A Review of Several Optimization Problems Related to Security in Networked System

Bhaskar DasGupta and Venkatkumar Srinivasan

Abstract Security issues are becoming more and more important to activities of individuals, organizations and the society in our modern networked computerized world. In this chapter we survey a few optimization frameworks for problems related to security of various networked system such as the internet or the power grid system.

1 Introduction

Security issues are essential to activities of individuals, organizations and the society as a whole in our modern networked computerized world in healthcare, power management, online purchase, banking, intra-business transactions and many other similar activities in distributed-computing settings. A typical activity in such a networked system involves a set of (digital) transactions between various components ("agents") of the system to perform a specific task such as online purchase of an item or submitting an online application for a job, and requires interaction with various computer servers/databases and encryption services. Any compromise of these activities due to other malicious agents within the system or outside may lead to severe consequences such as disruption of critical infrastructure or national economy, and thus making sure that these activities are secure against such attacks is of paramount importance. The significance of maintaining security of networked systems has in fact led to organization of many security competitions, such as the

Bhaskar DasGupta
Department of Computer Science, University of Illinois at Chicago, Chicago, IL 50507, USA, e-mail: bdasgup@uic.edu

Venkatkumar Srinivasan
Department of Computer Science, University of Illinois at Chicago, Chicago, IL 50507, USA, e-mail: vsrini7@uic.edu

international Capture The Flag [24] for researchers to discuss, discover and validate new solutions for security issues.

In this chapter, we survey several optimization problems related to the security issues in distributed networked systems. We start by surveying some of the grand mathematical challenges in developing and analyzing security of real-world networked systems in Section 2. Then, in the remaining sections we survey several optimization problems related to maintaining and evaluating securities of such systems.

2 Mathematical and Statistical Challenges In Networked Security

Some of the major mathematical and statistical challenges for security issues in networked systems are discussed in the two white papers [7, 18]. Based on these and other white papers, at least the following four possible challenge areas can be identified:

Data acquisition: The challenge here is to generate accurate trace and log data while maintaining their integrity throughout the lifetime of their intended use for scientific analysis and verification since lack of public data sets is a significant barrier in current research [19]. This challenge is also related to the so-called "utility versus privacy trade-off" issue [21] since making data publicly available may pose confidentiality and privacy issues.

Modelling networks: This challenge refers to the difficulties in developing mathematical network models that accurately model real-world networks and statistical methods for comparing networks (*e.g.*, see [9]). For example, a typical question could be whether the distribution of degrees of nodes over the entire network is governed by power-laws or its variants?

Detection and response to security threats: This challenge refers to the difficulty in formulating and solving problems such as malicious code or behavior detection that provide long-term proactive approach to network security. Research methods for overcoming this challenge may involve techniques from diverse areas of mathematics or computer science such as dynamic data modelling methods, optimization methods, machine learning methods and methods for uncertainty modelling via probabilistic models.

Modelling network dynamics: This challenge refers to developing appropriate mathematical models to understand the mechanism of spread of infections (*i.e.*, time evolution of malicious attacks) in networks. Ideas from game theory or dynamical systems may be particularly useful in this context.

3 Application Of Convex Optimization In Network Security

In this section, we review an application of convex or concave optimization methods by Vamvoudakis *et al.* [23] to model the complex behavior of a malicious attacker in a networked system. The model was incorporated in a cyber security advisory system to demonstrate its effectiveness. The optimization problem is formulated *from the point of view of a malicious attacker, i.e.*, the goal is to find an optimal allocation of available resources for an attacker to maximize the potential damage.

We start by specifying a *model* of the damage caused by a (malicious) attacker of the given network. In the model, $t \in \{1, 2, \ldots, T\}$ indicates the discrete time variable. Assume that there are a set of S services, indexed by $1, 2, \ldots, S$, in our networked system that may be attacked for disruption. The following parameters are used in the model:

$u^s_{\mathrm{AR}_t} \geq 0$: A scalar quantifying the *amount of attack resources* (*e.g.*, amount of money devoted to attack a particular resource) used by the attacker to attack service s at time t.

$x^s_{\mathrm{PD}_t} \geq 0$: A scalar denoting the *amount of potential damage* caused by attacks. In general $x^s_{\mathrm{PD}_t} = f^s_t\left(u^s_{\mathrm{AR}_t}\right)$ for some appropriate function $f^s_t : \mathbb{R}^+ \mapsto \mathbb{R}^+$.

g^s_t: $g^s_t\left(u^s_{\mathrm{AR}_t}\right)$ denotes the probability that the damage $f^s_t\left(u^s_{\mathrm{AR}_t}\right)$ is realized as a result of the attack $x^s_{\mathrm{PD}_t}$.

$y^s_{\mathrm{TD}_t}$: $y^s_{\mathrm{TD}_t} = g^s_t\left(u^s_{\mathrm{AR}_t}\right) f^s_t\left(u^s_{\mathrm{AR}_t}\right)$ is the *expected damage* caused by $u^s_{\mathrm{AR}_t}$.

y_{TD}: $y_{\mathrm{TD}} = \sum_{t=1}^{T} \sum_{s=1}^{S} y^s_{\mathrm{TD}_t}$ is the total expected damage.

U_{TR}: This is the total budget of attack resources available to the attacker.

In order to ensure that the resulting optimization problems are convex or concave, Vamvoudakis *et al.* [23] makes the following assumptions that are justified for real applications of the model:

- f^s_t is a *linear* function, *i.e.*,

$$f^s_t\left(u^s_{\mathrm{AR}_t}\right) = a^s_t + b^s_t u^s_{\mathrm{AR}_t} \tag{1}$$

  for some constants $a^s_t, b^s_t \in \mathbb{R}^+$. The constant a^s_t models the extent of damage without any attack whereas the constant b^s_t models the extent of damage per unit of attack resources employed. The equation has the realistic implication that an increase in attack resources leads to an increase in the potential damage caused.

- g^s_t is a *linearly increasing* function projected to the interval $[0, 1]$, *i.e.*,

$$g^s_t\left(u^s_{\mathrm{AR}_t}\right) = \begin{cases} 0, & \text{if } d^s_t u^s_{\mathrm{AR}_t} > c^s_t \\ c^s_t - d^s_t u^s_{\mathrm{AR}_t}, & \text{if } c^s_t - 1 \leq d^s_t u^s_{\mathrm{AR}_t} \leq c^s_t \\ 1, & \text{if } d^s_t u^s_{\mathrm{AR}_t} < c^s_t - 1 \end{cases} \tag{2}$$

for some given constants $c_t^s, d_t^s \geq 0$. The constant c_t^s models the probability of damage realization without any attack whereas the constant d_t^s models the decrease of the probability of damage realization per unit of attack resources employed. Note that this choice of g_t^s models the realistic assumption that an increase in attack resources decreases the realization probability of the potential damage since a large-scale attack is much more likely to trigger defense mechanisms.

Now we can consider two different optimization problems for optimal allocation of available resources by an attacker to maximize the potential damage depending on the availability of relevant data.

Optimization problem when all relevant damage data is known When all the relevant damage data, *i.e.*, all the numbers in $\left\{a_t^s, b_t^s, c_t^s, d_t^s \mid 1 \leq s \leq S, 1 \leq t \leq T\right\}$, are known *a-priori*, it is easy to see that the optimal attack resource allocation values (*i.e.*, the $u_{\mathrm{AR}_t}^s$'s) that maximizes the total expected damage y_{TD} can be obtained by solving the following constrained optimization problem:

$$\begin{array}{lll} \textit{maximize} & y_{\mathrm{TD}} & \\ \textit{subject to} & \sum_{t=1}^{T}\sum_{s=1}^{S} u_{\mathrm{AR}_t}^s \leq U_{\mathrm{TR}} & \\ & u_{\mathrm{AR}_t}^s \geq 0, & 1 \leq s \leq S,\ 1 \leq t \leq T \end{array} \tag{3}$$

Although in general (3) may be difficult to solve, Vamvoudakis *et al.* [23] show that the special choices of f_t^s in (1) and g_t^s in (2) ensure that the above optimization problem is *equivalent* to solving the following *concave maximization* (or, equivalently convex minimization)[1] problem with linear constraints involving an addition set of z_t^s variables:

$$\begin{array}{lll} \textit{maximize} & \sum_{t=1}^{T}\sum_{s=1}^{S} \left(a_t^s + b_t^s u_{\mathrm{AR}_t}^s\right)\left(c_t^s - d_t^s u_{\mathrm{AR}_t}^s - z_t^s\right) & \\ \textit{subject to} & \sum_{t=1}^{T}\sum_{s=1}^{S} u_{\mathrm{AR}_t}^s \leq U_{\mathrm{TR}} & \\ & c_t^s - d_t^s u_{\mathrm{AR}_t}^s - z_t^s \leq 1, & 1 \leq s \leq S,\ 1 \leq t \leq T \\ & u_{\mathrm{AR}_t}^s \geq 0, & 1 \leq s \leq S,\ 1 \leq t \leq T \end{array} \tag{4}$$

and, moreover, if $0 \leq c_t^s \leq 1$ for all s and t then one can set $z_t^s = 0$ for all s and t in the above concave optimization problem.

[1] A function h of k variables is convex (resp. concave) if and only if, for all $x_1, x_2, \ldots, x_k, y_1, y_2, \ldots, y_k$ and for all $0 < \lambda < 1$, $h((1-\lambda)(x_1, x_2, \ldots, x_k) + \lambda(y_1, y_2, \ldots, y_k)) \geq (1-\lambda)h(x_1, x_2, \ldots, x_k) + \lambda h(y_1, y_2, \ldots, y_k)$ (resp. $h((1-\lambda)(x_1, x_2, \ldots, x_k) + \lambda(y_1, y_2, \ldots, y_k)) \leq (1-\lambda)h(x_1, x_2, \ldots, x_k) + \lambda h(y_1, y_2, \ldots, y_k)$). When the objective function and all the constraints are convex (resp. concave), we have a convex (resp. concave) optimization problem. The convexity or concavity property often makes an optimization problem easier to solve as opposed to the general case; see [3] for further details.

Optimization problem when not all relevant damage data is known Often the parameter values in $\left\{a_t^s, b_t^s, c_t^s, d_t^s \mid 1 \le s \le S, 1 \le t \le T\right\}$ are *not* known *a-priori*. In that case, one needs to estimate these parameter values online based on past observations using some *machine learning* techniques such as the maximum-likelihood approach. Vamvoudakis *et al.* [23] propose the following approach for the parameter estimation problem:

- Assume that these parameters are generated by a *linear* dynamical system over time *t, i.e.*,

$$a_t^s = C_a^s x_a^s(t) \text{ where } x_a^s(t) \text{ is generated by } x_a^s(t) = A_a^s x_a^s(t-1) + B_a^s w^s(t-1) \quad (5)$$
$$b_t^s = C_b^s x_b^s(t) \text{ where } x_b^s(t) \text{ is generated by } x_b^s(t) = A_b^s x_b^s(t-1) + B_b^s w^s(t-1) \quad (6)$$
$$c_t^s = C_c^s x_c^s(t) \text{ where } x_c^s(t) \text{ is generated by } x_c^s(t) = A_c^s x_c^s(t-1) + B_c^s w^s(t-1) \quad (7)$$
$$d_t^s = C_d^s x_d^s(t) \text{ where } x_d^s(t) \text{ is generated by } x_d^s(t) = A_d^s x_d^s(t-1) + B_d^s w^s(t-1) \quad (8)$$

  where $\left\{A_j^s, B_j^s, C_j^s, D_j^s \mid 1 \le s \le S, j \in \{a,b,c,d\}\right\}$ are scalar parameters of the dynamics, and w_t^s are sequences generated by a random process with zero mean and variance z_t^s. Use historical data to estimate these dynamics using blackbox identification techniques.

- Now use online data to estimate the values of $\left\{a_t^s, b_t^s, c_t^s, d_t^s \mid 1 \le s \le S, 1 \le t \le T\right\}$ based on past observations using a k-step ahead predictor in the following manner. Let $\left\{a_t^s, b_t^s, c_t^s, d_t^s \mid 1 \le s \le S, 1 \le t \le k < T\right\}$ be the set of values observed (by the attacker) for these parameters up to some time $k < T$ and the attacker needs to compute the "future" values of $u_{\mathrm{AR}_t}^s$'s for $k < t \le T$. Then, one can do the following.

  – Estimate the values of $\left\{a_t^s, b_t^s, c_t^s, d_t^s \mid 1 \le s \le S, k < t \le T\right\}$ using (5)–(8). Let the estimated values for $a_t^s, b_t^s, c_t^s, d_t^s$ be denoted by $\widehat{a_t^s}, \widehat{b_t^s}, \widehat{c_t^s}, \widehat{d_t^s}$. Let $\widehat{f_t^s}$ and $\widehat{g_t^s}$ be the function values of f_t^s and g_t^s, respectively, for $k < t \le T$ when the estimated values $\widehat{a_t^s}, \widehat{b_t^s}, \widehat{c_t^s}, \widehat{d_t^s}$ are used, *i.e.*,

$$\widehat{f_t^s}\left(u_{\mathrm{AR}_t}^s\right) = \widehat{a_t^s} + \widehat{b_t^s} u_{\mathrm{AR}_t}^s \text{ and } \widehat{g_t^s}\left(u_{\mathrm{AR}_t}^s\right) = \begin{cases} 0, & \text{if } \widehat{d_t^s} u_{\mathrm{AR}_t}^s > \widehat{c_t^s} \\ \widehat{c_t^s} - \widehat{d_t^s} u_{\mathrm{AR}_t}^s, & \text{if } \widehat{c_t^s} - 1 \le \widehat{d_t^s} u_{\mathrm{AR}_t}^s \le \widehat{c_t^s} \\ 1, & \text{if } \widehat{d_t^s} u_{\mathrm{AR}_t}^s < \widehat{c_t^s} - 1 \end{cases}.$$

  – Compute $u_{\mathrm{AR}_t}^s$ for $k < t \le T$ by solving the following optimization problem:

$$\begin{aligned} \textit{maximize} \quad & \sum_{t=1}^{k} \sum_{s=1}^{S} g_t^s\left(u_{\mathrm{AR}_t}^s\right) f_t^s\left(u_{\mathrm{AR}_t}^s\right) + \sum_{t=k+1}^{T} \sum_{s=1}^{S} \widehat{g_t^s}\left(u_{\mathrm{AR}_t}^s\right) \widehat{f_t^s}\left(u_{\mathrm{AR}_t}^s\right) \\ \textit{subject to} \quad & \sum_{t=1}^{T} \sum_{s=1}^{S} u_{\mathrm{AR}_t}^s \le U_{\mathrm{TR}} \\ & u_{\mathrm{AR}_t}^s \ge 0, \quad 1 \le s \le S, k \le t \le T \end{aligned}$$

  which is again a convex minimization problem similar to (4).

The k-step lookahead predictor can be used in an online fashion by the attacker for every successive value of k.

4 Application Of Multi Objective Distributed Constraint Optimization In Network Security

In the previous section we saw how to formulate and solve some problems related to the security of networked systems as a convex constraint optimization problem with a *single* objective function. In this section, we review the results of Okimoto *et al.* [17] that apply *multi-objective distributed* constraint optimization methods to formulate and solve problems related to security issues of networked systems. Okimoto *et al.* [17] do this by first formulating the security problem for networked system as a multi-objective distributed constraint optimization problem (Mo-DCOP) using the formalization in [8], and then discussing some algorithmic approaches to solve such an optimization problem. Generally, multi-objective distributed constraint optimization methods are very suitable for formalizing applications related to multi-agent cooperation. An advantage of casting network security problems as a Mo-DCOP is that multiple criteria (*e.g.*, level of risk, loss of privacy, cost of operation) can be optimized *simultaneously* instead of separately; however, a disadvantage of this is that the resulting optimization problem may be computationally quite hard. The multi-objective distributed constraint optimization framework of [8] is an extension of *mono-objective* distributed constraint optimization framework in [15] for modelling applications related to multi-agent cooperation games.

The Mo-DCOP proposed by Okimoto *et al.* [17] is described by the following parameters:

- A 5-tuple $\langle \mathcal{S}, \mathcal{X}, \mathcal{D}, C, O \rangle$ where
  - $\mathcal{S} = \{\mathsf{agent}_1, \mathsf{agent}_2, \ldots, \mathsf{agent}_n\}$ is a set of n agents. An agent may be a human, a program, an organization, a country *etc*.
  - $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ is a set of n variables, where x_i is owned by agent_i.
  - $\mathcal{D} = \{D_1, D_2 \ldots, D_n\}$ is a set of n discrete domains, where D_i is the domain of values of variable x_i. The notation (x_i, d_i) will be used to denote an assignment of value $d_i \in D_i$ to variable x_i.
  - $O = \left\{O^1, O^2, \ldots, O^m\right\}$ is a set of m criteria that is to be optimized.
  - $C = \left\{C^1, C^2, \ldots, C^m\right\}$ is a set of m sets of constraints, where C^ℓ is the set of constraints corresponding to the ℓ^{th} criterion O^ℓ. A constraint relation $\left(\bowtie_{i,k}, \bowtie_{j,k}\right)^\ell \in C^\ell$ (for $k = 1, 2, \ldots$) denotes a constraint of the type $\left\{(x_i, d_i), \left(x_j, d_j\right)\right\}$ involving the variables x_i and x_j, and is used to describe the condition of cooperation of agent_i and agent_j on the objective C^ℓ.

- $\left\{ f_{i,j,k}^{\ell} : D_i \times D_j \mapsto \mathbb{R} \;\middle|\; 1 \le \ell \le m, 1 \le i, j \le n, k \in \mathbb{N}^+ \right\}$ is a given set of cost functions such that $f_{i,j,k}^{\ell}\left(d_i, d_j\right)$ gives, for each objective C^{ℓ} and pairs x_i, x_j such that $\left(\bowtie_{i,k}, \bowtie_{j,k}\right)^{\ell} = \left\{\left(x_i, d_i\right), \left(x_j, d_j\right)\right\} \in C^{\ell}$, the *cost* for an assignment (decision) $\left\{\left(x_i, d_i\right), \left(x_j, d_j\right)\right\}$.

For a set $\mathcal{A}$ of variable-value assignments and an objective O_{ℓ}, the cost incurred in optimizing this objective is then given by:

$$R^{\ell}(\mathcal{A}) = \sum_{k} \sum_{\substack{\left(\bowtie_{i,k}, \bowtie_{j,k}\right)^{\ell} = \left\{\left(x_i, d_i\right), \left(x_j, d_j\right)\right\} \in C^{\ell} \\ \left\{\left(x_i, d_i\right), \left(x_j, d_j\right)\right\} \subseteq \mathcal{A}}} f_{i,j,k}^{\ell}\left(d_i, d_j\right)$$

and the solution corresponding to this variable-value assignment $\mathcal{A}$ over all objectives is then characterized by the cost vector

$$\Re(\mathcal{A}) = \left(R^1(\mathcal{A}), R^2(\mathcal{A}), \ldots, R^m(\mathcal{A})\right)$$

A toy example of the above framework is depicted in Fig. 1 for the case of three agents *not* all pairs of which cooperate with each other all the time.

$$\mathcal{S} = \{\mathsf{agent}_1, \mathsf{agent}_2, \mathsf{agent}_3\}$$
$$\mathcal{X} = \{x_1, x_2, x_3\}$$
$$\mathcal{D} = \{D_1, D_2, D_3\},\ \forall i\colon D_i = \{\ \textit{scan},\ \textit{ignore}\ \}$$
$$\mathcal{O} = \{\ \textit{risk},\ \textit{resource budget}\ \}$$
$$C = \left\{C^1, C^2\right\}\colon \begin{array}{l} C^1 = \left\{\left\{\overbrace{(x_1,\ \textit{scan}\ ), (x_2,\ \textit{ignore}\ )}^{\bowtie^1_{1,2,1}}\right\}, \left\{\overbrace{(x_2,\ \textit{ignore}\ ), (x_3,\ \textit{ignore}\ )}^{\bowtie^1_{2,3,1}}\right\}\right\} \\ C^2 = \left\{\left\{\underbrace{(x_1,\ \textit{scan}\ ), (x_2,\ \textit{scan}\ )}_{\bowtie^2_{1,2,1}}\right\}, \left\{\underbrace{(x_1,\ \textit{scan}\ ), (x_2,\ \textit{ignore}\ )}_{\bowtie^2_{1,2,2}}\right\}\right\} \end{array}$$

x_1	x_2	x_3	ℓ	k	$f_{i,j,k}^{\ell}\left(d_i, d_j\right)$
$d_1 = \textit{scan}$	$d_2 = \textit{ignore}$		1	1	6
	$d_2 = \textit{ignore}$	$d_3 = \textit{ignore}$	1	1	3
$d_1 = \textit{scan}$	$d_2 = \textit{scan}$		2	1	7
$d_1 = \textit{scan}$	$d_2 = \textit{ignore}$		2	2	4

Fig. 1 A toy example for the Mo-Dcop framework of Okimoto *et al.* [17]. The shaded row indicates that when *resource budget* is the optimization criterion the cost of agent_1 opting to *scan* and agent_2 opting to *ignore* is 4.

Although ideally one would like to find a solution that optimizes *all* the objective functions *simultaneously*, such a solution may not even exist and thus one would resort to trade-offs among various objectives. One way to handle such a trade-off is

by adopting the concept of *Pareto optimality* from game theory [10] to the above Mo-Dcop formulation in the following manner.

Definition 1. A cost vector $\Re(\mathcal{A}) = \left(R^1(\mathcal{A}), R^2(\mathcal{A}), \ldots, R^m(\mathcal{A})\right)$ is said to (strictly) dominate another cost vector $\Re(\mathcal{A}') = \left(R^1(\mathcal{A}'), R^2(\mathcal{A}'), \ldots, R^m(\mathcal{A}')\right)$, denoted by $\Re(\mathcal{A}) \prec \Re(\mathcal{A}')$, if and only if both the following conditions hold:

- $R^\ell(\mathcal{A}) \leq R^\ell(\mathcal{A}')$ for $1 \leq \ell \leq m$, and
- there exists at least one $\ell \in \{1, 2, \ldots, m\}$ such that $R^\ell(\mathcal{A}) < R^\ell(\mathcal{A}')$.

A cost vector $\Re(\mathcal{A})$ is then called Pareto optimal solution if and only if there does not exist another another feasible cost vector $\mathcal{A}'$ such that $R(\mathcal{A}') \prec R(\mathcal{A})$.

Note that Pareto optimal solutions need *not* be unique. Okimoto *et al.* term a Pareto optimal solution as a *trade-off solution* in [17]. Algorithms for computing Pareto optimal solutions appear in the traditional computer science literature under names such as the *maximal vector computation* problem [11, 12]. A *pseudo-tree* based algorithm for solving multi-objective distributed constraint optimization problems appear in [14]. Okimoto *et al.* [17] extend the algorithmic approach in [14] by adding a pre-processing phase to design a new *branch-and-bound* search algorithm (`BnB`) for finding *all* trade-off solutions[2] using the branch-and-bound technique with a depth-first-search strategy. For evolutionary (genetic) algorithms to solve multi-objective distributed constraint optimization problems, see [4, 6]. One can also design approximation algorithms (heuristics) for solving multi-objective distributed constraint optimization problems, *e.g.*, see the *bounded multi-objective max-sum algorithm* in [8].

5 Optimization Problems In Security For Power Networks

Maintaining a *secure* electric power distribution and transmission system against malicious attacks is an extremely important issue since almost any modern society relies critically on the proper operation of these systems. In this chapter we review some basic optimization problems related to this issue, and the application of ℓ_1-relaxation techniques of Sou *et al.* [22] in solving these optimization problems.

To begin with, a power network model is one with the following components:

- The network topology is specified by a directed graph $G = (V, E)$ with n nodes (buses) and m arcs (transmission lines). The corresponding (directed) edge-node incidence matrix of the graph is denoted by $A \in \{-1, 0, 1\}^{n \times m}$ where
$$A[u, e] = \begin{cases} -1, \text{ if } e = (u, v) \in E \\ 1, \text{ if } e = (v, u) \in E \\ 0, \text{ otherwise} \end{cases}.$$

[2] Okimoto *et al.* [17] claim that an advantage of finding all trade-off solutions is that agents can *dynamically* change decisions in case of emergencies. Unfortunately, the number of trade-off solutions may be exponential in the worst case.

- The physical property of the network is described by a *nonsingular diagonal* matrix $D \in \mathbb{R}^{m\times m}$ such that the reactance of the transmission line (arc) e is $1/D[e,e]$.
- The states of the nodes of the network is summarized by a *state vector* $\theta \in [0,2\pi)^{n-1}$, assuming constant bus voltages throughout but non-constant bus phase angles and using one arbitrary bus (node) as a reference.
- Assuming the DC power flow model and under malicious data attacks, the *measurement vector* z of the states of the buses that is obtained by a state estimator

$$z = H\theta + \widehat{z} \quad \text{with} \quad H = \begin{pmatrix} PD\mathcal{A}^{\mathrm{T}} \\ Q\mathcal{A}D\mathcal{A}^{\mathrm{T}} \end{pmatrix} \tag{9}$$

  where

  – $\widehat{z} \in \mathbb{R}^{n-1}$ is the vector of malicious data attacks [13].
  – $\mathcal{A} \in \mathbb{R}^{(n-1)\times m}$ is obtained from A by removing the row corresponding to the reference bus (node).
  – P is a subset of rows of an identity matrix of appropriate dimension indicating flow measurements of which arcs (transmission lines) are actually taken.
  – Q is a subset of rows of an identity matrix of appropriate dimension indicating power injection measurements of which nodes (buses) are actually taken.

Typically, θ is estimated using the values in H and z. Assuming that the network is *observable* (in control-theoretic terms), it is known that an estimate $\widehat{\theta}$ of θ can be obtained using the following equation where W is a positive definite diagonal matrix [1, 16]:

$$\widehat{\theta} = \left(H^{\mathrm{T}}WH\right)^{-1} WH^{\mathrm{T}}z$$

To detect possible malicious attacks against the measurements via $\widehat{z}$, the commonly performed test [1,16] is used: *if the norm* $\|z - H\widehat{\theta}\|$ *of the following residual quantity*

$$z - H\widehat{\theta} = \left(I - H\left(H^{\mathrm{T}}WH\right)^{-1} WH^{\mathrm{T}}\right)\widehat{z}$$

is large then trigger the alarm.

Although the above test works well if there is a single malicious attack on one data measurement, it may fail under *coordinated* malicious attacks on *multiple* data measurements. For such scenarios, a notion of *security index* was introduced in by Sandberg *et al.* in [20]. Intuitively, a small security index implies that the power network is more vulnerable to malicious attacks. Let H_ℓ denote the ℓ^{th} row of H and the notation $\|X\|_0$ for a vector X denote the cardinality (number of non-zero elements) of X. The security index for the power flow measurement of the k^{th} transmission line (arc) for a given k is formulated as the optimal objective value of the following optimization problem:

$$\begin{array}{rl} \textit{minimize} & \|H\mathbf{x}\|_0 \\ \textit{subject to} & H_k\mathbf{x} = 1 \\ & \mathbf{x} \in \mathbb{R}^{n-1} \end{array} \tag{10}$$

The more general case when certain measurements are *protected* in the sense that they are too secure to be attacked can easily be handled by extending (10) in the following manner [2, 5, 13]. Let $\mathcal{I} \subset \{1,2,\ldots,m\}$ denote the indices of those transmission lines whose power flow measurements are protected and let $H_{\mathcal{I}}$ be the submatrix of H with rows indexed by $\mathcal{I}$. Then, (10) can be generalized to the following cardinality minimization problem:

$$\begin{array}{rl} \textit{minimize} & \|H\mathbf{x}\|_0 \\ \textit{subject to} & H_k\mathbf{x} = 1 \\ & H_{\mathcal{I}}\mathbf{x} = 0 \\ & \mathbf{x} \in \mathbb{R}^{n-1} \end{array} \tag{11}$$

In general, there are no efficient algorithms for solving cardinality minimization problems and thus heuristics are often employed. Sou *et al.* in [22] provides an efficient application of ℓ_1-relaxation techniques to solve an important special case of (11) that assumes $H = PD\mathcal{A}^{\mathrm{T}}$ instead of the more general form shown in (9). They prove that this special case is in fact contained in the following type of optimization problem of a more general nature:

$$\begin{array}{rl} \textit{minimize} & \|C_{\{1,2,\ldots,m\}\setminus\mathcal{I}}\,\mathbf{x}\|_0 \\ \textit{subject to} & C_k\mathbf{x} = 1 \\ & C_{\mathcal{I}}\mathbf{x} = 0 \\ & \mathbf{x} \in \mathbb{R}^{n-1} \end{array} \tag{12}$$

where

- $C \in \mathbb{R}^{m\times(n-1)}$ is a given *totally unimodular* matrix, *i.e.*, a matrix whose every square submatrix has a determinant of 0, 1 or -1,
- for any subset $Y \subset \{1,2,\ldots,m\}$ A_Y is the submatrix of A with rows indexed by Y, and
- A_k is the k^{th} row of A.

Then, a ℓ_1-relaxation of (12) can be obtained by replacing the objective function $\|C_{\{1,2,\ldots,m\}\setminus\mathcal{I}}\,\mathbf{x}\|_0$ by the objective function $\|C_{\{1,2,\ldots,m\}\setminus\mathcal{I}}\,\mathbf{x}\|_1$ that uses the ℓ_1 norm. This ℓ_1-relaxation can in turn be written down as a linear program and solved optimally. Sou *et al.* in [22] prove that an optimal solution of this linear program is in fact also an optimal solution of (12).

6 Conclusion

In this chapter we have surveyed a few optimization frameworks for problems related to security of networked system such as the internet or power grid system. There are other frameworks of modelling network security issues that we have not considered in this chapter, such as game-theoretic formulations or in the context of quantum computing. We believe that as networked systems of various nature become more common in everyday transactions, the corresponding security issues will give rise to more challenging optimization research questions.

Acknowledgements The authors were partially supported by NSF grant IIS-1160995.

References

1. A. Abur and A. Expósito. *Power System State Estimation*, Marcel Dekker, Inc., 2004.
2. R. Bobba, K. Rogers, Q. Wang, H. Khurana, K. Nahrstedt and T. Overbye. *Detecting false data injection attacks on dc state estimation*, 1st Workshop on Secure Control Systems, 2010.
3. S. Boyd and L. Vandenberghe. *Convex Optimization*, Cambridge University Press, 2004.
4. K. Bringmann, T. Friedrich, F. Neumann and M. Wagner. *Approximation-guided evolutionary multi-objective optimization*, 22nd International Joint Conference on Artificial Intelligence, 1198-1203, 2011.
5. G. Dán and H. Sandberg. *Stealth attacks and protection schemes for state estimators in power systems*, 1st IEEE International Conference on Smart Grid Communications, 214-219, 2010
6. K. Deb, S. Agrawal, A. Pratap and T. Meyarivan. *A fast and elitist multiobjective genetic algorithm: NSGA-II*, IEEE Transactions on Evolutionary Computation, 6(2), 182-197, 2002.
7. D. M. Dunlavy, B. Hendrickson and T. G. Kolda. *Mathematical challenges in cybersecurity*, Technical Report SAND2009-0805, Sandia National Laboratories, 2009.
8. F. M. D. Fave, R. Stranders, A. Rogers and N. R. Jennings. *Bounded decentralised coordination over multiple objectives*, 10th International Conference on Autonomous Agents and Multiagent Systems, 371-378, 2011.
9. S. Floyd and V. Paxson. *Difficulties in simulating the internet*, IEEE/ACM Transactions on Networking, 9(4), 392-403, 2001.
10. D. Fudenberg and J. Tirole. *Game theory*, MIT Press, 1991.
11. P. Godfrey, R. Shipley and J. Gryz. *Algorithms and Analyses for Maximal Vector Computation*, VLDB Journal, 16, 5-28, 2006.
12. H. T. Kung, F. Luccio and F. P. Preparata. *On finding the maxima of a set of vectors*, Journal of the ACM, 22(4), 469-476, 1975.
13. Y. Liu, M. Reiter and P. Ning. *False data injection attacks against state estimation in electric power grids*, 16th ACM Conference on Computer and Communication Security, 2132, 2009.
14. T. Matsui, M. Silaghi, K. Hirayama, M. Yokoo and H. Matsuo. *Distributed search method with bounded cost vectors on multiple objective dcops*, 15th International Conference on Principles and Practice of Multi-Agent Systems, 137-152, 2012.
15. P. Modi, W.-M. Shen, M. Tambe and M. Yokoo. *ADOPT: asynchronous distributed constraint optimization with quality guarantees*, Artificial Intelligence, 161(1-2), 149-180, 2005.
16. A. Monticelli. *State Estimation in Electric Power Systems A Generalized Approach*, Kluwer Academic Publishers, 1999.
17. T. Okimoto, N. Ikegai, K. Inoue, H. Okada, T. Ribeiro and H. Maruyama. *Cyber security problem based on multi-objective distributed constraint optimization technique*, 43rd Annual IEEE/IFIP Conference on Dependable Systems and Networks Workshop, 1-7, 2013.

18. J. Meza, S. Campbell and D. Bailey. *Mathematical and Statistical Opportunities in Cyber Security*, Report LBNL-1667E, Lawrence Berkeley National Laboratory, 2009.
19. V. Paxson. *Strategies for sound internet measurement*, 4th ACM SIGCOMM conference on Internet measurement, 263-271, 2004.
20. H. Sandberg, A. Teixeira and K. H. Johansson. *On security indices for state estimators in power networks*, 1st Workshop on Secure Control Systems, 2010.
21. A. J. Slagell, K. Lakkaraju, and K. Luo. *Flaim: A multi-level anonymization framework for computer and network logs*, 20th USENIX Large Installation System Administration Conference, 63-77, 2006.
22. K. C. Sou, H. Sandberg and K. H. Johansson. *On the exact solution to a smart grid cyber-security analysis problem*, IEEE Transactions on Smart Grid, 4(2), 856-865, 2013.
23. K. G. Vamvoudakis, J. P. Hespanha, R. A. Kemmerer, and G. Vigna. *Formulating Cyber-Security as Convex Optimization Problems*, Control of Cyber-Physical Systems, Volume 449 of Lecture Notes in Control and Information Sciences, 85-100, 2013.
24. G. Vigna, *The 2011 UCSB iCTF: Description of the game*, `http://ictf.cs.ucsb.edu/`, 2011.