ARGONNE NATIONAL LABORATORY
9700 South Cass Avenue
Argonne, Illinois 60439

# Optimal Security Response to Attacks on Open Science Grids

**Mine Altunay, Sven Leyffer, Jeffrey T. Linderoth, and Zhen Xie**

Mathematics and Computer Science Division

Preprint ANL/MCS-P1593-0309

March 25, 2009

# Contents

# Optimal Security Response to Attacks on Open Science Grids

Mine Altunay* Sven Leyffer[†] Jeffrey T. Linderoth[‡] Zhen Xie[§]

March 25, 2009

**Abstract**

Cybersecurity is a growing concern, especially in open grids, where attack propagation is easy because of prevalent collaborations among thousands of users and hundreds of institutions. The collaboration rules that typically govern large science experiments as well as social networks of scientists span across the institutional security boundaries. A common concern is that the increased openness may allow malicious attackers to spread more readily around the grid.

We consider how to optimally respond to attacks in open grid environments. To show how and why attacks spread more readily around the grid, we first discuss how collaborations manifest themselves in the grids and how this collaboration model affects the security risk model of grid participants. We present a new grid model and use optimization techniques to calculate the security risk associated with each grid participant. Given an attack scenario, our optimization model aims to minimize threat levels at unaffected participants while maximizing the uninterrupted scientific production (continuing collaborations). By manipulating some of the collaboration rules (e.g., suspending a collaboration or shutting down a site), the model finds optimal response scenarios to contain an attack scenario.

**Keywords:** Cybersecurity, optimization, network, multi-objective, integer optimization.
**AMS-MSC2000: 90C11, 90C29, 90B10, 90B25, 90B80.**

## 1 Introduction and Background

The emergence of open grid infrastructures, such as the Open Science Grid [15], TeraGrid [18] and Earth System Grid [6], has enabled scientists to exploit unprecedented computing resources for data-intensive research opportunistically share the computing resources. A common concern is that the increased openness may allow malicious attackers to spread more readily around the grid. Thus, cybersecurity has become a growing concern especially in open grids, where the increasing number of collaborations makes attack propagation easy.

The open grids seek to bring scientists together with the necessary computational powers by making institutional barriers transparent. Therefore, in grids, it is the norm to have cross-domain

*Fermi National Laboratory, maltunay@fnal.gov

[†]Mathematics and Computer Science Division, Argonne National Laboratory, leyffer@mcs.anl.gov

[‡]Department of Industrial and Systems Engineering, Department of Computer Sciences University of Wisconsin-Madison, linderoth@wisc.edu

[§]Mathematics and Computer Science Division, Argonne National Laboratory, zhenxie@mcs.anl.gov

accesses, where computing power and users are spread across multiple institutional security domains. For example, the widely anticipated Large Hadron Collider (LHC), the world's largest accelerator, will generate data at petabyte scale that will require exascale computing powers. Two major experiments on the LHC are the Compact Muon Solenoid (CMS) and A Toroidal LHC Apparatus (Atlas). Each experiment has approximately 3,000 scientists who will collaborate to process and analyze the data in the Open Science Grid (OSG). Likewise, the Earth System Grid (ESG) has close to 10,000 scientists collaborating to analyze climate data sets. The scientists and the resources devoted to these experiments are all drawn from different institutions, such as Department of Energy laboratories, supercomputing centers, and universities.

The security perimeters traditionally defined at institutional boundaries are ineffective against attacks on these grids. The attackers take advantage of grid middleware that is designed to cross these boundaries seamlessly, thus making it difficult to contain an incident at a single institution. In order to succeed, the response systems should use grid-specific information to their advantage. To do so, we must understand how collaborations form in grids and how this collaboration model affects the security of grid participants. We also must understand how attack spread patterns take advantage of collaborations between grid participants and how we can use this knowledge to our advantage to predict an attack spread and take preventive measures.

Currently, no tools exists that can help the open grid with attack response and risk analysis. As a consequence, for example, the OSG security team spends from 3 to 14 days for each incident response, on average requiring 5 days to close an incident investigation. A significant reason for this challenging problem is the difficulty of collecting information in a grid environment, where the architecture is distributed across hundreds of institutions. Grid architectures naturally favor autonomous, distributed, and reliable services, where each service is run by a different grid participant. During an incident response, a team of security experts needs to gain a "big-picture" view of the grid to make security decisions. Gathering security information from different institutions is time consuming and prone to error. Our collaboration-based grid model collects the necessary information ahead of time and builds a model of the grid that enables security experts easily to analyze attack scenarios.

Our risk model takes the collaboration-based grid model as input and uses optimization techniques to calculate the risk level for each grid participant. The risk of a compromise is calculated based on the collaborations a grid member has participated in and whether the collaborators have been affected by the compromise. By calculating the risk level of unaffected participants, we observe how collaborations may help spread an attack. Our optimization techniques allow us to find an optimal response to the attack. We optimize by changing some of the collaboration rules, for example, shutting down a resource or interrupting or monitoring a collaboration between two participants. Our goal is to minimize the risk levels for all participants while maximizing the grid productivity. We define grid productivity as the uninterrupted continuity of any scientific activity over the grid.

The remainder of this paper is organized as follows. In Section 2, we discuss grid computing and how collaborations shape within grids, and present a new collaboration-based grid model. Section 3 we describe our approach for estimating threat levels at unaffected sites, which will form the basis for the optimal response model that we present in Section 4. In Section 5 we present a case study that demonstrates the viability of our approach using real network data from the Open Science Grid. In Section 6, We close with a few comments on how to extend our models.

**Notation** We will use the following conventions throughout. Quantities related to the graph of the network appear in cursive script (e.g. $\mathcal{G}$). We use upper-case characters to denote parameters and constants. Lower-case characters are reserved for variables. The symbols and notation used in this paper are summarized in Table 1 in Section 5.

## 2 Collaboration-based Grid Model

A grid resource is a computing element or a storage element. A grid site is defined as a collection of grid resources under a single administrative domain. A grid site belongs to an institution such as a university or a Department of Energy laboratory. The site security is managed by the institution.

Security has played a crucial role since the inception of grid computing, partly due to the lack of security models and mechanisms for distributed collaborations. Globus Security Infrastructure (GSI) [5] has addressed the issue of authentication and authorization. GSI provides a common authentication layer between grid resources by leveraging Public Key Infrastructure (PKI) and X.509 credentials [13]. A grid resource uses GSI to authenticate incoming user requests.

In grid computing, science experiment is modeled as a Virtual Organization (VO). VO managers define the collaboration rules among the scientists and experiment resources, including individual duties in each group and the sites and resources contributing to the experiment. VO Management Service (VOMS) [2] and Community Authorization Service (CAS) [16] are two tools developed to capture authorization-related collaboration rules. Both tools have a database of groups and scientist names in a VO. Sites that wish to contribute to a VO point to the VOMS or CAS server and download the group and scientist names. Tools such as GUMS [11] and grid-map file, which is included in Globus Toolkit [7], provides this functionality for the sites.

A VO can be represented as a collaboration graph [9, 3] such that two nodes, representing scientists, are linked if both belong to the same experiment. When finer granularity is needed, two nodes can be linked with an edge if both belong to the same VO group. A node representing a VO group can be linked to another node if both groups are in the same VO.

However, a VO collaboration graph that solely consists of scientists does not tell us the whole scope of the collaborations taking place in the VO. A VO rarely consists of only humans; a VO requires grid resources to achieve its goal. Although scientists have always collaborated by engaging in in-person discussions and exchanging papers, the collaboration form that benefits the most from grid computing is the one that involves access to shared grid resources. In fact, grid computing helps very little to the direct, person-person collaboration form. Thus, a realistic collaboration-based grid model should include both grid resources and scientist and should indicate how resources are utilized by the VOs. Our new grid model enables us to analyze how an attack exploits the commonly used resources and scientists to spread across multiple grid sites.

Figure 1 illustrates our new model. The CMS VO has access to resources X and Z, while Atlas VO has access to resources X, Y, and W. Resources X, Y, Z, and W all belong to different grid sites and different security domains. By contributing to Atlas VO, however, resources Y, X, and W can be linked to one another. Likewise, resources X and Z can be linked together because of their joint contribution to the CMS VO. An attack over one of these resources is likely to spread to the linked resources. Consider an attack scenario (over Figure 1). A security incident occurs at resource Z such that the account of *user1* account is compromised and his proxy credentials are stolen. The credentials of *user1* allow access to resource X as well. Thus, resource X is vulnerable to attack even though it belongs to a different grid site from that of resource Z. Let us assume that the attacker accesses and compromises the account CMS B1 on resource X. Not only would the proxy credentials of *user3* be stolen, but the attacker would also have a chance of compromising other accounts on resource X (Atlas *user6* and Atlas *user7*). There is a nonzero probability that the attacker can find and exploit a system-level vulnerability on resource X to elevate his privileges

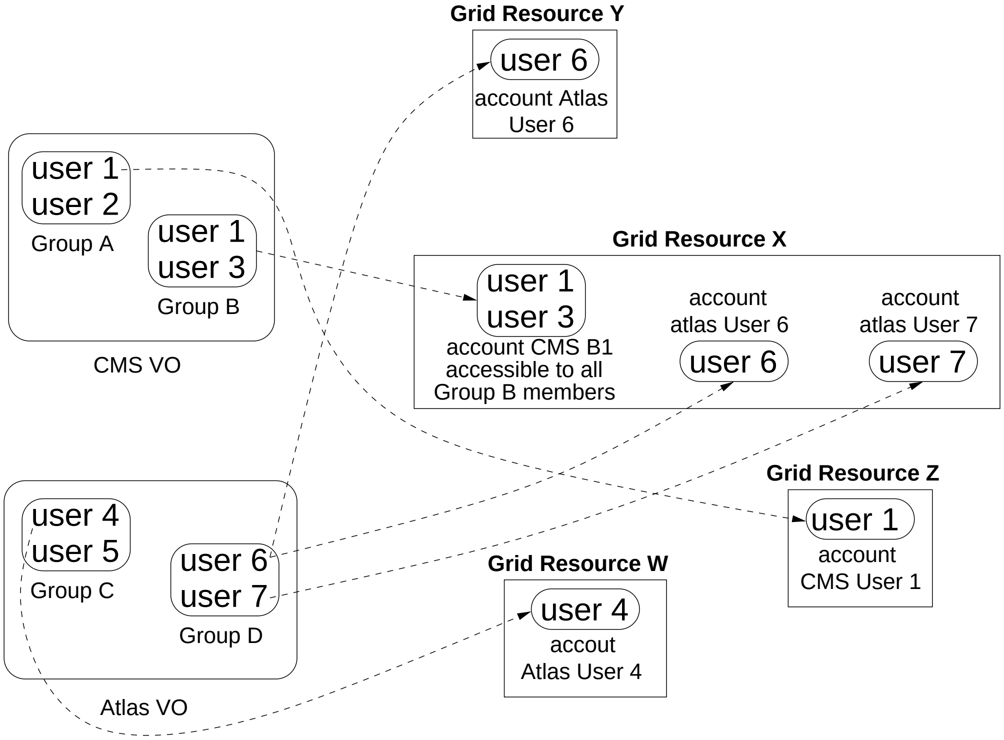


Figure 1: Collaborations forming in the grid. Resources X, Y, Z, and W belong to different grid sites. Scientists are mapped into Unix accounts on resources indicated by dashed arrows. Some scientists, e.g. *user1* and *user3*, are mapped into the same account, whereas, others, *user6* and *user7*, has individual accounts. Scientists (*user1*) can be members of multiple groups (groups A and B)or VOs simultaneously.

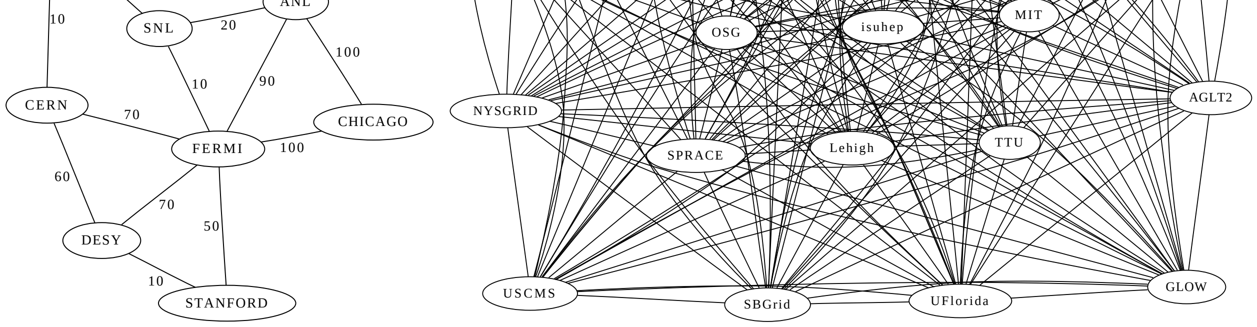
to root and compromise other accounts on resource X. In fact, observation of past attacks over OSG, ESG, and TG show that once an attacker is inside the resource, his chances of gaining root privileges increase significantly.

We identify four types of linkage between resources and humans: (1) two resources are linked because the same user can access to both resources (resource Y and X are linked because of *user6*); (2) two users are linked because they use the same resource (*user1* and *user6* are linked because they both have accounts on resource X); (3) two users are linked because they belong to the same VO (*user6* and *user4* are linked because they are members of Atlas VO); and (4) two resources are linked because they contribute to the same VO but there is no common user between them (resource Y and W are linked because they contribute to Atlas VO although they have no common users).

Based on these linkage types, we develop a simple grid model. Our model in its initial phase considers only the linkage type 1, resources with common users. The grid is modeled as an undirected graph of the network by $\mathcal{G} := (\mathcal{S}, \mathcal{E})$, where $\mathcal{E}$ is the set of edges $\{i, j\}$ for $i, j \in \mathcal{S}$, where $\mathcal{S}$ represents the set of grid resources. An edge $\{i, j\} \in \mathcal{E}$ exists if and only if there exists at least one common user between sites $i$ and $j$. Our grid model can be extended to represent all four linkage types; however, in this paper we chose not to so that we can discuss our risk model and optimization technique without the added complexity. Once we discuss how calculation of risk levels is affected by collaborations, we can introduce more complex grid models to account for different linkage types.

We implemented our grid model based on the data we received from the Open Science Grid. The OSG has 150 registered grid resources, approximately 8,000 users with 46 registered VOs. We chose to work with OSG because of its diverse set of VOs and the availability of monitoring and accounting services that allowed us to extract data to implement our grid model.

We obtained suitable type 1 connection data by using the OSG accounting service Gratia [1] that is run centrally by the OSG infrastructure to collect accounting summaries from each grid resource. An accounting report is compiled for each resource every week. The report lists the name of each user who has used the resource during the week and for how long. We queried the OSG repository for the week of December 15–21, 2008. For a resource $r \in \mathcal{R}$ where $\mathcal{R}$ is the set of grid resources, we consider $r_1$ linked to $r_2$ if $r_2$ and $r_1$ are both used by $u \in \mathcal{U}$, where $\mathcal{U}$ is the set of scientists in OSG. In our grid model, we represent resources as nodes and connect two resources with an edge if the resources are linked. We assign weights to edges representing the number of common users between $r_1$ and $r_2$. The weight of an edge is defined as the number of users $\mathcal{U}$ between $r_1$ and $r_2$ such that $u_1, u_2 \in \mathcal{U}$, where $u_1$ causes $r_1$ to be linked to $r_2$, and $u_2$ causes $r_1$ to be linked to $r_2$. The resulting grid model is shown in Figure 2. Note that we did not build our model for each grid resource. We aggregated resources at a site and treated them as a single node (when a user submitted a job to $r_1$ element of site 1 and $r_2$ element of site 2, we increased the weight of $W_{12}$ by one. If a user $u$ submitted a job to $r_1$ and $r_2$ both elements of site 1, this did not shown in our graph). For the rest of the paper, we keep referring to sites instead of resources.



(a) TOY

(b) OSG

Figure 2: A simplified collaboration graph of the open grid network. The left one is a toy problem, and right panel is from a real OSG problem. All the ellipses are the abstraction of grid sites, and the links between any two sites indicate the number of common users between those sites.

## 3 Estimating the Threat Levels

Risk is defined as Risk = Threat Level (probability) × Loss. In this section, we quantify the threat or threat levels and estimate the probability that a site in the open grid is compromised. The threat level clearly depends on the collaboration graph or model of the open grid network, and we assume that sites where an attack has been detected have threat levels equal to one. We start by defining the graph of the open grid network.

We let $\mathcal{S}$ be the set of all sites (nodes in our graph), and we assume that we are given a partition of $\mathcal{S}$ into compromised sites, $\mathcal{S}_c$, and uncompromised sites, $\mathcal{S}_u$, and $\mathcal{E}$ be the set of edges. The weight of an edge $\{i,j\} \in \mathcal{E}$ can be defined as the number of common users between site $i$ and $j$. To simplify the notation, we write

$$W_{ij} \;:=\; W_{\{i,j\}} \;:=\; \text{number of common users of site } i \text{ and } j, \qquad \forall \{i,j\} \in \mathcal{E}, \tag{3.1}$$

where we use the convention $W_{ij} = W_{ji}$. The load of a site $i \in \mathcal{S}$ is defined as the total number of users:

$$L_i \;:=\; \text{number of users of site } i. \tag{3.2}$$

We assume that the threat $t_i$ to site $i$ is proportional to the threat of a connected site times the proportion of the load from that site. In practice, a threat or compromise is not usually detected instantaneously. In Section 4, we will show how to estimate initial threat levels that we can then use in our optimal response model . We let $P_0 \in [0,1]$ be the probability that an attack spreads before we detect it. The threat at site $i$ can now be obtained by solving the following system:

$$t_i = 1 \qquad \forall i \in \mathcal{S}_c, \tag{3.3a}$$

$$t_i = P_0 \sum_{\{i,j\} \in \mathcal{E}} t_j \frac{W_{ij}}{L_j}, \qquad \forall i \in \mathcal{S}_u. \tag{3.3b}$$

Equations (3.3) show that the threat level depends on the collaboration graph of the grid. In the next section we will define an optimization problem that allows the security team of the grid to temporarily close some sites or links to mitigate the effect of an attack.

Next, we give a sufficient condition that ensures that the threat levels $(t_i)$ are between zero and one.

**Proposition 3.1** *Consider the linear system of equations (3.3) in $t_i$. Assume that*

$$\max_{i \in \mathcal{S}} \left\{ P_0 \sum_{\{i,j\} \in \mathcal{E}} t_j \frac{W_{ij}}{L_j} \right\} \;\le\; 1. \tag{3.4}$$

*It follows that $0 \le t_i \le 1$.*

**Proof.** We can interpret (3.3) as a fix-point iteration. Condition (3.4) ensures that the iteration matrix, $M$, corresponding to the right-hand-side of (3.3) is a contraction. Thus, it follows that if we initialize the fix-point iteration with $0 \le t_i^k \le 1$, then

$$\|t^{k+1}\| = \|Mt^k\| \le \|M\|\|t^k\| \le 1.$$

Moreover, the right-hand-side in the iteration is positive, so it follows that $t_i^{k+1} \geq 0$. The fix-point iteration converges, and we get $0 \leq t_i \leq 1$. □

# 4 Modeling Optimal Response to Network Attacks

Local security contacts can respond to an attack by shutting down or monitoring parts of a network. This response has two competing goals: first, we would like to reduce the threat to uncompromised sites as much as possible, and second, we would like to minimize the impact of the response on the remaining grid or maximizing the utility (Equation 4.9) of the grid. This gives rise to a multiobjective optimization problem. Next, we describe a model (Section 4.1) that allows the network administrator to shut down individual sites. The second model (Section 4.2) includes monitoring of links and considers shutting down certain links only.

Closing links between sites affects the threat level, and this will be our main mechanism to ensure that threat levels are below some maximum level, $T_{\max}$. As a consequence, the threat-levels will be variables in the next section.

## 4.1 Responding by Closing Sites

We model the closure of a site (all edges associated with it is closed) with a binary decision variable:

$$x_i = \begin{cases} 1 & \text{if site/node } i \in \mathcal{S} \text{ is open,} \\ 0 & \text{else} \end{cases}$$

and we define the utility of the network as the total number of users that can continue to use the network:

$$\text{utility} = \sum_{\{i,j\} \in \mathcal{E}} W_{ij} x_i x_j. \tag{4.1}$$

By closing site $j$, we affect the threat level $t_i$ at all sites $i$ that are connected to the closed site $j$. We assume that closing a site stops the spread of an attack from that site, and we eliminate the corresponding term in (3.3). Thus, we have

$$t_i = T_i^0 + P_1 \sum_{\{i,j\} \in \mathcal{E}} t_j x_j \frac{W_{ij}}{L_j}, \qquad \forall i \in \mathcal{S}_u, \tag{4.2}$$

where $0 \leq P_1 \leq 1$, is the probability that an attack spreads after we have detected it. Clearly, we require that $P_0 + P_1 \leq 1$, where $P_0$ is the probability that an attack spreads before we detect it; see (3.3). The parameters $T_i^0$ are the initial threat levels that model the fact that site $i$ may have been compromised before we had a chance to take corrective action. We can estimate values for $T_i^0$ by solving (3.3). We can also readily include endogenous factors such as the probability that a site is compromised by an unrelated attack in the values of $T_i^0$. Considering that only the threat level for those open sites is interesting to us, we can modify Equation 4.2 as

$$t_i = T_i^0 x_i + P_1 \sum_{\{i,j\} \in \mathcal{E}} t_j x_i x_j \frac{W_{ij}}{L_j}, \qquad \forall i \in \mathcal{S}_u. \tag{4.3}$$

Hence, the threat level for any closed sites is set explicitly to 0.

Now we have the definition of the objective (Equation 4.1) and the main threat level for each sites (Equation 4.3). Unfortunately, they both contain the bilinear terms $x_i x_j$, making it nonconvex. We can easily linearize this equation by introducing new variables $u_{ij}$ defined as

$$u_{ij} = x_i x_j, \quad \forall \{i, j\} \in \mathcal{S}.$$

We then exploit the binary domain of $x_i$ and $x_j$ to linearize this equation:

$$\begin{aligned} u_{ij} &\geq 0 && (4.4a)\\ u_{ij} &\leq x_i && (4.4b)\\ u_{ij} &\leq x_j && (4.4c)\\ u_{ij} &\geq x_i + x_j - 1, && (4.4d) \end{aligned}$$

which corresponds to the so-called Big-M formulation. In this case, however, we can choose $M = 1$, because all the variables are bounded by $1$.

The bilinear term $x_i x_j$ can be avoided by introducing new variable $u_{ij}$, but there is still a bilinear term $t_j u_{ij}$ in the constrain (Equation 4.3). Similarly, we can introduce another new variable

$$v_{ij} = u_{ij} t_j,$$

with a set of new constraints as follows:

$$\begin{aligned} 0 \leq \; & v_{ij} \leq u_{ij} && (4.5a)\\ t_i - (1 - u_{ij}) \leq \; & v_{ij} \leq t_i + (1 - u_{ij}). && (4.5b) \end{aligned}$$

Next, we define our second objective, which is to minimize the threat level. To avoid the solution of a multiobjective integer optimization problem (see, e.g., [14, 17]), we add a constraint on the maximum allowable threat level, $t_j \leq T_j, \ \forall j \in \mathcal{S}_u$. This approach corresponds to goal programming in multiobjective optimization and allows us to trace the Pareto set of solution of the multiobjective optimization problem. In addition, the set of constraints for $v_{ij}$, as in (4.5), can be tightened as follows:

$$\begin{aligned} 0 \leq \; & v_{ij} \leq u_{ij} T_j && (4.6a)\\ t_i - (1 - u_{ij}) T_j \leq \; & v_{ij} \leq t_i. && (4.6b) \end{aligned}$$

One immediate response to compromised sites is to shut them all down immediately. This obviously will reduce the usability of the network. Optimistically, we would keep up as many as possible sites even if they were compromised, with the criterion on that the threat for each uncompromised open site $i$ is still less than the threshold $T_i$. For those uncompromised sites $i \in \mathcal{S}_u$, there is a constraint $0 \leq t_i \leq T_i$ to control whether the sites should open or close. Before shutting down any uncompromised site, it should be better to shut down some compromised sites and check to see whether this action will save the uncompromised site. Mathematically, this situation can be formulated as

$$\begin{aligned} p\left(|\mathcal{S}_u| - |\mathcal{S}_c|\right) &\leq \sum_{i\in\mathcal{S}_u} x_i - \sum_{i\in\mathcal{S}_c} x_i && \text{(4.7a)}\\ \frac{\sum_{i\in\mathcal{S}_c} x_i}{|\mathcal{S}_c|} &\leq p \leq \sum_{i\in\mathcal{S}_c} x_i && \text{(4.7b)}\\ & p \in \{0,1\}. && \text{(4.7c)} \end{aligned}$$

Note that a new binary variable $p$ is introduced in order to implement this constraint.

Our first model is summarized as the following mixed-integer program (MIP):

$$\begin{aligned} \underset{u,v,x}{\text{maximize}} \quad & \sum_{\{i,j\}\in\mathcal{E}} W_{ij} u_{ij} && && \text{(4.8a)}\\ \text{subject to} \quad & 0 \leq t_i \leq T_i && \forall i \in \mathcal{S}_u && \text{(4.8b)}\\ & t_i = T_i^0 x_i + P_1 \sum_{\{i,j\}\in\mathcal{E}} v_{ij} \frac{W_{ij}}{L_j} && \forall i \in \mathcal{S}_u && \text{(4.8c)}\\ & t_i = 1 && \forall i \in \mathcal{S}_c && \text{(4.8d)}\\ & x_i + x_j - 1 \leq u_{ij} \leq x_i,\ u_{ij} \leq x_j && \forall \{i,j\} \in \mathcal{E} && \text{(4.8e)}\\ & 0 \leq v_{ij} \leq u_{ij} T_j && \forall \{i,j\} \in \mathcal{E} && \text{(4.8f)}\\ & t_i - (1-u_{ij}) T_j \leq v_{ij} \leq t_i && \forall \{i,j\} \in \mathcal{E} && \text{(4.8g)}\\ & 0 \leq u_{ij} \leq 1,\ 0 \leq v_{ij} \leq T_j && \forall \{i,j\} \in \mathcal{E} && \text{(4.8h)}\\ & p\left(|\mathcal{S}_u| - |\mathcal{S}_c|\right) \leq \sum_{i\in\mathcal{S}_u} x_i - \sum_{i\in\mathcal{S}_c} x_i && && \text{(4.8i)}\\ & \frac{\sum_{i\in\mathcal{S}_c} x_i}{|\mathcal{S}_c|} \leq p \leq \sum_{i\in\mathcal{S}_c} x_i && && \text{(4.8j)}\\ & p, x_i \in \{0,1\} && \forall i \in \mathcal{S}_u, && \text{(4.8k)} \end{aligned}$$

where $0 < P_1 \leq 1$, $0 \leq T_i^0 \leq 1$, $0 < T_i < 1$ $(\forall i \in \mathcal{S}_u)$, and $T_i = 1$ $(\forall i \in \mathcal{S}_c)$. We envisage solving this problem for a sequence of parameters $T_{\max}$ to trace out the Pareto curve of this multiobjective optimization problem, allowing network administrators to evaluate the trade-offs between security and usability.

## 4.2 Responding by Closing and Monitoring Links

A less intrusive approach to model (4.8) is to close or monitor only the links, because closing a site is equivalent to closing all the links to that site. In addition, we allow monitoring of traffic along an arc in the network. A downside of this model is that we introduce many more binary variables (two per link rather than one per node in the network).

We model the two actions (shutting down or monitoring a link) with two binary decision variables that are defined on each link $\{i,j\} \in \mathcal{E}$:

$$y_{ij} \;=\; \begin{cases} 1 & \text{if link } \{i,j\} \in \mathcal{E} \text{ is open} \\ 0 & \text{else} \end{cases}$$

$$z_{ij} \;=\; \begin{cases} 1 & \text{if link } \{i,j\} \in \mathcal{E} \text{ is monitored} \\ 0 & \text{else.} \end{cases}$$

Clearly, we can monitor only those links that are open. We can express this logical relationship between $y_{ij}$ and $z_{ij}$ as

$$z_{ij} \le y_{ij}, \quad \forall\, \{i,j\} \in \mathcal{E}.$$

In the case where both end points of a link are compromised sites, we strengthen this constraint to

$$z_{ij} = y_{ij}, \quad \forall\, \{i,j\} \in \mathcal{E} :\; i,j \in \mathcal{S}_c.$$

This constraint ensures that traffic between two compromised sites is either monitored or disconnected.

We also add an upper bound on the number of links that can be monitored, which becomes a resource constraint of the form

$$\sum_{\{i,j\} \in \mathcal{E}} z_{ij} \le K$$

for some integer parameter $K$.

The utility of the network is again defined as the total number of users who can continue to use the network, which now becomes

$$\text{utility} \;=\; \sum_{\{i,j\} \in \mathcal{E}} W_{ij} y_{ij}. \tag{4.9}$$

Next, we model the effect of closing or monitoring a connection on the threat level $t_i$. If we close a connection, then we stop the spread of an attack along that link, and we can eliminate the corresponding term in (3.3). If we monitor a connection, then we assume that the threat level is reduced by a constant factor $(1 - D)$ for some discount rate $0 < D < 1$. We can model these conditions with the binary variables as

$$t_i = T_i^0 + P_1 \sum_{\{i,j\} \in \mathcal{E}} t_j \left( y_{ij} - D z_{ij} \right) \frac{W_{ij}}{L_j}, \qquad \forall i \in \mathcal{S}_u.$$

This equation, however, contains bilinear terms $t_j y_{ij}$ and $t_j z_{ij}$, making it a nonconvex equation. This nonconvexity makes it harder to guarantee global optimality of optimization methods. Fortunately, we can reformulate these terms by introducing a new set of variables

$$r_{ij} = t_j \left( y_{ij} - D z_{ij} \right), \quad \forall \{i,j\} \in \mathcal{E}.$$

We can now exploit the binary domain of the variables $y_{ij}$ and $z_{ij}$ to derive the following equivalent linear formulation:

$$\begin{aligned} t_j - (1 - y_{ij} + z_{ij})\, T_j &\le r_{ij} \le \quad t_j && (4.10a)\\ (1-D)(t_j - (1 - z_{ij})\, T_j) &\le r_{ij} \le \quad (1-D)t_j + (1-z_{ij})D\, T_j && (4.10b)\\ 0 &\le r_{ij} \le \quad y_{ij}\, T_j. && (4.10c) \end{aligned}$$

As before, we model the second objective (minimize the threat level) as a constraint, and obtain the following MIP:

$$\begin{aligned} \underset{r,t,y,z}{\text{maximize}} \quad & \sum_{\{i,j\}\in\mathcal{E}} W_{ij} y_{ij} && && (4.11a)\\ \text{subject to} \quad & t_j - (1 - y_{ij} + z_{ij})\, T_j \le r_{ij} \le t_j \le T_j, && \forall \{i,j\} \in \mathcal{E} : j \in \mathcal{S}_u && (4.11b)\\ & (1-D)(t_j - (1 - z_{ij})\, T_j) \le r_{ij} && \forall \{i,j\} \in \mathcal{E} : j \in \mathcal{S}_u && (4.11c)\\ & r_{ij} \le (1-D)t_j + (1-z_{ij})D\, T_j && \forall \{i,j\} \in \mathcal{E} : j \in \mathcal{S}_u && (4.11d)\\ & 0 \le r_{ij} \le y_{ij}\, T_j && \forall \{i,j\} \in \mathcal{E} : j \in \mathcal{S}_u && (4.11e)\\ & 0 \le t_i = T_i^0 + P_1 \sum_{\{i,j\}\in\mathcal{E}} r_{ij} \frac{W_{ij}}{L_j}, && \forall i \in \mathcal{S}_u && (4.11f)\\ & t_i = 1 && \forall i \in \mathcal{S}_c && (4.11g)\\ & z_{ij} \le y_{ij} && \forall \{i,j\} \in \mathcal{E} && (4.11h)\\ & z_{ij} = y_{ij} && \forall \{i,j\} \in \mathcal{E} :\ i,j \in \mathcal{S}_c && (4.11i)\\ & \sum_{\{i,j\}\in\mathcal{E}} z_{ij} \le K && && (4.11j)\\ & y_{ij}, z_{ij} \in \{0,1\} && \forall \{i,j\} \in \mathcal{E}, && (4.11k) \end{aligned}$$

where $0 < P_1 \le 1$, $0 < D < 1$, $0 \le T_i^0 \le 1$, $0 < T_j \le 1$, and the integer $K > 0$ are parameters that model the way in which the threat is spread, and $W_{ij} \ge 0$ and $L_j$ are parameters defined in (3.1) and (3.2). We note that we can readily add constraints to keep certain important links open by setting the corresponding $y_{ij} = 1$. Similarly, we can add constraints to always monitor certain links by setting the corresponding $z_{ij} = 1$.

## 5 Numerical Experience

Along with the models discussed in Sections 3 and 4, we constructed eight data sets for our numerical experiments. The first data set is a toy problem, and the next data set is derived from usage data from the Open Science Grid for periods December 15–21, 2008. Note that we simplified "similar" sites into one to make the network much smaller. For each data set we choose 1, 2, 3, or 4 compromised sites. Table 2 shows the size for this instances, and Table 3 shows the parameters that are unchanged throughout.

Based on the model in Section 3, we estimated the initial threat levels for sites in the TOY and OSG problems. The results are shown in Tables 4 and 5.

Since the model to estimate the initial threat level for each site is basically solving a linear system, these estimation can be performed quickly. Tables 4 and 5 provide information on the site's initial threat; in reality, it may be much more complicated, but for simplification in testing

Table 1: Summary of symbols and notation used throughout the paper

| Symbol | Description |
|---|---|
| $\mathcal{S}$ | set of all sites |
| $\mathcal{S}_c$ | set of compromised sites |
| $\mathcal{S}_u$ | set of uncompromised sites |
| $\mathcal{S}_d$ | set of sites need to shut down |
| $i, j$ | site index |
| $L_i$ | load of site $i$ |
| $L_m$ | minimum load of sites among the whole set of sites |
| $L_M$ | maximum load of sites among the whole set of sites |
| $t_i$ | threat level for site $i$ |
| $T_i$ | maximum threat level for site $i$ |
| $T_{\text{max}}$ | uniform maximum threat level for sites |
| $T_i^0$ | initial threat level for site $i$ |
| $x_i$ | binary decision variable showing whether site $i$ is open or not |
| $\mathcal{E}$ | set of all edges |
| $\mathcal{E}_c$ | set of closed edges |
| $\mathcal{E}_m$ | set of monitored edges |
| $ij \equiv \{i, j\}$ | an edge |
| $W_{ij}$ | weight of an edge |
| $W_m$ | minimum weight of edges |
| $W_M$ | maximum weight of edges |
| $y_{ij}$ | binary decision variable showing where edge $\{i, j\}$ is open |
| $z_{ij}$ | binary decision variable showing where edge $\{i, j\}$ is monitored |
| $n_c$ | number of continuous variables |
| $n_b$ | number of binary variables |
| $n_m$ | number of constraint equations |
| $B$ | number of branch-and-bound nodes |
| $U_r$ | usability of the current network (the objective value) |
| $U_T$ | initial total usability of the network |
| $G$ | optimality gap between MIP and LP solution |
| $r_u$ | network usability ratio |
| $r_T$ | relative tolerance option in MIP |
| time | CPU time to run the model |
| $\lvert \cdot \rvert$ | number of elements of a set |

Table 2: Description of size of test problem instances

| Instance | $\lvert\mathcal{S}\rvert$ | $\lvert\mathcal{S}_c\rvert$ | $\lvert\mathcal{E}\rvert$ | $W_m$ | $W_M$ | $L_m$ | $L_M$ | $U_T$ |
|---|---|---|---|---|---|---|---|---|
| TOY-1 | 11 | 1 | 17 | 10 | 100 | 20 | 390 | 740 |
| TOY-2 | 11 | 2 | 17 | 10 | 100 | 20 | 390 | 740 |
| TOY-3 | 11 | 3 | 17 | 10 | 100 | 20 | 390 | 740 |
| TOY-4 | 11 | 4 | 17 | 10 | 100 | 20 | 390 | 740 |
| OSG-1 | 23 | 1 | 253 | 2 | 54 | 70 | 486 | 1490 |
| OSG-2 | 23 | 2 | 253 | 2 | 54 | 70 | 486 | 1490 |
| OSG-3 | 23 | 3 | 253 | 2 | 54 | 70 | 486 | 1490 |
| OSG-4 | 23 | 4 | 253 | 2 | 54 | 70 | 486 | 1490 |

Table 3: Common model parameters

| Name | Description | Value |
|---|---|---|
| $D$ | discount factor due to monitoring links | 0.90 |
| $K$ | upper bound on number of monitored links | 5 |
| $P_0$ | probability that attack spreads before it is detected | 0.25 |
| $P_1$ | probability that attack spreads after it is detected | 0.75 |
| $T_i(= T_{\max})$ | maximum threat level | 0.25 |

Table 4: Initial threat estimation for the TOY problem (as shown in Figure 2(a)) in the case of different number of compromised sites

| SITE | $\lvert\mathcal{S}_c\rvert = 1$ | $\lvert\mathcal{S}_c\rvert = 2$ | $\lvert\mathcal{S}_c\rvert = 3$ | $\lvert\mathcal{S}_c\rvert = 4$ |
|---|---|---|---|---|
| ANL | 0.0099 | 0.0679 | 0.0680 | 1.0000 |
| PNNL | 0.0003 | 0.0015 | 0.0018 | 0.0213 |
| LBNL | 0.0004 | 0.0007 | 0.0037 | 0.0056 |
| CERN | 0.1138 | 0.1521 | 1.0000 | 1.0000 |
| ORNL | 0.0001 | 0.0007 | 0.0008 | 0.0106 |
| FERMI | 0.1462 | 1.0000 | 1.0000 | 1.0000 |
| SNL | 0.0012 | 0.0079 | 0.0082 | 0.0277 |
| DESY | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| STANFORD | 0.0225 | 0.0499 | 0.0499 | 0.0499 |
| LLNL | 0.0022 | 0.0035 | 0.0194 | 0.0226 |
| CHICAGO | 0.0104 | 0.0712 | 0.0712 | 0.1683 |

Table 5: Initial threat estimation for the OSG problem (as shown in Figure 2(b)) in the case of different number of compromised sites

| SITE | $\|\mathcal{S}_c\| = 1$ | $\|\mathcal{S}_c\| = 2$ | $\|\mathcal{S}_c\| = 3$ | $\|\mathcal{S}_c\| = 4$ |
|---|---|---|---|---|
| NYSGRID | 0.0335 | 0.0558 | 0.0781 | 0.1045 |
| MIT | 0.0122 | 1.0000 | 1.0000 | 1.0000 |
| SBGrid | 0.0235 | 0.0452 | 0.0610 | 0.0772 |
| Nebraska | 0.0122 | 0.0278 | 0.0420 | 0.0503 |
| UCSDT2 | 0.0120 | 0.0231 | 0.0311 | 0.0458 |
| NERSC | 0.0112 | 0.0171 | 0.0246 | 0.0387 |
| USCMS | 0.0119 | 0.0229 | 0.0308 | 0.0390 |
| SPRACE | 0.0119 | 0.0229 | 0.0308 | 0.0390 |
| AGLT2 | 0.0122 | 0.0278 | 0.0420 | 0.0503 |
| UFlorida | 0.0220 | 0.0336 | 0.0482 | 1.0000 |
| Lehigh | 0.0119 | 0.0229 | 0.0308 | 0.0390 |
| RENCI | 0.0109 | 0.0166 | 0.0239 | 0.0315 |
| LIGO | 0.0119 | 0.0229 | 0.0308 | 0.0390 |
| FNAL | 1.0000 | 1.0000 | 1.0000 | 1.0000 |
| FLTECH | 0.0119 | 0.0229 | 0.0308 | 0.0390 |
| TTU | 0.0119 | 0.0229 | 0.0308 | 0.0390 |
| CIT | 0.0112 | 0.0216 | 1.0000 | 1.0000 |
| GLOW | 0.0122 | 0.0278 | 0.0420 | 0.0503 |
| SMU | 0.0119 | 0.0229 | 0.0308 | 0.0390 |
| isuhep | 0.0119 | 0.0229 | 0.0308 | 0.0390 |
| BNL | 0.0122 | 0.0278 | 0.0420 | 0.0503 |
| OSG | 0.0217 | 0.0330 | 0.0474 | 0.0624 |
| OCI | 0.0109 | 0.0166 | 0.0239 | 0.0315 |

the models, we set $T_i^0 = 0.1$.

We implemented models (4.8) and (4.11) in the modeling language GAMS [4] and used the MIP solver CPLEX [8] to solve the resulting problems. Our numerical experiments were conducted on a desktop Linux machine with Intel Core2 DUO CPU 6300 @ 1.86 GHz, 1 GB RAM. The data and model are available upon request.

In model (4.8) the number of binary variables equals the number of uncompromised sites (plus 1 for the additional variable $p$, see Equation 4.7), and the optimal solution can be reached quickly, as shown in Table 6. The results show that the network usability ratio ($r_u$) is low, and this may due to the action of shutting some sites down entirely in order to satisfy the maximum threat level criterion. The next model (4.11) improves the usability ratio, as shown in Table 7. We observe that we can solve these models in a few seconds, making it feasible to deploy our approach in practice.

Table 6: Modeling results by shutting down sites only for TOY and OSG problem instances

| | MIP Data | | | | | Network Data | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | $r_T$ | $n_c$ | $n_b$ | $n_m$ | $B$ | $\|\mathcal{S}_d\|$ | $\|\mathcal{E}_c\|$ | $U_r$ | $G$ | $r_u$ | time [s] |
| TOY-1 | 0 | 63 | 12 | 177 | 66 | 7 | 14 | 290 | 0 | 0.39 | 0 |
| TOY-2 | 0 | 63 | 12 | 175 | 16 | 5 | 13 | 140 | 0 | 0.19 | 0 |
| TOY-3 | 0 | 63 | 12 | 173 | 56 | 6 | 14 | 130 | 0 | 0.18 | 0 |
| TOY-4 | 0 | 63 | 12 | 171 | 0 | 4 | 9 | 130 | 0 | 0.18 | 0 |
| OSG-1 | 0 | 783 | 24 | 2325 | 82 | 3 | 63 | 678 | 0 | 0.46 | 2 |
| OSG-2 | 0 | 783 | 24 | 2323 | 17 | 4 | 82 | 600 | 0 | 0.40 | 1 |
| OSG-3 | 0 | 783 | 24 | 2321 | 490 | 8 | 148 | 566 | 0 | 0.38 | 4 |
| OSG-4 | 0 | 783 | 24 | 2319 | 39 | 6 | 117 | 696 | 0 | 0.47 | 2 |

In addition, the network changes for the TOY problem are shown in Figure 3. The figure shows only the case with four compromised sites. As can be seen, the network almost breaks down when four sites are compromised initially. In addition, we observe that shutting down individual links (Figure 3(b)) is clearly less intrusive than closing down sites (Figure 3(a)). In particular, the utility for model (4.11) is higher than the utility for model (4.8).

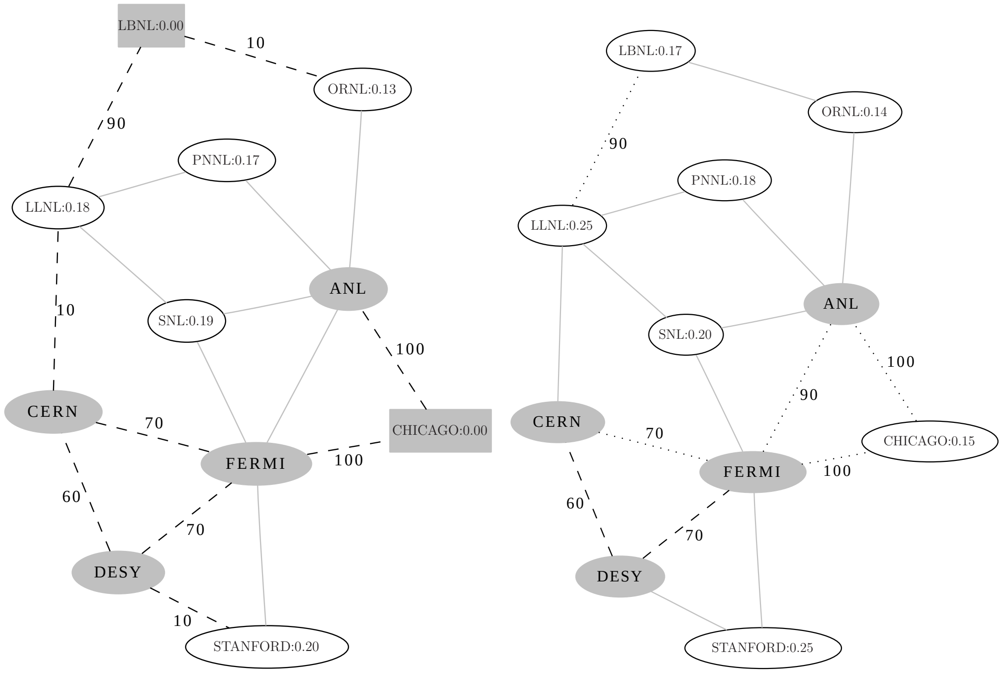
Initially, we ran CPLEX for model (4.11) with a tight relative tolerance of $10^{-3}$; that is, CPLEX fathomed nodes whenever they were integer feasible or infeasible or if

$$\frac{|\text{upper bound} - \text{LP solution}|}{|\text{upper bound}|} \leq \text{relative tolerance} = 10^{-3}.$$

Unfortunately, these runs failed because of the large memory demand and long execution time even for the small test problem as OSG. This situation will obviously limit the application of the model. However, by increasing the relative tolerance to $10^{-2}$ we can solve all models within a few of CPU seconds. The results are shown in Table 7.

We note that the optimality gaps ($G$) in Table 7 are not zero for the OSG instances because of the loose relative tolerance. However, the gap is not that big at large relative tolerance, and thus, we can obtain good solutions at a small computational cost, making our approach feasible for the management of real grids. We suspect that the large increase in computing time arises because there are many similar solutions due to symmetry.

Similar to Figure 3, Figure 4 shows the solution of the OSG problem for models (4.8) and (4.11). The figure shows the link changing if we close or monitor a link. Different from Figure 4(a) for



(a) TOY-4:S (b) TOY-4:L

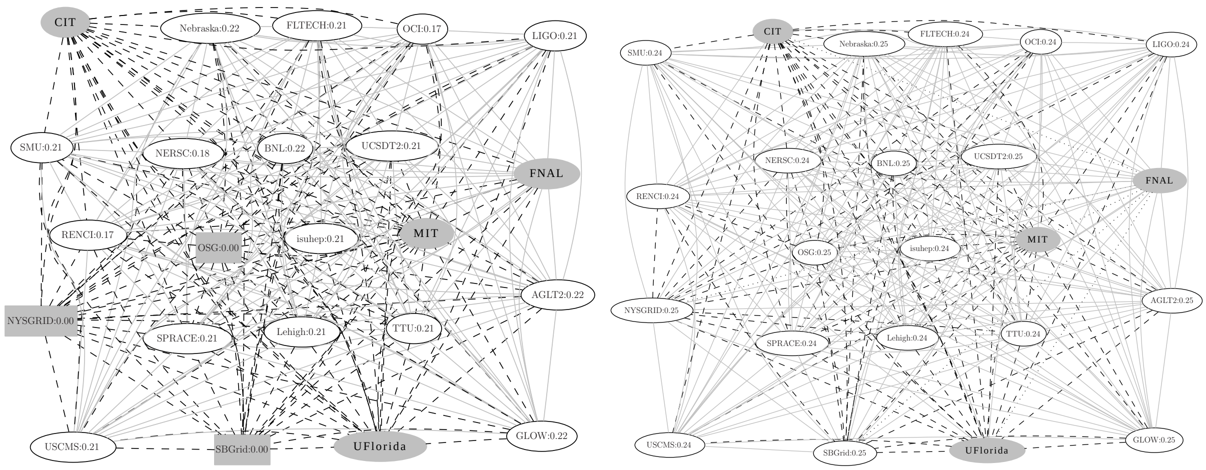
Figure 3: Toy network at $T_i^0 = 0.1$ and $T_{\max} = 0.25$. The grey ellipses are the compromised sites, and the grey boxes are the sites needed to be shut down. The dashed lines are the links needed to be cut, and the light grey lines are the original edges left untouched. The number adjacent to the site name is the final threat level of that site. The number associated with each cut edge indicates the weight of that edge. (a) is for the model 4.8 corresponding to shut down sites only and (b) is for the model 4.11 corresponding to disconnect and monitor edges.

Table 7: Modeling results by closing and monitoring links for TOY and OSG problem instances at different (MIP) relative tolerance levels

| | MIP Data | | | | | Network Data | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Name | $r_T$ | $n_c$ | $n_b$ | $n_m$ | $B$ | $\|\mathcal{E}_m\|$ | $\|\mathcal{E}_c\|$ | $U_r$ | $G$ | $r_u$ | time [s] |
| TOY-1 | 0 | 43 | 34 | 194 | 18 | 5 | 3 | 660 | 0.00 | 0.89 | 0 |
| TOY-2 | 0 | 37 | 34 | 163 | 0 | 5 | 2 | 670 | 0.00 | 0.91 | 0 |
| TOY-3 | 0 | 34 | 34 | 148 | 0 | 5 | 3 | 650 | 0.00 | 0.88 | 0 |
| TOY-4 | 0 | 29 | 34 | 122 | 0 | 5 | 2 | 610 | 0.00 | 0.82 | 0 |
| OSG-1 | 0.010 | 508 | 506 | 2719 | 190 | 5 | 33 | 1276 | 11.37 | 0.86 | 2 |
| OSG-2 | 0.010 | 486 | 506 | 2608 | 217 | 5 | 45 | 1226 | 11.62 | 0.82 | 2 |
| OSG-3 | 0.010 | 464 | 506 | 2498 | 115 | 5 | 59 | 1182 | 11.47 | 0.79 | 1 |
| OSG-4 | 0.010 | 442 | 506 | 2389 | 225 | 5 | 71 | 1132 | 10.93 | 0.76 | 2 |
| OSG-1 | 0.005 | 508 | 506 | 2719 | 5669991 | 5 | 33 | 1278 | 6.39 | 0.86 | 12831 |
| OSG-2 | 0.005 | 486 | 506 | 2608 | 20060 | 5 | 45 | 1230 | 5.40 | 0.83 | 52 |
| OSG-3 | 0.005 | 464 | 506 | 2498 | 1918653 | 5 | 61 | 1186 | 4.59 | 0.80 | 3568 |
| OSG-4 | 0.005 | 442 | 506 | 2389 | 862202 | 5 | 69 | 1136 | 4.83 | 0.76 | 1635 |

model (4.8), however, no site is closed (a site closed in model (4.11) means all the links associate with that site are disconnected). Along with the great discount due to the "monitoring a link", the network usability ratios as shown in Table 7 are much higher than those for model (4.8).

Figure 3(a) and 4(a) show the response model when we close sites, and Figure 3(b) and 4(b) show the result when we close or monitor the links only. Clearly, the latter approach is less intrusive and provides a higher utility for the same threat level. This observation should not come as a surprise because we can interpret (4.11) as a relaxation of (4.8).



(a) OSG-4:S

(b) OSG-4:L

Figure 4: OSG network at $T_i^0 = 0.1$ and $T_{\max} = 0.25$. The grey ellipses are the compromised sites. The dashed lines are the links need to cut, the dotted links are under monitoring, and the light grey lines are the original edges left untouched. The number adjacent to the site name is the final threat level of that site. The number associated with each cut or monitored edge indicates the weight of that edge. (a) is for the model 4.8 corresponding to shut down sites only and (b) is for the model 4.11 corresponding to disconnect and monitor edges.

## 5.1 Exploring the Pareto Surface of the Model

We noted above that our model is in fact a multiobjective MIP, that simultaneously minimizes the threat to uncompromised sites and maximizes the utility of the network. Unfortunately, no convenient solver exists for multiobjective MIPs. Instead, we explore the Pareto surface by repeatedly solving test OSG problem over a range of values of the maximum allowable threat level ($T_{\max}$). Figure 5 shows an approximation of the Pareto surface.

We observe that the Pareto curve in Figure 5 is not convex. The reason for this nonconvexity is that the MIP itself is not convex, and we typically expect a piecewise constant Pareto curve (as $T_{\max}$ is increased, we add links at discrete increments). In reality the Pareto curve is piecewise constant, but it is not practical to find the breakpoints in $T_{\max}$.

We envisage our model being used iteratively. The system operators would solve the MIP (4.11) for a small number of values of $T_{\max}$ to explore the trade-offs between network utility and security, possibly refining the resolution of $T_{\max}$.

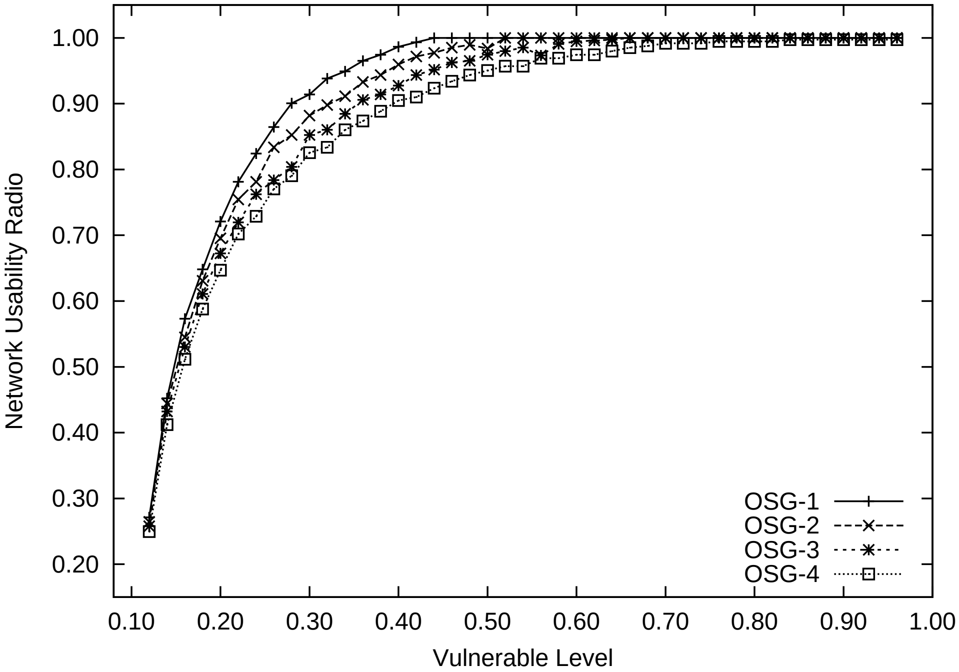


Figure 5: Pareto curve of OSG problem with 1-4 compromised sites. We show the network utility ratio as a function of increasing maximum allowable threat, $T_{\max}$. The relative tolerance is set to 0.05.

## 6 Extensions and Conclusions

We have presented two models that allow us to design an optimal response to an attack on a grid. The two models respond by shutting down parts of the network to limit threat levels in the remaining network and maximize the utility of the remaining network. We have shown that the models can be formulated as mixed-integer linear optimization problems, and we have solved a number of problems arising out of real Open Science Grid data.

Our preliminary numerical results are encouraging. We are able to solve a realistic model within a few seconds. Our results are intuitive in the sense that increasing the number of compromised sites increases the number of sites or links that must be shut down to maintain a given threat level at uncompromised sites. We observe that in terms of the final utility of the remaining network, closing or monitoring links is preferable to closing down sites.

We envisage using our model as a decision support tool for grid administrators. In particular, our model allows us to examine the trade-offs between threat levels and level of service, or utility. Since solution times are moderate, we can in principle trace a complete Pareto curve.

In practice, the threat levels, $T_i^0$ may not be known exactly, but rather contain some uncertainty that should be taken into account in our optimal response. In this case, the $T_i^0$ could either be given as distributions or as ranges. Taking into account uncertain threat levels requires us to solve either a stochastic MIP or a robust counterpart of the model we have developed. We are developing techniques to solve these extensions efficiently.

We can extend our model to design monitoring patterns that depend on actual usage. The goal would be to find the optimum configuration of links that, if monitored, increase the likelihood that we will detect unusual patterns or an attack on the grid.

Another extension arises if we reverse the focus of the model and ask ourselves where we should attack the open grid to cause maximum disruption (reducing utility as much as possible). To find the site that causes maximum disruption, we can solve the MIP (4.11) and set every site as compromised in turn. This would be a likely point of attack for an intelligent attacker, and we can use this knowledge to devise monitoring strategies or harden the most disruptive site against attack.

Formally, the problem of finding the $K > 1$ compromised sites that cause maximum havoc is a bilevel optimization problem, where the upper level minimizes utility subject to an optimal response at the lower level. The lower level problem, however, is a MIP, and there exist no close form optimality conditions for MIPs (except in some special cases). Hence, we cannot apply the usual approach to the bilevel problem and solve an optimization problem with complementarity constraints (e.g. [12, 10]). Unfortunately, this problem is largely intractable at present.

## Acknowledgments

This work was supported by the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under Contract No. DE-AC02-06CH11357.

## References

[1] Gratia. https://twiki.grid.iu.edu/bin/view/Accounting/WebHome.

[2] R. Alfieri, R. Cecchini, V. Ciaschini, L. dell'Agnello, Á. Frohner, K. Lorentey, and F. Spataro. From gridmap-file to VOMS: managing authorization in a grid environment. *Future Generation Computer Systems*, 21(4):549 – 558, 2005. High-Speed Networks and Services for Data-Intensive Grids: the DataTAG Project.

[3] Vladimir Batagelj and Andrej Mrvar. Some analyses of erdös collaboration graph. *Social Networks*, 22(2):173–186, 2000.

[4] A. Brooke, D. Kendrick, A. Meeraus, and R. Raman. *GAMS A user's guide*. GAMS Developments Corporation, 1217 Potomac Street, N.W., Washington DC 20007, USA, December 1998.

[5] R. Butler, V. Welch, D. Engert, I. Foster, S. Tuecke, J. Volmer, and C. Kesselman. A national-scale authentication infrastructure. *Computer*, 33(12):60–66, Dec 2000.

[6] Earth System Grid, http://www.earthsystemgrid.org.

[7] Ian T. Foster. Globus toolkit version 4: Software for service-oriented systems. In Hai Jin, Daniel A. Reed, and Wenbin Jiang, editors, *NPC*, volume 3779 of *Lecture Notes in Computer Science*, pages 2–13. Springer, 2005.

[8] GAMS Corporation. *CPLEX 11*, 2008.

[9] Casper Goffman. And what is your erdös number? *American Mathematical Monthly*, 76:791, 1979.

[10] S. Leyffer. Complementarity constraints as nonlinear equations: Theory and numerical experience. In S. Dempe and V. Kalashnikov, editors, *Optimization and Multivalued Mappings*, pages 169–208. Springer, Berlin, 2006.

[11] M. Lorch, D. Kafura, I. Fisk, K. Keahey, G. Carcassi, T. Freeman, T. Peremutov, and A.S. Rana. Authorization and account management in the open science grid. In *Grid Computing, 2005. The 6th IEEE/ACM International Workshop on*, pages 8 pp.–, Nov. 2005.

[12] Z.-Q. Luo, J.-S. Pang, and D. Ralph. *Mathematical Programs with Equilibrium Constraints*. Cambridge University Press, Cambridge, UK, 1996.

[13] Cooper et al. Internet x.509 public key infrastructure certificate and certificate revocation list. http://tools.ietf.org/html/rfc5280.

[14] K. Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, Boston, 1999.

[15] Open Science Grid, http://www.opensciencegrid.org.

[16] L. Pearlman, V. Welch, I. Foster, C. Kesselman, and S. Tuecke. A community authorization service for group collaboration. In *Policies for Distributed Systems and Networks, 2002. Proceedings. Third International Workshop on*, pages 50–59, 2002.

[17] R. E. Steuer. *Multiple Criteria Optimization: Theory, Computation and Applications*. John Wiley & Sons, New York, 1986.

[18] TeraGrid, http://www.teragrid.org/.