attention here to deterministic defense policies; generalization is immediate if we discretize randomized policies.

We begin by focusing on the attacker's best response problem, an algorithmic challenge in its own right, and subsequently proceed to propose a local search heuristic to obtain a defender's policy in which the attacker's optimization problem is a subroutine. We assume henceforth that the interdependencies among the targets are modeled using the dependency graph and independent failure cascades.

4.2.1 Approximating an Optimal Attack

The attacker's problem is to choose a subset of L targets to attack so as to maximize his expected utility $V(s,A)$. This problem is a generalization of the well-known problem of influence maximization (Kempe et al 2003), in which a decision maker aims to maximize the expected number of individuals (rather than utility) affected by a cascade started from the chosen nodes. Kempe et al. showed that the problem of choosing an optimal subset of L nodes to seed when subsequent influence spreads according to an independent cascades model is NP-Hard. In our setting, the attacker's problem is a slight generalization of this model, and NP-Hardness of the attacker's problem is therefore immediate (setting $w_t = 1$ for all nodes recovers the original influence maximization problem).

Theorem 1 *Computing an optimal attack strategy is NP-Hard.*

An important algorithmic insight by Kempe et al. is that while solving the influence maximization problem optimally is hard, the objective function is *submodular*. Consequently, a simple greedy heuristic yields a constant factor approximation and, in practice, gives nearly optimal solutions. While our setting is slightly more general, we can readily extend this submodularity result.

Theorem 2 *The attacker's objective function is submodular.*

Proof Note that the cascade process can be equivalently formulated by first flipping the biased coins for each edge, keeping the edge between t and t' with probability $p_{t,t'}(o_t, o_{t'})$ and deleting it otherwise. The total utility to the attacker given such a realization is the sum of the worths of all targets affected by the attacker's decision A. Let T_t be the set of targets with a finite path from a particular target t, and let $T_R = \cup_{r \in R} T_r$ be the set of targets reachable from any target in a set R. Finally, for any set of targets $R \subseteq T$, define $U(R) = \sum_{r \in R} w_r$, that is, the total worth of all targets in R.

Suppose $R \subseteq S \subseteq T$ be targets of initial attack and consider attacking an additional target t'. The attacker's utility when the set R of targets is attacked is $U(T_R)$, while the utility from attacking targets in $R \cup t'$ is $U(T_{R \cup t'})$. Then,

$$U(T_{R \cup t'}) - U(T_R) = \sum_{r \in T_{R \cup t'}} w_r - \sum_{r \in T_R} w_r = \sum_{r \in T_{R \cup t'} - T_R} w_r.$$

Now, observe that if $R \subseteq S$, $T_{S \cup t'} - T_S \subseteq T_{R \cup t'} - T_R$, which implies that

$$\sum_{r \in T_{R \cup t'} - T_R} w_r \geq \sum_{r \in T_{S \cup t'} - T_S} w_r = U(T_{S \cup t'}) - U(T_S),$$

which in turn implies that for every realization of the random cascade graph, the attacker utility is submodular. Since submodularity is preserved under linear transformations, the attacker expected utility is also submodular. □

The implication is that for a fixed defense policy s we can approximate the optimal attack to a factor of $1 - 1/e$ with an iterative greedy algorithm which chooses, in each iteration, the target to attack that attains the highest increase in expected utility with respect to previously chosen targets (Nemhauser et al 1978).

4.2.2 Computing a Defense Policy

Thus far we have shown that we can compute a near-optimal strategy for the attacker reasonably fast. We now come to the main problem: computing a defense policy. First, we observe that while the attacker's problem is submodular, this is not the case for the defender: defense decisions have complementarities. These arise because targets are interdependent and, therefore, defending one target may have little effect until other targets connected to it are also defended. The presence of such complementarities would in principle make the combinatorial optimization problem faced by the defender extremely difficult. However, we offer a simple local search heuristic and show empirically that it is highly effective, particularly when combined with random restarts.

To begin, let us make several basic structural observations. First, suppose that the cascade probabilities $p_{t,t'}(s)$ only depend on the security configuration at t and t', which we refer to as o and o'. To make this restriction explicit, we can denote the corresponding cascade probabilities by $p_{t,t'}(o, o')$. In this very natural special case, if a particular security configuration o is less effective than another, o', and is at the same time more expensive than o', we can prune it from consideration, since it is *dominated* by o', a notion which we now formally define.

Definition 41 *A security configuration r is* stronger *than o if $z(o,t) \geq z(r,t)$ for all $t \in T$, $p_{t,t'}(r,r) \leq p_{t,t'}(r,o)$, $p_{t,t'}(r,r) \leq p_{t,t'}(o,r)$, and $p_{t,t'}(r,r) \leq p_{t,t'}(o,o)$ for all $t,t' \in T$, with at least one inequality being strict.*

In words, we say that a security configuration r is stronger than o if both the probability of failure due to a direct attack is smaller under r, and using o' instead of o reduces the indirect exposure to cascades.

Definition 42 *A security configuration o is* dominated *if $\exists r \in O$ with $c_{r,t} \leq c_{o,t}\ \forall\ t \in T$ that is stronger than o (i.e., r is both stronger and cheaper).*

Second, suppose that cascade probabilities do not depend on security configurations (a special case of our model). In this case, increasing the amount of defense (formally, choosing a stronger security configuration that is more expensive) at a particular target has no value to the defender unless either this target is attacked, or the defender simultaneously increases defense at another target that is. The reason is that since the attacker's decision is not affected, the only consequence is the increased cost to the defender. While this observation is no longer true when cascade probabilities depend on defense, we nevertheless base our local search on it, and view it as a heuristic in the general case.

We propose a simple local search algorithm (Algorithm 1) that iteratively chooses a single target at a time, distinguishing between those that are currently attacked and those that are not based on the second observation above, and chooses a locally optimal security configuration for that target.

```
Data: Starting defense policy s_0, number of iterations I
Result: Final defense policy s
s ← s_0;
prune all dominated o ∈ O;
for i = 1 to I do
    A ← computeAttack(s) // targets attacked under s;
    for t ∈ A do
        // fix all other decisions
        // compute the local optimum at target t
        o_t ← computeBest(t);
        s ← {s_1, ..., o_t, ..., s_n};
    end
    for t ∉ A do
        // compute local optimum, considering only decreasing security
        o_t ← computeBestDecrease(t);
        s ← {s_1, ..., o_t, ..., s_n};
    end
end
```

Algorithm 1: Local search for a defense policy.

Algorithm 1 requires as input an initial defense policy from which to start local search. Two natural candidates are the weakest and strongest policies, i.e., a policy in which every target is using a weakest (resp. strongest) security configuration, if these exist. As an example, one usually has an option of "no security", which is the weakest option, and "high security", which would be the strongest. A third natural candidate is a well-known heuristic, choosing individuals to defend in decreasing order of degree; this is commonly referred to as *targeted vaccination* (Pastor-Satorras

and Vespignani 2002); since this heuristic plays an important role in the literature on vaccination on networks, below we show experimentally that in isolation it is significantly worse than our local search method. Finally, we can start from a random defense policy. Ultimately, since this is only a local search, and our problem exhibits complementarities, we would not expect it to yield optimal solutions in general. Therefore, our full approach runs the local search from the weakest and strongest defense policy, if these exist, then from a configuration based on targeted vaccination, and finally runs it from P random starting policies. Below, we show empirically that the local search often yields nearly optimal solutions even without random restarts.

Note that local search implicitly invokes a subroutine for computing an optimal attacker strategy; this is actually explicit in the *computeAttack*(s) function call and implicit in both functions computing locally best security configuration at a given target. If we could compute this strategy optimally, we could guarantee that our overall approach converges to an optimal defense with probability 1 if we let the number of random restarts grow without bound. While this is easy to guarantee when the attacker can only attack a single target, it is no longer reasonable when the attacks can happen on multiple targets simultaneously. Nevertheless, if the game is nearly constant-sum (in the sense we formalize presently), computing an approximately optimal attacker strategy suffices to guarantee convergence to an approximately optimal defense. For convenience, suppose that both the attacker and defender always obtain non-negative payoffs.

Definition 43 *A security game is ϵ-constant-sum if there exists $c \geq 0$ such that $c-\epsilon \leq U_{s,a} + V_{s,a} \leq c + \epsilon$ for all s, a.*

Theorem 3 *Suppose that the game is ϵ-constant-sum. Additionally, suppose that $\hat{A}(s)$ is an α-approximation of an optimal attacker strategy $A^*(s)$ for a given defense policy s. Let $\hat{s}$ be an optimal defender policy if the attacker response is measured according to $\hat{A}$, and let s^* be the true optimal policy. Then $U(\hat{s}, A^*(\hat{s})) \geq U(s^*, A^*(\hat{s})) - (\alpha - 1)V(s^*, A^*(\hat{s})) - 2\epsilon(\alpha + 1)$.*

Proof Choose an arbitrary defense policy s. Since $\hat{A}(s)$ is an α-approximation (for $\alpha \geq 1$),

$$\alpha V(s, \hat{A}(s)) \geq V(s, A^*(s)).$$

Using $c - \epsilon \leq U(s, A) + V(s, A) \leq c + \epsilon$ for all s, A, this implies that

$$\alpha(c - U(s, \hat{A}(s)) + \epsilon) \geq c - U(s, A^*(s)) - \epsilon,$$

or, equivalently,

$$U(s, A^*(s)) \geq \alpha U(s, \hat{A}(s)) - c(\alpha - 1) - \epsilon(\alpha + 1).$$

Since this is true for every s,

$$\begin{aligned} U(\hat{s}, A^*(\hat{s})) &\geq \alpha U(\hat{s}, \hat{A}(\hat{s})) - c(\alpha - 1) - \epsilon(\alpha + 1) && (4)\\ &\geq \alpha U(s^*, \hat{A}(s^*)) - c(\alpha - 1) - \epsilon(\alpha + 1) && (5)\\ &\geq \alpha U(s^*, A^*(s^*)) - c(\alpha - 1) - \epsilon(\alpha + 1) - 2\epsilon && (6)\\ &= \alpha U(s^*, A^*(s^*)) - c(\alpha - 1) - \epsilon(\alpha + 3), && (7) \end{aligned}$$

where inequality 5 follows because of optimality of $\hat{s}$ for the defender under $\hat{a}$ and inequality 6 is due to the fact that $\hat{a}$ is suboptimal for the attacker. Rearranging and letting $c \leq U(s^*, A^*(s^*)) + V(s^*, A^*(s^*)) + \epsilon$ we get the desired result. □

4.3 Special Case: Single-Node Attacks and Security-Independent Cascades

The multiple-LP formulation 3 for the general case requires us to have a variable for each possible security configuration vector *and* requires us to solve an LP for each subset of L targets. Since the number of possible configurations, as well as the number of possible subsets of targets, is exponential in the number of targets, exact security policy computation cannot scale beyond very small instances. However, if we assume that the attacker can attack at most a single target, restrict random failures to a single target at a time, and assume that the defender's utility only depends on the target being attacked or failing (Condition 1), we can obtain a far more compact and scalable formulation. Under these assumptions, we can treat the defense configuration for each target $q_{o,t}$ in isolation, as we no longer need to randomize over joint defense schedules. Moreover, we need only solve n LPs, one for each target $\hat{t}$ of possible attack. The LP formulation for a representative target $\hat{t}$ is shown in Equations 8a-8d.

$$\max \quad r\left(\sum_o U(o,\hat{t})q^{\hat{t}}_{o,\hat{t}}\right) + (1-r)\left(\sum_{t,o} g_t U(o,t)q^{\hat{t}}_{o,t}\right) - \sum_t \sum_o c_{o,t}q^{\hat{t}}_{o,t}. \tag{8a}$$

s.t.

$$\forall_{o,t}\ q^{\hat{t}}_{o,t} \in [0,1] \tag{8b}$$

$$\forall_t \sum_o q^{\hat{t}}_{o,t} = 1 \tag{8c}$$

$$\forall_t \sum_o V(o,t)q^{\hat{t}}_{o,t} \leq \sum_o V(o,\hat{t})q^{\hat{t}}_{o,\hat{t}} \tag{8d}$$

Notice that we can easily incorporate additional linear constraints. For example, it is often useful to add a budget constraint of the form:

$$\forall_{\hat{t},t} \quad \sum_o c_{o,t}q^{\hat{t}}_{o,t} \leq C.$$

4.3.1 The Impact of Sampling Noise

While we can compute the expected utilities exactly in certain important special cases (see Letchford and Vorobeychik (2012)), in general we must sample cascades to estimate expected utilities of players, and solve the optimization problem (8a-8d) using estimated utilities. This raises a natural question: does this approach yield a solution close to optimal if we take sufficient samples of cascades, and thereby obtain an arbitrarily good estimate of utilities for all outcomes? The answer, it turns out, is non-trivial, because sampling noise does not merely affect the objective functions of the LPs we solve, but also the constraints.

To appreciate what can go wrong, consider an example with two targets, 1 and 2, and suppose that there are only two security configurations: a target can either be covered or not. Let U^u_t and U^c_t be the defender's actual utilities if target t is uncovered and covered, respectively, and, similarly, let V^u_t and V^c_t be the corresponding utilities for the attacker, and let $r = 1$. Moreover, suppose that $V^u_1 = V^c_1 = V^u_2 = V^c_2 = 1$, that is, the attacker is completely indifferent between the targets and defender strategy choices. Assume that $U^u_1 = -K$, and $U^c_1 =$

$U_2^u = U_2^c = 0$. That is, the defender prefers that the attacker attacks target 2. Finally, let the cost of leaving a target uncovered be 0, and coverage costs be $c_1 = c_2 = K/2$. Clearly, the optimal defender strategy is to cover nothing, because the attacker's indifference will result in him attacking target 2 in a strong Stackelberg equilibrium.

Now, suppose that we add some mean-zero random noise to the attacker's payoffs. With probability 1/24, the attacker's payoffs will be perceived to be ordered as follows: $\hat{V}_2^c < \hat{V}_2^u < \hat{V}_1^c < \hat{V}_1^u$. This ordering implies that the attacker will prefer to attack target 1 *no matter what the defender's strategy is*. Thus, the LP for target 2 will be infeasible, and the LP for target 1 is always feasible. The objective value of the LP for target 1 can be written as

$$\max_{q_1,q_2} \frac{K}{2} q_1 - \frac{K}{2} q_2,$$

where q_1 and q_2 are the probabilities of covering targets 1 and 2 respectively. Clearly, the optimal solution is to have $q_1 = 1$ and $q_2 = 0$, yielding an actual loss to the defender of $K/2$ (due to unnecessary security expenditures), compared to 0 in an optimal solution.

We now show that if we restrict the game to be strictly competitive (i.e., zero-sum), we do indeed obtain convergence to an optimal solution if we increase the number of samples. Let O^* be the true optimal utility of the defender (when the utilities are computed exactly), define $\hat{q}$ as an optimal solution when the player utilities are computed from samples, and let $O(\hat{q})$ denote the actual defender utility when the security policy is $\hat{q}$. Let $\hat{U}(o,t)$ denote the estimate of the defender's utility function.

Theorem 4 *Suppose that the game is strictly competitive and suppose that* $|\hat{U}(o,t) - U(o,t)| \leq \epsilon$ *for all* o, t. *Then* $O(\hat{q}) \geq O^* - 2\epsilon$.

Proof When the game is zero-sum, an optimal solution can be computed using the following simpler, single-LP formulation:

$$\max \quad r\left(\min_t \sum_o U(o,t) q_{o,t}\right) + (1-r)\left(\sum_{t,o} g_t U(o,t) q_{o,t}\right) - \sum_t \sum_o c_{o,t} q_{o,t} \tag{9a}$$

s.t.

$$\forall_{o,t}\ q_{o,t} \in [0,1] \tag{9b}$$

$$\forall_t \sum_o q_{o,t} = 1. \tag{9c}$$

First, note that the solution $\hat{q}$ obtained when utilities are estimated is feasible for program 9 where actual utilities are used. Thus, we can focus just on the objective value. Then,

$$\begin{aligned} O(\hat{q}) &= r\left(\min_t \sum_o U(o,t)\hat{q}_{o,t}\right) + (1-r)\left(\sum_{t,o} g_t U(o,t)\hat{q}_{o,t}\right) - \sum_t \sum_o c_{o,t}\hat{q}_{o,t} \\ &\geq r\left(\min_t \sum_o \hat{U}(o,t)\hat{q}_{o,t}\right) + (1-r)\left(\sum_{t,o} g_t \hat{U}(o,t)\hat{q}_{o,t}\right) - \sum_t \sum_o c_{o,t}\hat{q}_{o,t} - \epsilon \end{aligned}$$

$$\geq r\left(\min_t \sum_o \hat{U}(o,t)q^*_{o,t}\right) + (1-r)\left(\sum_{t,o} g_t \hat{U}(o,t)q^*_{o,t}\right) - \sum_t \sum_o c_{o,t}q^*_{o,t} - \epsilon$$

$$\geq r\left(\min_t \sum_o U(o,t)q^*_{o,t}\right) + (1-r)\left(\sum_{t,o} g_t U(o,t)q^*_{o,t}\right) - \sum_t \sum_o c_{o,t}q^*_{o,t} - 2\epsilon$$

$$= O^* - 2\epsilon.\square$$

Since the number of security configurations o and targets t is finite, we can obtain the uniform bound required by Theorem 4 directly from the law of large numbers. Thus, the theorem implies that as we take more samples, the resulting solutions converge to optimal in terms of the defender's utility.

5 Illustrations

In this section we illustrate our framework on two simple examples. The first is an artificial supply chain example that we constructed. The second uses a graph of interdependencies among critical infrastructure and key resource sectors obtained from the DHS and FEMA websites. For both these examples, we use the exact approach in the restricted setting with an attacker only attacking a single node and cascades that do not depend on security decisions.

5.1 A Simple Supply Chain

Consider a seven-node supply chain (directed acyclic graph) shown in Figure 1. We suppose that the entire supply chain (or at least the relevant security decisions) is controlled by a single firm which is primarily concerned with manufacturing two types of cars, one more profitable than the other. The actual components that ultimately comprise the cars are not intrinsically valuable to the manufacturer (or are valued so low relative to the final product as to make them effectively unimportant in this decision). All parts of the supply chain may be inspected at some cost c, or not (in which case no cost is incurred).

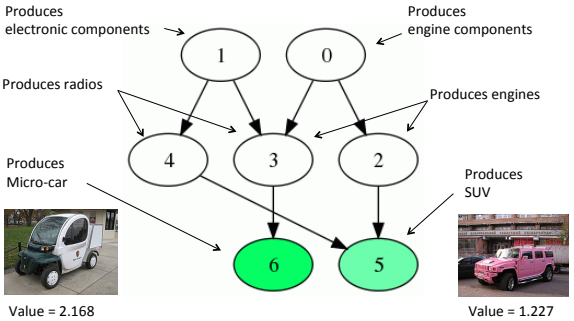

Fig. 1 A simple supply chain example.

The first step in our framework is to compute (or estimate) the expected utility for each node in the supply chain. To do this, we first specify the probability that an attacked node is affected (in this case, becomes faulty), $z(o,t)$. We let $z(o,t) = 1$ when node t is not inspected and $z(o,t) = 0$ when it is. Next, we must specify the contagion probabilities for each edge. We use $p_{t,t'} = 0.5$ for all edges here, and assume that they are independent of security decisions. Moreover, we assume that the attacker only attacks a single target.

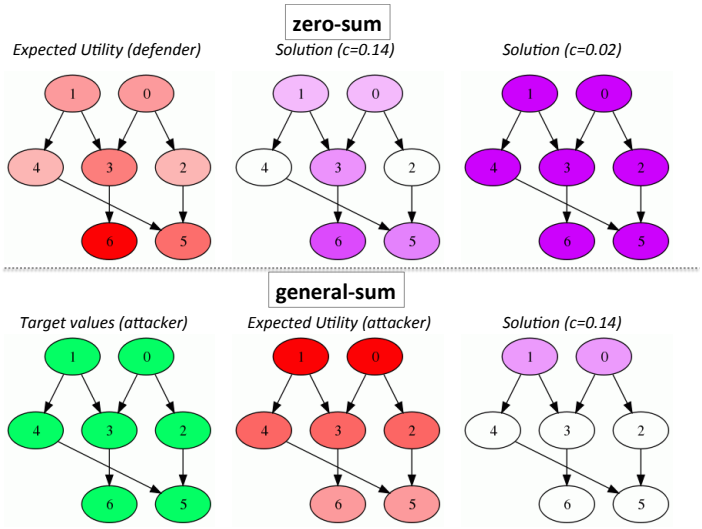

Fig. 2 Solutions for the zero-sum (top) and general-sum (bottom) variants of the simple supply chain example.

The results are color coded in Figure 2: the darker colors correspond to more valuable nodes. Note that while intrinsic worth is only ascribed to the final products, all components carry some value, due to their indirect impact on the final product (for example, a faulty part will, with some probability, make the component which uses it faulty as well). First, suppose that the game is zero-sum. We show the results for two different inspection costs, $c_{high} = 0.14$ and $c_{low} = 0.02$ in Figure 2 (top). The higher cost setting (Figure 2, top, middle solution) yields a security configuration in which five of the seven nodes incur some probability of inspection, with the heavier colors corresponding to a higher inspection probability. The low-cost setting (Figure 2, top, solution on the right) yields a solution in which every node is defended with probability 1. Next, consider a non-zero-sum variant in which the defender's utility is as before, while the attacker has uniform valuations (worths) over targets. The solution for this case with cost 0.14 is shown in Figure 2, bottom (the figure also shows the attacker's worths, as well as expected utilities derived from the dependency graph). This solution would at first sight seem quite unintuitive: the defender defends *only* the two targets at the top, which have the least value to him! The reason is that these targets happen to

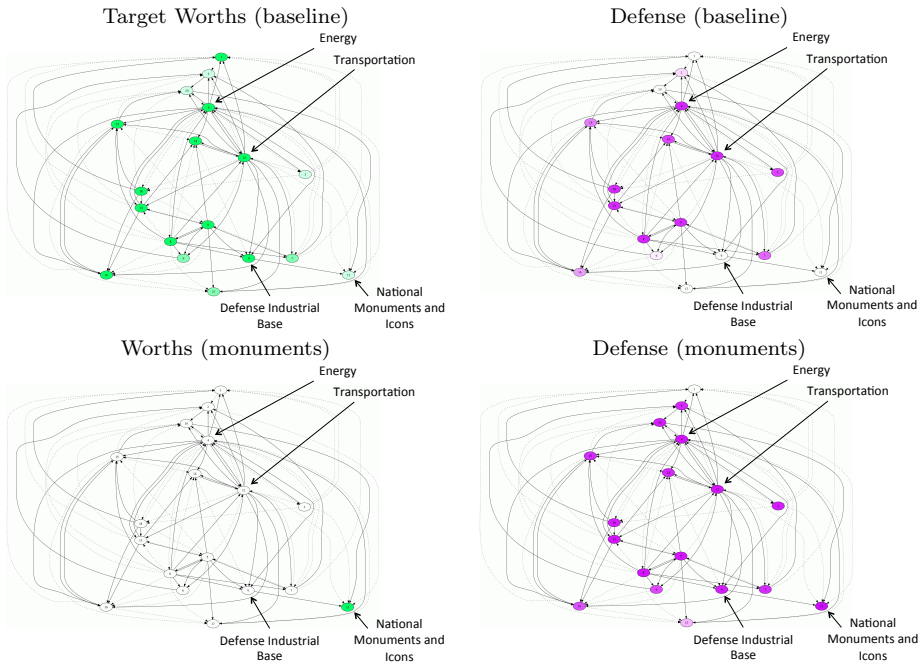

Fig. 3 Defending critical infrastructure and key resources. Top: baseline, with node worths based on rough economic impact. Bottom: an anomalous valuation function where only monuments and icons sector has positive worth.

have the highest expected utility for the attacker, since they result in the greatest utility from cascades, because the attacker's worths are identical for all targets. The defender will partially defend these targets, and given the defender's strategy, the attacker will still prefer to attack one of these, but will now be caught with positive probability.

5.2 Defending Critical Infrastructure: The Lobby Effect

Our second illustration of the framework developed above is on a graph representing dependencies between the critical infrastructure and key resource sectors listed on the DHS and FEMA websites. We used these websites to also infer the dependencies between the sectors, as well as the relative strengths of these dependencies. We then grouped these into "high" and "low" strength, with cascade probability set to 0.5 in the former and 0.1 in the latter cases. Defense cost is fixed at $c = 0.2$, and when a target is defended, it is assumed that no direct attack on it can succeed, while an attack on an undefended target succeeds with probability 1.

Figure 3 offers a view of the defense configuration in two cases: first (top), the baseline case in which importance of nodes is roughly representative of its economic value, and second (bottom), a comparative example in which only the monuments and icons sector is deemed valuable. One motivation for this particular

contrast is to illustrate a lobby effect which makes the value of a particular sector appear "out-of-whack" with economic considerations.

One interesting observation is that in the baseline case, even though every node has positive worth, not all nodes are defended with positive probability. For example, the defense industrial base sector is left undefended, as is the monuments and icons sector. In contrast, if there is a highly effective lobby on behalf of monuments and icons, to one's surprise *nearly all nodes are fully defended*, and defense expenditures are *much higher than in the baseline case*. This difference is due to the nature of dependencies: monuments and icons has either direct, or indirect but strong dependencies on almost all other sectors. The broader policy insight we may glean is that lobbying can have compounding effects on the budget, and a global impact well beyond what is intended by the direct lobbying effort due to systemic interdependencies.

6 Experiments

The goal of this section is to illustrate the value of our framework as a computational tool for designing security in interdependent settings. Specifically, we aim to demonstrate that our approach clearly improves on state-of-the-art alternatives, and offers a scalable solution for realistic security problems. We pursue this aim by randomly constructing dependency graphs using Erdos-Renyi (ER) and Preferential Attachment (PA) generative models (Newman 2010), as well as using a graph representing a snapshot of Autonomous System (AS) interconnections generated using Oregon routeviews (of Oregon Route Views Project 2013); this graph contains 6474 targets and 13233 edges and thus offers a reasonable test of scalability. In the ER model, every directed link is made with a specified and fixed probability p; we refer to it as ER(p). The PA model adds nodes in a fixed sequence, starting from an arbitrary seed graph with at least two vertices. Each node i is attached to m others stochastically (unless $i \leq m$, in which case it is connected to all preceding nodes), with probability of connecting to a node j proportional to the degree of j, d_j.

For the randomly generated networks, all data presented is averaged over 80-100 graph samples. Since we generate graphs that may include undirected cycles, we obtain expected utilities for all nodes on a given graph using 1000-10,000 simulated cascades (below we show that this is more than sufficient). Intrinsic worths w_t are generated uniformly randomly on $[0, 1]$. Cascade probabilities $p_{t,t'}$ (when independent of security strategies) were set to 0.5 unless otherwise specified. Except where otherwise specified, we restrict the defender to two security configurations at every target, one with a cost of 0 which stops attacks with probability 0 and one with a cost of c which prevents attacks with probability 1.

Where relevant, we run local search starting from 20 random starting points in addition to the three described above, unless specified otherwise. Finally, unless otherwise specified, we consider games with 50 targets for the general setting, and 100 targets for the restricted setting with security-independent cascades. We note that even with only 50 targets the running time of local search with random restarts on a given game instance was on the order of hours for large L. A single data point in many experiments below is therefore a product of as much as 400 processor-hours.

6.1 Scalability

An important question given the complexity of our framework is whether it can scale to realistic defense scenarios. To test this, we ran our *restricted* framework (i.e., a single target of attack and security-independent cascades) on the AS graph consisting of 6474 targets and 13233 edges. Since this is a large undirected graph containing cycles, a sampling approach was required, but the total running time (including both sampling and solving linear programs) amounted to less than 1 hour on a single 64 bit Linux 2.6.18-164.el5 computer with 96 GB of RAM and two quad-core hyperthreaded Intel Xeon 2.93 GHz processors. Given the importance of security, and the fact that *distributions* of security settings are computed once (or at least infrequently, as long as significant changes to the interdependency structure are not very frequent), this seems a relatively small computational burden.

6.2 Comparison to State-of-the-Art Alternatives

There are two prime computational alternatives to our framework. The first is to assume that targets are independent. While it is not difficult to show that in the worst case this can be quite a poor approximation, we offer empirical support to the added value of our approach below. The second is to use a well-known heuristic developed in the context of vaccination strategies on networks. This latter heuristic would in our case defend nodes in order of their connectivity (degree), until the defense budget is exhausted. Figure 4 (left) compares our approach in the restricted setting (single-target attack and security-independent cascades) to the former, while Figure 4 (middle, right) compares it to the latter. In both cases, computing optimal defense strategies using our framework yields much higher utility to the defender than the alternatives.

In the general case, one trivial way to compute an optimal solution is to search all possible defender (leader) actions, compute the best response of an attacker, and choose the action for the defender maximizing his utility. This trivial approach is linear in the size of the game. The problem is that the game size grows exponentially with the number of targets. Here we compare our simple local search routine with no random restarts to the optimal search in terms of running time and expected attained utility for the defender. The comparison is done in a simplified setting where we generate networks of interdependencies according to an Erdos-Renyi generative model with edge probability 0.4. We fix cascade probabilities to be $p_{t,t'} = 0.2$ whenever there is an edge between t and t' and t' is not defended; when t' is defended, we set $p_{t,t'} = 0$. We also fix defense costs at $c = 0.2$ and limit attacks to a single target ($L = 1$). Figure 5 (left) shows that local search is dramatically more scalable; indeed, optimal search quickly becomes intractable. Figure 5 (right) demonstrates that there are no (statistically significant) differences between the optimal objective value and that of the local search solution (confidence intervals omitted for clarity).

Since our model of security is partly motivated by epidemic spread (e.g., bioterrorism), it is natural to compare our approach to targeted vaccination on networks (widely recognized as state-of-the-art when initial infections are random (Pastor-Satorras and Vespignani 2002, Miller and Hyman 2007, Gallos et al 2007)), where

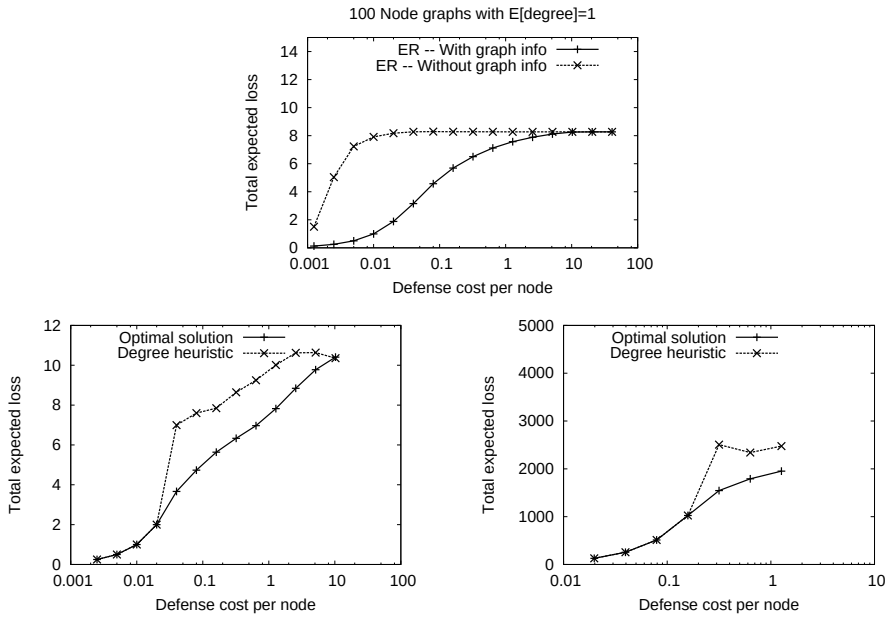

Fig. 4 Top: Comparison between our approach ("with graph info") and one assuming independence ("without graph info") using the ER(0.1) generative model. Bottom: Comparison of total expected loss (disutility to the defender) with the degree-based heuristic in the restricted setting. Bottom left: On PA graphs. Bottom right: On the AS graph.

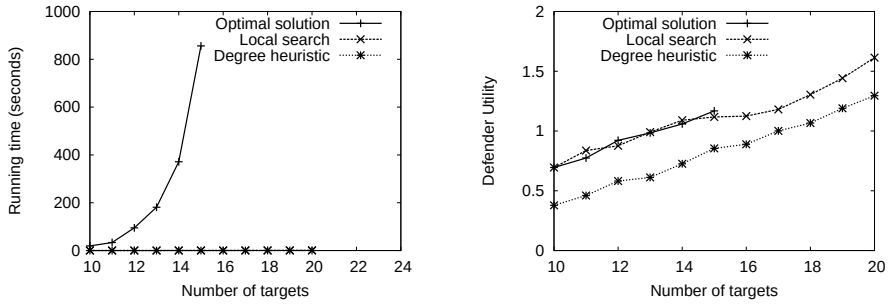

Fig. 5 Comparison between local search, optimal search, and targeted (degree-based) vaccination. Values are generated according to a Pareto distribution with $\gamma = 1.1$ (the results are robust to variations of this distribution and other parameters). Left: runtime comparison. Right: utility comparison.

nodes are defended in decreasing order of degree.[4] Figure 5 (right) shows that the *targeted vaccination* heuristic performs significantly worse than local search, even when we completely remove inoculated nodes from the network.

[4] There are a plethora of minor variations on this general heuristic, but the performance of the best tends to be similar to this baseline.

7 Applications to Interdependent Security Analysis

In this section we apply our framework to several network security domains. For simplicity, we restrict attention to zero-sum security games. As above, we consider ER and PA generative models, although we utilize a generalized version of PA. In a generalized PA model, connection probabilities are $\frac{(d_i)^\mu}{\sum_j (d_j)^\mu}$, such that when $\mu = 0$ the degree distribution is relatively homogeneous, just as in ER, $\mu = 1$ recovers the "standard" PA model, and large values of μ correspond to highly inhomogeneous degree distributions. Throughout, we use $\mu = 1$ unless otherwise specified. All parameters are set as in the experiments section, unless otherwise specified. In addition to the generative models of networks, we explore two networks derived from real security settings: one with 18 nodes that models dependencies among critical infrastructure and key resource sectors (CIKR), as inferred from the DHS and FEMA websites, and the second with 66 nodes that captures payments between banks in the core of the Fedwire network (Soramaki et al 2007). For the CIKR network, each node was assigned a low, medium, or high worth of 0.2, 0.5, or 1, respectively, based on perceived importance (for example, the energy sector was assigned a high worth, while the national monuments and icons sector a low worth). Each edge was categorized based on the importance of the dependency (gleaned from the DHS and FEMA websites) as "highly" or "moderately" significant, with cascade probabilities of 0.5 or 0.1 respectively. For the Fedwire network, all nodes were assigned an equal worth of 0.5, and cascade probabilities were discretely chosen between 0.05 and 0.5 in 0.05 increments depending on the weight of the corresponding edges in Soramaki et al (2007).

7.1 The Impact of Marginal Defense Cost

Our first analysis deals with the impact of marginal defense cost c on total defense expenditures (*total costs*), total losses due to failure cascades (or simply *total loss*), and total expenses incurred (or simply *total expense*, corresponding to negative defender utility, or the sum of total costs and total loss). The results for ER and BA (both with 100 nodes and average degree of 2), as well as CIKR and Fedwire networks are shown in Figure 6. All the plots feature a clear pattern: expected loss and (negative) utility are monotonically increasing, as expected, while total costs start at zero, initially rise, and ultimately fall (back to zero in 3 of the 4 cases). It may at first be surprising that total costs eventually fall even as marginal costs continue to increase, but this clearly must be the case: when c is high enough, the defender will not wish to invest in security at all, and total costs will be zero. What is much more surprising is the presence of two peaks in PA and Fedwire networks. Both of these networks share the property that there is a non-negligible fraction of nodes with very high connectivity (Newman 2010, Soramaki et al 2007). When the initial peak is reached, the network is fully defended, and as marginal costs rise further, the defender begins to reduce the defense resources expended on the less important targets. At a certain point, only the most connected targets are protected, and since these are so vital to protect, total costs begin increasing again. After the second peak is reached, c is finally large enough to discourage the defender from fully protecting even the most important targets, and the subsequent fall of total costs is no longer reversed.

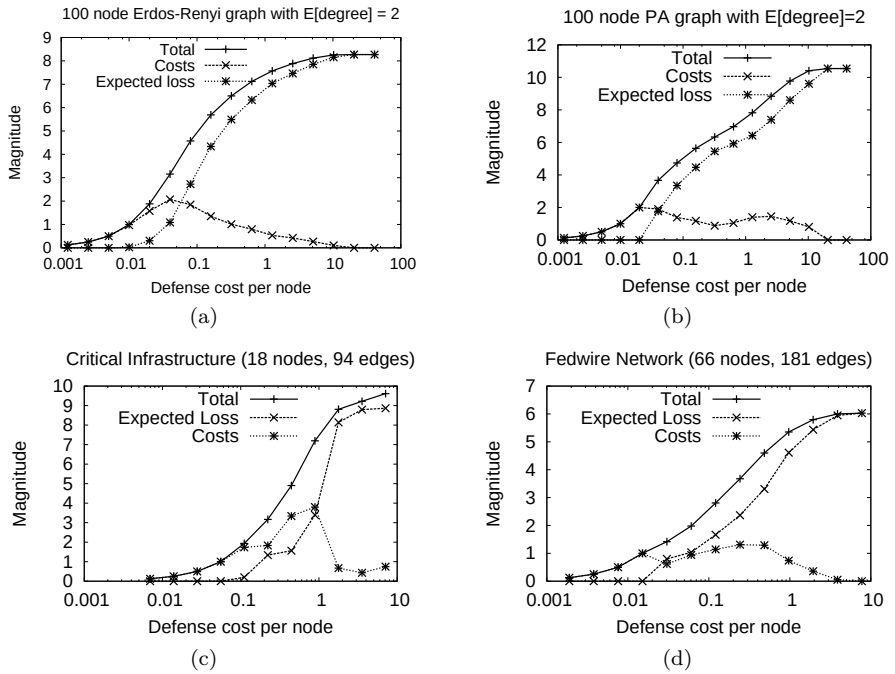

Fig. 6 Expected loss, cost, and their sum in (a) 100-node ER(0.2), (b) 100-node PA, (c) 18-node critical infrastructure, and (d) 66-node core of the Fedwire networks as defense cost increases. The results for ER and PA are averages over 100 stochastic realizations of these networks.

7.2 Changing the Number of Attacked Targets

Our next analysis concerns an important extension that traditional Stackelberg security game approaches cannot handle in a scalable way: allowing an attacker to attack more than a single target. Specifically, we study the impact of the number of targets L an attacker can attack on total defense expenditures, total losses due to failure cascades, and total expenses incurred. We do this while keeping cascade probabilities $p_{t,t'}$ independent of defense configuration; we set all of these to $p = 0.2$. Moreover, we generate the dependency graphs based on the Erdos-Renyi generative model with edge probabilities fixed at 0.05.

Total defense expenditures (costs) are shown in Figure 7 (left) for three different values of cost per target defended, c (we also call this marginal defense cost). The difference between the three cost regimes is negligible when only a single target can be attacked, yet the behavior of defense expenditures as L increases exhibits striking qualitative differences, and techniques that only consider $L = 1$ would therefore be blind to these. In all three cases, there is a critical threshold L_c of the number of attacked targets. When $L < L_c$, defense expenditures remain very low and relatively stable, but when $L \approx L_c$, expenditures rise sharply, ultimately leveling off at a much higher value which again remains relatively stable for $L > L_c$. Surprisingly, increasing marginal defense cost c causes L_c to increase: it takes greater attacker capability to stimulate the defender to invest more in

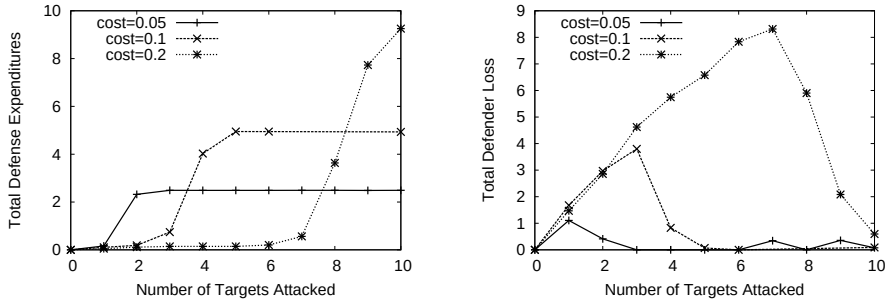

Fig. 7 Total defense expenditures (left) and losses due to cascading failures (right) as the number of attacked targets increases for three different defense cost values (i.e., cost of defending a single target): 0.05, 0.1, and 0.2.

security; however, the rise in security investment is greater for higher c once the threshold L_c is reached.

Figure 7 (right) shows the total loss as a function of the attacker's capability L. The result is somewhat counter to initial intuition: the total losses are non-monotonic. The reason comes from the observation we had already made about total expenditures: until a threshold L_c is reached, few defense resources are deployed, and total losses rise, but after the threshold, defense expenditures ramp up substantially, and, as long as c is sufficiently low, the defender will ultimately come to defend every target. The pattern of total defender expenses (the sum of losses and total expenditures; not shown) is largely predictable: expenses increase monotonically with L, and are higher for higher c.

7.3 Resilience to Targeted Attacks: The Impact of Network Structure

One of the important streams in the network science literature is the question of relative resilience of different network topologies to failures, random or targeted. One feature of network topology, the distribution of degrees (number of node neighbors) has received particular attention. There is, in particular, one measure of degree distribution—its *homogeneity*—that plays an especially important role. (For example, an Erdos-Renyi network has a homogeneous degree distribution, while a heavy-tailed distribution, such as Pareto, is inhomogeneous.) Two very disparate streams of literature tie homogeneity of the degree distribution to network resilience. The first of these features a widely replicated finding that networks with an inhomogeneous (e.g., scale-free) degree distribution exhibit poor tolerance to targeted attacks as compared to Erdos-Renyi graphs (Albert et al 2000, Newman 2010). On the other hand, when failures are random (no attacks), scale-free graphs have been found to be more resilient than Erdos-Renyi counterparts. The second stream of literature demonstrates that scale-free graphs are particularly easy to defend against epidemic spread, as inoculating high-degree nodes dramatically reduces the expected number of infections; however, this stream does not model targeted attacks.

Our framework allows us to cleanly unify both these streams of literature and present a much more refined analysis of the relationship between the homogeneity

of the degree distribution and network resilience to cascading failures. Specifically, we undertake here a study of the total losses and costs incurred by the defender under a variety of network regimes.

As a starting point, consider Figure 8 (left), which shows the defender's utility for three different network topologies, PA, ER, and Fedwire as a function of cost c. The results presented in this figure are generated based on our special case when cascade probabilities $p_{t,t'}$ are independent of security decisions, and when the attacker can only attack a single target (through the rest of this paper, we focus only on the impact of deliberate attacks and fix the probability of "nature" to 0). In light of the previous discussion, what we can readily observe in Figure 8 (left) would appear quite remarkable: network topology seems to play little role in resilience. A superficial difference here is that we consider a cascading failure model, while most of the previous work on the subject involving targeted attacks focused on diminished connectivity due to attacks. We contend that the most important distinction, however, is that previous work studying resilience did not account for a simple observation that most important targets of potential attacks are also most heavily defended; indeed, to the best of our knowledge, none of the previous work on resilience in the face of attacks allows for endogenous defense decisions. Indeed, we can observe from the figure that once defense costs c are sufficiently high, PA leads to substantially higher losses (greater disutility to the defender), confirming previous results in this rather extreme setting.

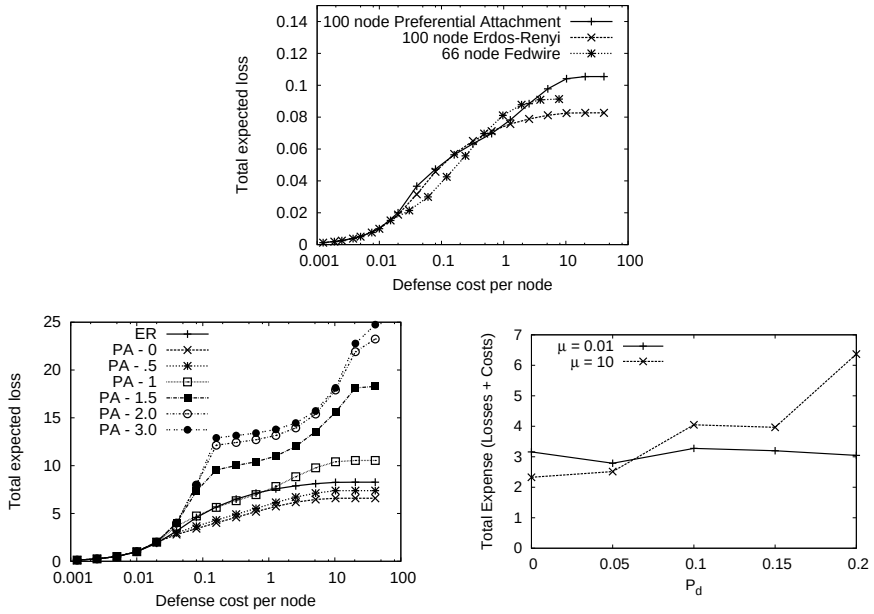

Fig. 8 Top: Expected total loss: comparison across different network structures. Bottom, left: Expected defender disutility in the generalized PA model as we vary μ (keeping average degree fixed at 2). ER is also shown for comparison. Bottom, right: Total defender expenses (total expenditures + losses from cascades) as a function of p_d for $\mu = 0.01$ (nearly Erdos-Renyi) and $\mu = 10$ (highly hub-like structure). Cascade probabilities of undefended nodes are fixed at $p = 0.2$. Cost of defending each node is fixed at 0.5.

To investigate the impact of network topology on resilience further, we consider the generalized PA model in which we systematically vary the homogeneity of the degree distribution by way of the parameter μ. Figure 8 (middle) shows the results for the special case of security-independent cascades with the attacker restricted to attack only one target. In this graph, we do observe clear variation in resilience as a function of network topology, but the operational factor in this variation is *homogeneity in the distribution of expected utilities, rather than degrees*: increasing homogeneity of the utility distribution *lowers* network resilience. This seems precisely the opposite of the standard results in network resilience, but the two are in fact closely related, as we now demonstrate. Superficially, the trend in the figure seems to follow the common intuition in the resilience literature: as the degree distribution becomes more inhomogeneous (more star-like), it becomes more difficult to defend. Observe, however, that ER is actually more difficult to defend than PA with $\mu = 0$. The lone difference of the latter from ER is the fact that nodes that enter earlier are more connected and, therefore, the degree distribution in the PA variant should actually be more *inhomogeneous* than ER! The answer is that random connectivity combined with inhomogeneity of degrees actually makes the distribution of *utilities* less homogeneous in PA with $\mu = 0$, and, as a result, fewer nodes on which defense can focus as compared to ER. On the other hand, as the graph becomes more star-like, the utilities of all nodes become quite similar; in the limiting case, all nodes are only two hops apart, and attacking any one of them yields a loss of many as a result of cascades.

Our final exploration in this vein considers a more general setting where security decisions have some (varying) effect on the likelihood of cascade spread. Specifically, define the parameter p_d as the probability that a cascade spreads to a node which is defended, and fix the probability that a cascade spreads to an undefended node at 0.2. Thus, if $p_d = 0$, we have an instance of perfect inoculation: if a node is defended (inoculated), it is equivalent to removing that node from the network entirely. At the other end of the spectrum, $p_d = 0.2$ will imply that defense has no impact on the probability of cascades. Figure 8 (right) presents the total defender disutility (losses due to cascading failures + defense costs incurred) as a function of p_d for two extreme cases of μ, one ($\mu = 0.01$) corresponding to a highly homogeneous degree distribution, while the other ($\mu = 10$) to a highly inhomogeneous one. The two classes of graphs exhibit dramatically different resilience behavior as a function of p_d which paints a more complete picture than the literature on network resilience to date. When $p_d = 0.2$ (equal to the cascade probability when a node is not defended), hub-like structures are far less resilient to targeted attacks as compared to a graph with a homogeneous degree distribution; this is inline with previous results, which suggest that inhomogeneous graphs are less resilient (Albert et al 2000). With $p_d = 0$, on the other hand, hub-like networks are highly resilient, since it suffices for the defender to target the few hubs; this is similar to the observation that targeted vaccination is more effective on scale-free graphs (Pastor-Satorras and Vespignani 2002), although in that stream of literature failures are assumed to arise randomly, rather than in a targeted manner. At the high level, the resilience of the hub-like network decreases with increasing p_d, whereas a homogeneous network remains relatively unaffected by p_d. The reason is that when p_d is high, a hub-like structure implies low diameter. Unless the hub itself is actually removed from the network by the defense action, it can serve as the conduit for failure cascades started at other nodes; therefore,

when p_d is high the defense of the hub is insufficient to make the network resilient, and vastly greater defense expenditures are required. In contrast, a homogeneous network has no such hubs with global connectivity, and is therefore less sensitive to p_d.

There is another aspect of network topology that has an important impact on resilience: network density. Figure 9 (left) shows a plot of an Erdos-Renyi network with the probability of an edge varying between 0.0025 to 0.08 (average degree between .25 and 8) and cost c fixed at 0.04. Clearly, expected utility and loss of the defender are increasing in density, but it is rather surprising to observe how sharply they jump once the average degree exceeds 1 (the ER network threshold for a large connected component); in any case, network density has an unmistakable impact. The reason is intuitive: increased density means more paths between targets, and, consequently, greater likelihood of large cascades in the event that a target is compromised. Total cost initially increases in response to increased density, in part to compensate for the increased vulnerability to attacks, but eventually falls, since it is too expensive to protect everything, and anything short of that is largely ineffective.

7.4 Interaction Between Cascade Probabilities and the Number of Targets Attacked

In this section we study the impact of the cascade probability to a defended node, p_d, while at the same time varying the attacker's capability L. As in the previous section, we maintain the probability that a failure cascades to an undefended node at 0.2. We generate the dependency graphs based on the Erdos-Renyi generative model with edge probabilities fixed at 0.05.

Figure 9 (right) shows the total defense expenditures (cost per target defended fixed at 0.1). While the differences are relatively small, there is a clear pattern: when the number of targets attacked is low (below L_c), increasing the impact of defense on cascade probability prompts the defender to increase investment in security (defense has an increasing marginal value), but once attacker capabilities are high, defense expenditures fall when p_d falls (i.e., defense has higher impact). In the latter case, making the network sufficiently resilient to attacks requires relatively fewer protected nodes and, therefore, lower defense expenditures. Indeed, decreasing p_d systematically reduces total defender expenses (sum of losses due to cascades and defense costs).

8 Conclusion

We presented a framework for computing and approximating optimal security policies in network domains. Our framework involves a general model of asset interdependencies, which we instantiate using a dependency graph between assets and a cascading failures model based on a common epidemiological model of disease contagion. In the general case, we offer an effective approximation technique based on a combination of submodular optimization and a local search heuristic. Moreover, we show that in an important special case which restrict the attacker's capabilities to only attack one target and restricts the cascade probabilities to

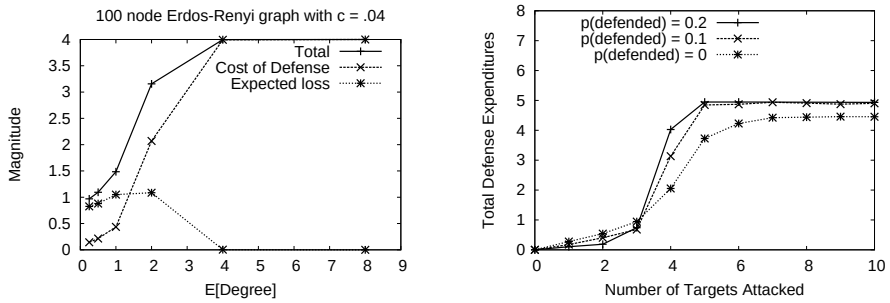

Fig. 9 Left: Expected loss, cost, and their sum in 100-node Erdos-Renyi networks as a function of network density (equivalently, expected degree). Right: Total defense expenditures as the number of attacked targets increases for three different values of p_d (p(defense) in the legend): 0.2 (cascades independent of defense), 0.1 (defense partially protects from cascades), and 0 (defense fully protects from cascades). Graphs are ER(0.05).

be independent of security decisions, we can effectively decouple simulations that estimate player expected utilities from a linear programming formulation which subsequently computes an optimal security policy. Our results demonstrate the value of our approach as compared to alternatives, and show that it is scalable to realistic security settings. Furthermore, we used our framework to analyze four models of interdependencies: two based on random graph generation models, a simple model of interdependence between critical infrastructure and key resource sectors, and a model of the Fedwire interbank payment network.

9 Acknowledgments

Much of this work was performed while Yevgeniy Vorobeychik was at Sandia National Laboratories and Joshua Letchford was at Duke University. Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

References

Albert R, Jeong H, Barabasi AL (2000) Error and attack tolerance of complex networks. Nature 406:378–382

Anderson RJ (2008) Security Engineering, 2nd edn. Wiley

August T, Tunca TI (2011) Who should be responsible for software security? A comparative analysis of liability policies in network environments. Management Science 57(5):934–959

Avenhaus R, von Stengel B, Zamir S (2002) Inspection games. In: Aumann R, Hart S (eds) Handbook of Game Theory, Elsevier Science Publishers, pp 1947–1987

Brown G, Carlyle M, Salmeron J, Wood K (2006) Defending critical infrastructure. Interfaces 36(6):530–544

Brown GG, Carlyle WM, Harney RC, Skroch EM, Wood RK (2009) Interdicting a nuclear-weapons project. Operations Research 57(4):866–877

Cavusoglo H, Mishra B, Raghunathan S (2004) A model for evaluating IT security investments. Communications of the ACM 47(7):87–92

Cavusoglu H, Raghunathan S (2004) Configuration of detection software: A comparison of decision and game theory approaches. Decision Analysis 1(3):131–148

Cavusoglu H, Mishra B, Raghunathan S (2005) The value of intrusion detection systems in information technology security architecture. Information Systems Research 16(1):28–46

Cavusoglu H, Cavusoglu H, Zhang J (2008) Security patch management: Share the burden or share the damage. Management Science 54(4):657–670

Cavusoglu H, Raghunathan S, Cavusoglu H (2009) Configuration of and interaction between information security technologies: The case of firewalls and intrusion detection systems. Information Systems Research 20(2):198–217

CERT (1999) Frequently asked questions about the melissa virus. CERT Program at Software Engineering Institute, Carnegie Mellon University, URL `http://www.cert.org/tech_tips/Melissa_FAQ.html`

Conitzer V, Korzhyk D (2011) Commitment to correlated strategies. In: Twenty-Fifth National Conference on Artificial Intelligence, pp 632–637

Conitzer V, Sandholm T (2006) Computing the optimal strategy to commit to. In: Seventh ACM Conference on Electronic Commerce, pp 82–90

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

Cremonini M, Nizovtsev D (2006) Understanding and influencing attackers' decisions: Implications for security investment strategies. In: Workshop on the Economics of Information Security

Dodds PS, Watts DJ (2005) A generalized model of social and biological contagion. Journal of Theoretical Biology 232:587–604

Domingos P (1999) Metacost: A general method for making classifiers cost-sensitive. In: ACM International Conference on Knowledge Discovery and Data Mining

Duggan DP, Thomas SR, Veitch CKK, Woodard L (2007) Categorizing threat: Building and using a generic threat matrix. Tech. rep., Sandia National Laboratories, sAND2007-5791

Energy Sector Control Systems Working Group (2011) Roadmap to achieve energy delivery systems cybersecurity. Energetics, Inc, URL `https://www.controlsystemsroadmap.net/ieRoadmap\%20Documents/roadmap.pdf`

Gallos LK, Liljeros F, Argyrakis P, Bunde A, Havlin S (2007) Improving immunization strategies. Physical Review E 75:045,104

Grossklags J, Christin N, Chuang J (2008) Secure or insure? A game-theoretic analysis of information security games. In: Seventeenth International World Wide Web Conference, pp 209–218

Jain M, Kardes E, Kiekintveld C, Tambe M, Ordonez F (2010a) Security games with arbitrary schedules: A branch and price approach. In: Twenty-Fourth National Conference on Artificial Intelligence

Jain M, Tsai J, Pita J, Kiekintveld C, Rathi S, Tambe M, Ordóñez F (2010b) Software assistants for randomized patrol planning for the lax airport police and the federal air marshal service. Interfaces 40:267–290

Jain M, Korzhyk D, Vanek O, Conitzer V, Pechoucek M, Tambe M (2011) A double oracle algorithm for zero-sum security games on graphs. In: Tenth International Conference on Autonomous Agents and Multiagent Systems

Kempe D, Kleinberg JM, Éva Tardos (2003) Maximizing the spread of influence in a social network. In: Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp 137–146

Kiekintveld C, Jain M, Tsai J, Pita J, Ordóñez F, Tambe M (2009) Computing optimal randomized resource allocations for massive security games. In: Proceedings of the Eighth International Conference on Autonomous Agents and Multiagent Systems

Korzhyk D, Conitzer V, Parr R (2010) Complexity of computing optimal stackelberg strategies in security resource allocation games. In: In AAAI-10

Krutz R, Vines RD (2001) The CISSP Prep Guide. Wiley Computer Publishing

Kunreuther H, Heal G (2003) Interdependent security. Journal of Risk and Uncertainty 26(2-3):231–249

Lee W, Miller M, Stolfo S, Jallad K, Park C, Zadok E, Prabhakar V (2002) Toward cost-sensitive modeling for intrusion detection. Journal of Computer Security 10(1/2):5–22

Letchford J, Conitzer V (2010) Computing optimal strategies to commit to in extensive-form games. In: Eleventh ACM conference on Electronic commerce, ACM, New York, NY, USA, EC '10, pp 83–92

Letchford J, Vorobeychik Y (2012) Computing optimal security strategies for interdependent assets. In: Twenty-Eighth Conference on Uncertainty in Artificial Intelligence

Miller JC, Hyman JM (2007) Effective vaccination strategies for realistic social networks. Physica A 386:780–785

MITRE (2012) Common attack pattern enumeration and classification. URL `http://capec.mitre.org/`

Mounzer J, Alpcan T, Bambos N (2010) Integrated security risk management for IT-intensive organizations. In: Sixth International Conference on Information Assurance and Security, pp 329–334

Nehme MV (2009) Two-person games for stochastic network interdiction: Models, methods, and complexities. PhD thesis, The Unversity of Texas at Austin

Nemhauser G, Wolsey L, Fisher M (1978) An analysis of the approximations for maximizing submodular set functions. Mathematical Programming 14:265–294

Newman M (2010) Networks: An Introduction. Oxford University Press

Ogut H, Menon N, Raghunathan S (2005) Cyber insurance and IT security investments: Impact of interdependent risk. In: Workshop on the Economics of Information Security

Ogut H, Cavusoglu H, Raghunathan S (2008) Intrusion-detection policies for IT security breaches. INFORMS Journal on Computing 20(1):112–123

of Oregon Route Views Project U (2013) Online data and reports. http://www.routeviews.org

Paruchuri P, Pearce JP, Marecki J, Tambe M, Ordóñez F, Kraus S (2008) Playing games with security: An efficient exact algorithm for Bayesian Stackelberg games. In: Proceedings of the Seventh International Conference on Autonomous Agents and Multiagent Systems, pp 895–902

Pastor-Satorras R, Vespignani A (2002) Immunization of complex networks. Physical Review E 65:036,104

Pita J, Jain M, Ordóñez F, Portway C, Tambe M, Western C, Paruchuri P, Kraus S (2009) Using game theory for los angeles airport security. AI Magazine 30(1):43–57

Provost F, Fawcett T (1997) Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions. In: KDD, pp 43–48

Roberson B (2006) The colonel Blotto game. Economic Theory 29:1–24

Rosencrance L (2002) Melissa virus author sentenced. PC World, URL `http://www.pcworld.com/article/97964/melissa_virus_author_sentenced.html`

Shieh E, Yang R, Tambe M, Baldwin C, DiRenzo J, Maule B, Meyer G (2012) PROTECT: A deployed game theoretic system to protect the ports of the United States. In: Proceedings of the Eleventh International Conference on Autonomous Agents and Multiagent Systems, pp 13–20

Soramaki K, Bech ML, Arnold J, Glass RJ, Beyeler W (2007) The topology of interbank payment flows. Physica A 379:317–333

Stamp JE, Laviolette RA, Phillips LR, Richardson BT (2009) Final report: Impacts analysis for cyber attack on electric power systems. Sandia National Laboratories Technical Report, SAND2009-1673

von Stengel B, Zamir S (2010) Leadership games with convex strategy sets. Games and Economic Behavior 69(2):446–457

Tsai J, Yin Z, young Kwak J, Kempe D, Kiekintveld C, Tambe M (2010) Urban security: Game-theoretic resource allocation in networked physical domains. In: Twenty-Fourth National Conference on Artificial Intelligence

Tsai J, Nguyen TH, Tambe M (2012) Security games for controlling contagion. In: Twenty-Sixth National Conference in Artificial Intelligence, to appear

Ulvila JW, Gaffney JE (2004) A decision analysis method for evaluating computer intrusion detection systems. Decision Analysis 1(1):35–50

Vorobeychik Y, Wellman MP (2008) Stochastic search methods for Nash equilibrium approximation in simulation-based games. In: Seventh International Conference on Autonomous Agents and Multiagent Systems, pp 1055–1062

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

Woodruff DL (ed) (2003) Network Interdiction and Stochastic Integer Programming. Kluwer Academic Publishers

Yue WT, Bagchi A (2003) Tuning the quality parameters of a firewall to maximize net benefit. In: International Workshop on Distributed Computing, pp 321–329

Zhuang J, Bier V (2007) Balancing terrorism and natural disasters—Defensive strategy with endogenous attacker effort. Operations Research 55(5):976–991