Linear Quadratic Gaussian Control Under False Data Injection Attacks

Andrew Clark and Luyao Niu

***Abstract*— In a false data injection attack, an adversary compromises one or more sensors of a networked system and introduces false measurements in order to bias the control and degrade the system performance. In this paper, we investigate the problem of designing controllers for linear systems with Gaussian noise in order to minimize a quadratic cost under both normal operating conditions and false data injection attacks. We develop a two-stage approach, in which the controller chooses a set of admissible control signals in the first stage, which limits the worst-case damage that the adversary can cause by introducing false data. The control action at each time step is then selected at the second stage. We demonstrate that both stages can be solved optimally using convex optimization techniques and present efficient algorithms for choosing the optimal control policy. Our approach is evaluated through numerical study.**

I. Introduction

Advances in embedded systems have enabled integration of highly accurate sensors into diverse control applications. Such sensors are key enablers of cyber-physical systems (CPS), with examples including GPS and other positioning systems for vehicle navigation [1], haptic feedback sensors for telerobotics [2], and state estimation systems in power grids [3]. Sensors may either be co-located with the plant and actuation, or geographically distributed; in the latter case, the sensors may be unattended and communicate with the controller via wired or wireless networks, as is typically the case in networked control systems.

Sensors are appealing targets for malicious attacks on CPS for a variety of reasons. First, their low cost and the inherently open nature of sensing and wireless communication makes them easy to compromise via attacks including physical capture of sensors, compromise of communication channels, and introduction of deceptive signals to bias the sensor measurements [4], [5]. Second, since such attacks do not necessarily violate any cryptographic or other security measures, they may only be detectable through statistical tests on sensor readings, which can be deceived by an intelligent adversary. Third, false data that are introduced into sensors by an adversary may lead to incorrect controller behavior, resulting in suboptimal performance or even safety and stability violations. Such behaviors have been demonstrated in diverse application domains including vehicular [6], energy [7], and water infrastructure systems [8].

The severity of the threat of false data injection has led to significant research interest in modeling and detecting such attacks, as well as developing resilient state estimation techniques [9], [10], [11], [12]. These existing works have focused on identifying the optimal strategy for an adversary to maximally disrupt a targeted system [13], as well as new methodologies for detecting which sensors have been compromised [14], [15]. At present, however, the problem of synthesizing controllers that provide performance guarantees in the presence of false data injection attacks, while still preserving system performance in the absence of such attacks, has received relatively little attention in the literature.

In this paper, we investigate the problem of optimal Linear Quadratic Gaussian (LQG) control in the presence of an adversary that has compromised one or more sensors. We focus on the LQG control problem due to its widespread use, and as a basic building block of more complex controller designs. We consider a min-max setting, in which the controller first selects a policy that determines the control response to each set of sensor measurements. The adversary then observes the policy and chooses a set of false data to inject in order to maximally disrupt the system performance, as measured by a quadratic cost function.

Our approach is based on the insight that the worst-case performance achievable by the adversary is bounded above by the maximum cost resulting from any set of control inputs that can be achieved by injecting a sequence of measurements. Motivated by this, we develop a two-stage approach to designing an LQG controller. In the first stage, the system chooses control actions that are admissible in order to optimize a trade-off between the expected performance in the absence of an attacker, and the worst-case performance in the presence of an attacker. In the second stage, for the given set of observed sensor measurements, the system selects an optimal control strategy.

Due to the complexity of the problem, we adopt a receding-horizon design in which, at each time step, the controller chooses the next L control actions by estimating the cost function at future time steps. We show that under this design, both stages described above can be formulated as convex optimization problems and solved in polynomial time. We present our proposed algorithms and evaluate them through numerical study, which shows that our proposed approach significantly reduces the quadratic cost under adversarial attacks compared to optimal LQG control, while experiencing only a small degradation in performance in a non-adversarial setting.

This paper is organized as follows. Section II reviews the related work. Section III presents the system and adversary models. Section IV presents our problem formulation and solution approach. Section V discusses our proposed algorithms. Section VI contains the results of our numerical

A. Clark and L. Niu are with the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA 01609, USA {aclark, lniu}@wpi.edu. This paper was supported by NSF grant CNS-1656981.

study. Section VII concludes the paper.

II. Related Work

Linear Quadratic Gaussian control has received extensive attention in the control community spanning several decades [16]. The standard methodologies, however, assume disturbances that are stochastic and independent of the control action, which does not hold in an adversarial setting. In general, the impact of cyber and other malicious attacks on the performance of control systems has only recently gained significant interest. False data injection attacks on power systems were studied in [11]. Techniques for detecting false data injection attacks have been proposed in [17], [15], [14], [9]. The related problem of correctly estimating the system state in the presence of false data attacks has been studied in works including [18], [19], [12].

The impact of false data injection attacks on control performance has mainly been studied from the perspective of the adversary, i.e., computing the optimal adversary strategy in order to maximally disrupt the system performance. In [13], the optimal adversary strategy to maximize a quadratic cost function for a system using an LQG controller with Kalman filter and χ^2 failure detector was characterized. This approach, however, does not address how the controller design can be modified in order to mitigate the attack.

Resilient control synthesis problems have been studied for other classes of attacks. Resilient control in the presence of denial of service attacks such as jamming was formulated in [20], [21]. More powerful adversaries who are able to directly tamper with control measurements were studied in [22]. These works consider related attacks but are not directly generalizable to false data injection.

III. Preliminaries

This section presents the system and adversary models.

A. System Model

We consider a discrete-time LTI system with time index $k = 1, 2, \ldots$, state $\mathbf{x}_k \in \mathbb{R}^n$, input $\mathbf{u}_k \in \mathbb{R}^m$, and output $\mathbf{z}_k \in \mathbb{R}^p$. In the absence of any adversary, the system state and output have dynamics

$$\begin{aligned} \mathbf{x}_{k+1} &= A\mathbf{x}_k + B\mathbf{u}_k + \mathbf{w}_k \\ \mathbf{z}_k &= C\mathbf{x}_k + \mathbf{v}_k \end{aligned}$$

where A, B, and C are matrices with known dimensions and $\mathbf{w}_k$ and $\mathbf{v}_k$ are i.i.d. Gaussian random noise vectors with distributions $\mathbf{w}_k \sim N(0, \Sigma_w)$ and $\mathbf{v}_k \sim N(0, \Sigma_v)$. Both $\mathbf{v}_k$ and $\mathbf{w}_k$ are independent of each other and the preceding values of $\mathbf{z}$, $\mathbf{w}$, $\mathbf{u}$, and $\mathbf{v}$. The initial state $\mathbf{x}_0$ is assumed to be distributed as $\mathbf{x}_0 \sim N(0, \Sigma_0)$.

The controller has knowledge of the matrices A, B, C, and the covariance matrices Σ_w and Σ_v. The controller observations up to time k are given by $I_k = \{\mathbf{u}_0, \ldots, \mathbf{u}_{k-1}\} \cup \{\mathbf{z}_0, \ldots, \mathbf{z}_k\}$. Letting $\mathcal{I}_k$ denote the set of all possible values of I_k, a *control policy* is a set of mappings $\mu_k : \mathcal{I}_k \to \mathbb{R}^m$ for $k = 0, 1, \ldots,$ which describe the control action selected at each time k based on the prior observations of the controller. In this paper, we focus on deterministic control policies. For any μ_k, we let $\mathcal{U}_{\mu_k}(\mathcal{I}_k) = \{\mu_k(I_k) : I_k \in \mathcal{I}_k\} \subseteq \mathbb{R}^m$, i.e., the set of control actions that are chosen by policy μ_k for some set of observations I_k. We define $\boldsymbol{\mu} = \{\mu_k : k = 0, 1, \ldots, \}$ to be a sequence of policies over time.

The goal of the controller is to choose a policy that minimizes the cost function

$$J(\mu) = \mathbf{E}\left\{\sum_{k=0}^{L} (\mathbf{x}_k^T Q_k \mathbf{x}_k + \mathbf{u}_k R_k \mathbf{u}_k)\right\}, \qquad (1)$$

where Q_k and R_k are symmetric, positive-definite matrices and the expectation is taken over the sensor noise $\mathbf{v}_k$ and process noise $\mathbf{w}_k$.

B. Adversary Model

The adversary is assumed to corrupt one or more sensors. By injecting false sensor measurements, the adversary attempts to bias the system into reaching an undesired operating point. We let $\boldsymbol{\alpha} \subseteq \{1, \ldots, p\}$ denote the set of adversary-controlled sensors. The adversary is able to inject arbitrary inputs to the sensors in $\boldsymbol{\alpha}$, so that the output z_k^i of sensor i at time k is given by

$$z_k^i = \begin{cases} C_i \mathbf{x}_k + \mathbf{v}_k^i, & i \notin \boldsymbol{\alpha} \\ a_k^i, & i \in \boldsymbol{\alpha} \end{cases}$$

where C_i denotes the i-th row of C, $\mathbf{v}_k^i$ denotes the i-th entry of $\mathbf{v}_k$, and a_k^i is the signal introduced by the adversary at time k. The set of compromised sensors is a random variable, with $Pr(\boldsymbol{\alpha})$ denoting the probability of a given set of compromised sensors $\boldsymbol{\alpha}$. The event $\bar{\boldsymbol{\alpha}}$ occurs when $\boldsymbol{\alpha} = \emptyset$, and we let $Pr(\bar{\boldsymbol{\alpha}})$ denote the probability that no sensor has been compromised. For a given $\boldsymbol{\alpha}$, we let $\mathbf{z}_k^{\boldsymbol{\alpha}} = (z_k^i : i \in \boldsymbol{\alpha})$ and $\hat{\mathbf{z}}_k^{\boldsymbol{\alpha}} = (z_k^i : i \notin \boldsymbol{\alpha})$ denote the outputs provided by compromised and uncompromised sensors, respectively. The set $I_k^{\boldsymbol{\alpha}} = \{\mathbf{u}_0, \ldots, \mathbf{u}_k\} \cup \{\hat{\mathbf{z}}_0^{\boldsymbol{\alpha}}, \ldots, \hat{\mathbf{z}}_k^{\boldsymbol{\alpha}}\}$.

At each time k, the adversary is assumed to have knowledge of the control policy μ_k, the state value $\mathbf{x}_k$, and the previous measurements $\mathbf{z}_0, \ldots, \mathbf{z}_k$. The control policy μ_k can be inferred by observing the system behavior over a period of time. The state value $\mathbf{x}_k$ may be estimated by the adversary by deploying a sensor network to monitor the targeted system. The sensor measurements $\mathbf{z}_k$ may be captured by eavesdropping on the communication channel between the sensors and controller. Taken together, these modeling assumptions capture the worst case of the attacker information. The adversary's information set at time k is denoted

$$I_k^{\mathcal{A}} = \{\mathbf{z}_0, \ldots, \mathbf{z}_k\} \cup \{\mathbf{u}_0, \ldots, \mathbf{u}_k\} \cup \{\mathbf{x}_0, \ldots, \mathbf{x}_k\}.$$

We let $\mathcal{I}_k^{\mathcal{A}}$ denote the set of possible adversary observations up to time k. Conversely, it is assumed that the controller has knowledge of the probability distribution $Pr(\boldsymbol{\alpha})$, but not the value of $\boldsymbol{\alpha}$ itself.

The adversary's strategy consists of a policy $\tau_k : \mathcal{I}_k^{\mathcal{A}} \to \mathbb{R}^{|\boldsymbol{\alpha}|}$, for $k = 0, 1, \ldots$, which is a (possibly stochastic) mapping from the set of observations by the adversary up to time k to a set of false inputs at time k. We let $\boldsymbol{\tau} = \{\tau_k :$

$k = 0, 1, 2, \ldots, \}$ denote a sequence of adversary policies over time. The goal of the adversary is to select a policy that disrupts the system performance by maximizing the system's cost function (1).

IV. Problem Formulation and Solution Approach

This section gives an overview of our proposed approach to LQG in the presence of false data injection. We first give the problem formulation, and then describe a two-stage solution approach.

A. Problem Formulation

The problem formulation is given as

$$\min_{\boldsymbol{\mu}} \max_{\boldsymbol{\tau}} \left\{ \mathbf{E} \left(\sum_{k=0}^{L} (\mathbf{x}_k^T Q_k \mathbf{x}_k + \mathbf{u}_k^T R_k \mathbf{u}_k) \right) \right\} \tag{2}$$

where the expectation is over $\boldsymbol{\alpha}$, $\mathbf{w}_k$, and $\mathbf{v}_k$. Under the min-max formulation of (2), the controller attempts to choose a policy μ that minimizes the expected cost function $J(\mu)$ defined in (1), while the adversary chooses a policy τ that maximizes the cost after observing μ.

We take a receding-horizon approach to solving (2). Under our approach, at each time k the controller chooses a policy μ_k that maps the information set I_k to a sequence of control actions $\mathbf{u}_k, \ldots, \mathbf{u}_{k+L}$, with the goal of selecting a mapping μ_k that minimizes

$$\max_{\tau} \mathbf{E} \left(\sum_{l=k}^{k+L} (\mathbf{x}_l^T Q_l \mathbf{x}_l + \mathbf{u}_l^T R_l \mathbf{u}_l) \right).$$

The following lemma gives a bound on the worst-case cost experienced by a given policy. As a preliminary, define $\mathbf{u}_{k:(k+L)} = (\mathbf{u}_k, \ldots, \mathbf{u}_{k+L})$, $\mathbf{x}_{k:(k+L)} = (\mathbf{x}_k, \ldots, \mathbf{x}_{k+L})$, and

$$c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}) = \sum_{l=k}^{k+L} (\mathbf{x}_l^T Q_l \mathbf{x}_l + \mathbf{u}_l^T R_l \mathbf{u}_l).$$

Define

$$\mathcal{U}_\mu(I_k^{\boldsymbol{\alpha}}) = \bigcup_{\mathbf{z}_{0:k}^{\boldsymbol{\alpha}}} \mathcal{U}_\mu(I_k^{\boldsymbol{\alpha}} \cup \{\mathbf{z}_{0:k}^{\boldsymbol{\alpha}}\})$$

$$\mathcal{U}_{k:(k+L)} = \bigcup_{l=k}^{k+L} \mathcal{U}_\mu(I_l^{\boldsymbol{\alpha}}).$$

The set $\mathcal{U}_\mu(I_k^{\boldsymbol{\alpha}})$ is the set of possible control actions when $I_k^{\boldsymbol{\alpha}}$, while $\mathcal{U}_{k:(k+L)}$ is the set of all possible control actions during steps k to $(k+L)$.

Lemma 1: For any control policy μ, the expected cost is bounded above by

$$J(\mu) \leq \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}) | \bar{\boldsymbol{\alpha}}) Pr(\bar{\boldsymbol{\alpha}}) + \sum_{\boldsymbol{\alpha}} \Big[Pr(\boldsymbol{\alpha}) \cdot \int_{\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}} \int_{\mathbf{x}_k} \max_{\substack{\mathbf{u}_{k:(k+L)} \\ \in \mathcal{U}_\mu(\mathcal{I}_k^{\boldsymbol{\alpha}})}} \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)})) \; d\mathbf{x}_k \; d\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}} \Big] \tag{3}$$

Proof: We have that

$$J(\mu) = \mathbf{E}_\mu(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}) | \bar{\boldsymbol{\alpha}}) Pr(\bar{\boldsymbol{\alpha}}) + \sum_{\boldsymbol{\alpha}} Pr(\boldsymbol{\alpha}) \max_{\boldsymbol{\tau}} \left\{ \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}) | \boldsymbol{\tau}) \right\}$$

where conditioning on $\boldsymbol{\tau}$ implies that the measurements $\mathbf{z}_k^{\boldsymbol{\alpha}}$ are chosen according to the adversary policy $\boldsymbol{\tau}$. The second term can be bounded by

$$\max_{\tau} \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)} | \tau)) \tag{4}$$

$$= \int_{\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}} \int_{\mathbf{x}_k} \Big\{ \max_{\tau} \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}) | \tau, \hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}, \mathbf{x}_k) \tag{5}$$

$$\cdot Pr(\mathbf{x}_k | \hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}) Pr(\hat{\mathbf{z}}_{0:k} | \boldsymbol{\alpha}) \; d\mathbf{x}_k \; d\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}} \Big\}$$

$$\leq \int_{\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}} \int_{\mathbf{x}_k} \Big\{ \max_{\mathbf{z}_{0:k}^{\boldsymbol{\alpha}}} \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}) | \tau, \hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}, \mathbf{x}_k) \tag{6}$$

$$\cdot Pr(\mathbf{x}_k | \hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}) Pr(\hat{\mathbf{z}}_{0:k} | \boldsymbol{\alpha})) \; d\mathbf{x}_k \; d\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}} \Big\}$$

$$= \int_{\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}} \int_{\mathbf{x}_k} \max_{\substack{\mathbf{u}_{k:(k+L)} \in \\ \mathcal{U}_{k:(k+L)}(I_k^{\boldsymbol{\alpha}})}} \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}) | \hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}, \mathbf{x}_k)$$

$$\cdot Pr(\mathbf{x}_k | \hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}) Pr(\hat{\mathbf{z}}_{0:k} | \boldsymbol{\alpha}) \; d\mathbf{x}_k \; d\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}} \tag{7}$$

Eq. (5) follows from the fact that the adversary's policy τ takes $\mathbf{x}_k$ and $\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}$ as inputs. Eq. (6) holds because the adversary attempts to maximize the cost function over all functions of the form $\tau : \mathcal{I}_{\boldsymbol{\alpha}}^{\mathcal{A}}(k) \rightarrow \{\mathbf{z}_{0:k}^{\boldsymbol{\alpha}}\}$, which is bounded by the maximizer over all $\mathbf{z}_{0:k}^{\boldsymbol{\alpha}}$. Finally, Eq. (7) holds because $\mathbf{u}_l \in \mathcal{U}_l(\mathcal{I}_l^{\boldsymbol{\alpha}})$ if and only if there is a sequence of measurements $\mathbf{z}_{0:k}^{\boldsymbol{\alpha}}$ such that $\mu(\mathbf{z}_{0:k}) = \{\mathbf{u}_k, \ldots, \mathbf{u}_{k+L}\}$. Summing Eq. (7) over $\boldsymbol{\alpha}$ yields the desired upper bound. ■

The implication of Lemma 1 is that the adversary will attempt to increase the cost $J(\mu)$ by introducing false inputs that cause the controller to implement the set of actions that maximizes the cost function. When the control policy is deterministic, the adversary can effectively choose the set of control actions by manipulating the inputs.

Motivated by this result, we develop a two-stage approach to the controller design problem. At the first stage, we select a set of admissible control inputs $\mathcal{U}$ at time k based on the measurements $I_k^{\boldsymbol{\alpha}}$ for each possible set of compromised sensors $\boldsymbol{\alpha}$. The controller effectively "commits" to only choosing control actions from this set in order to minimize the impact of the attack. At the second stage, we select a control action from within this set in order to minimize the cost function based on the observations $\mathcal{I}_k$. The two stages of our approach are described in the following two sections.

B. Selecting the Admissible Control Inputs

This section presents our approach to selecting a set of admissible control inputs $\mathcal{U}_\mu(I_k)$ at time k. Our approach is to select $\mathcal{U}_\mu$ in order to ensure that the second term of (3) is bounded above. Intuitively, this corresponds to setting an upper bound on the worst-case cost that can be achieved by false data injection.

For each set $\boldsymbol{\alpha}$ with $Pr(\boldsymbol{\alpha}) > 0$, we choose a set $\mathcal{U}^{\boldsymbol{\alpha}}(I_k^{\boldsymbol{\alpha}})$ based on the observations from sensors outside $\boldsymbol{\alpha}$. We then

take

$$\mathcal{U}(I_k) = \bigcap_{\boldsymbol{\alpha}} \mathcal{U}(I_k^{\boldsymbol{\alpha}}).$$

In particular, we select a set of bounds $\{\gamma(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^n\}$ and then construct $\mathcal{U}_\mu(\mathcal{I}_k)$ as

$$\mathcal{U}(\gamma) = \bigcap_{\mathbf{x}\in\mathbb{R}^n} \left\{\mathbf{u}_{k:(k+L)} : \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)})|\mathbf{x}_k = \mathbf{x}) \leq \gamma(\mathbf{x})\right\}. \quad (8)$$

The first stage of the optimal control problem is therefore to choose a set of γ values as

$$\min_{\gamma} \left\{ \mathbf{E}\left(\min_{\substack{\mathbf{u}_{k:(k+L)} \\ \in \mathcal{U}(\gamma)}} \{\mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)})|\bar{\boldsymbol{\alpha}})\} \right) Pr(\bar{\boldsymbol{\alpha}}) + \sum_{\boldsymbol{\alpha}} Pr(\boldsymbol{\alpha}) \int_{\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}} \int_{\mathbf{x}_k} \gamma(\mathbf{x}_k) Pr(\mathbf{x}_k, \hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}})\, d\mathbf{x}_k\, d\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}} \right\} \quad (9)$$

The first challenge in solving this problem is the fact that there are uncountably many possible values of $\mathbf{x}_k$. We mitigate this problem by discretizing the domain. We divide $\mathbb{R}^n$ into regions of the form

$$\Lambda_i^{\boldsymbol{\alpha}} = \{\mathbf{x} : Pr(\mathbf{x}|\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}) \in [\beta_i, \beta_{i+1}]\}$$

for some values of $\beta_1, \ldots, \beta_N \in [0, 1]$ with $0 \leq \beta_1 < \cdots < \beta_N \leq 1$. Since $\mathbf{x}_k$ and $\hat{\mathbf{z}}_k^{\boldsymbol{\alpha}}$ are Gaussian random variables, the set Λ_i is equivalent to

$$(\mathbf{x} - \hat{\mathbf{x}}_k^{\boldsymbol{\alpha}})^T \Sigma_{k,\boldsymbol{\alpha}}^{-1} (\mathbf{x} - \hat{\mathbf{x}}_k^{\boldsymbol{\alpha}}) \in [\sigma_{i+1}, \sigma_i]$$

where $\sigma_i = -\log \beta_i$ and $\hat{\mathbf{x}}_k$ and $\Sigma_{k,\boldsymbol{\alpha}}$ are the mean and covariance of $\mathbf{x}_k$ conditioned on $\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}$, which can be tracked over time by a Kalman filter.

The set $\mathcal{U}(\gamma)$ is then defined by

$$\mathcal{U}(\gamma) = \bigcap_{i=1}^{N} \Big\{ \mathbf{u}_{k:(k+L)} : \max_{\mathbf{x}_k \in \Lambda_i^{\boldsymbol{\alpha}}} \left\{\mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)})|\mathbf{x}_k)\right\} \leq \gamma_i^{\boldsymbol{\alpha}} \Big\}.$$

The inequality

$$c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}|\mathbf{x}_k) \leq \gamma_i^{\boldsymbol{\alpha}}$$

can be expressed in a more tractable form as

$$\mathbf{x}_k^T \overline{Q} \mathbf{x}_k + \mathbf{u}_{k:(k+L)}^T \overline{R} \mathbf{u}_{k:(k+L)} + \mathbf{u}_{k:(k+L)}^T \overline{S} \mathbf{x}_k \leq \gamma_i^{\boldsymbol{\alpha}}$$

by defining

$$\overline{Q} = \sum_{l=k}^{k+L} (A^{l-k})^T Q_l A^{l-k}$$

$$\overline{S} = 2(\overline{S}_{k+1} \cdots \overline{S}_{k+L-1})$$

$$\overline{R} = \begin{pmatrix} \overline{R}_{11} & \cdots & \overline{R}_{1L} \\ \vdots & \cdots & \vdots \\ \overline{R}_{L1} & \cdots & \overline{R}_{LL} \end{pmatrix}$$

Here,

$$S_{l'} = \sum_{l=l'+1}^{k+L} (A^{l-k})^T Q_l A^{l-1-l'} B$$

and

$$R_{l'm} = \begin{cases} \sum_{l=\max(l',m)+1}^{k+L} B^T (A^{l-1-l'})^T Q_l A^{l-1-m} B, & l' \neq m \\ \sum_{l=\max(l',m)+1}^{k+L} B^T (A^{l-1-l'})^T Q_l A^{l-1-m} B + R_m, & l' = m \end{cases}$$

Since computing

$$\max_{\mathbf{x}_k \in \Lambda_i^{\boldsymbol{\alpha}}} \{\mathbf{x}_k^T \overline{Q} \mathbf{x}_k + \mathbf{u}_{k:(k+L)}^T \overline{R} \mathbf{u}_{k:(k+L)} + \mathbf{u}_{k:(k+L)}^T \overline{S} \mathbf{x}_k\}$$

for a fixed $\mathbf{u}_{k:(k+L)}$ involves maximizing a convex function of $\mathbf{x}_k$, testing membership in $\mathcal{U}(\gamma)$ may still be computationally intractable. Hence, we introduce one additional discretization by constructing the sets

$$\Lambda_{ij}^{\boldsymbol{\alpha}} = \{\mathbf{x} : Pr(\mathbf{x}|\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}) \in [\beta_i, \beta_{i+1}],\ \mathbf{x}^T \overline{Q} \mathbf{x} \in [\delta_j, \delta_{j+1}]\}.$$

The problem of selecting the set of control actions is then equivalent to choosing $\gamma_{ij}^{\boldsymbol{\alpha}}$ for $i, j = 1, \ldots, N$, where $\gamma_{ij}^{\boldsymbol{\alpha}}$ serves as an upper bound for $\mathbf{x}_k \in \Lambda_{ij}^{\boldsymbol{\alpha}}$. Under this approach, the definition of $\mathcal{U}(\gamma)$ is then given as

$$\mathcal{U}(\gamma) = \bigcap_{i,j=1}^{N} \Big\{ \mathbf{u}_{k:(k+L)} : \max_{\mathbf{x}_k \in \Lambda_{ij}^{\boldsymbol{\alpha}}} \left\{\mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_k)|\mathbf{x}_k)\right\} \leq \gamma_{ij}^{\boldsymbol{\alpha}} \Big\}.$$

We are now ready to state the formulation for the problem of selecting $\mathcal{U}$. Define the function $g(\gamma)$ as

$$g(\gamma) = \min \left\{ \mathbf{E}\left(\sum_{l=k}^{k+L} \mathbf{x}_l^T Q_l \mathbf{x}_l + \mathbf{u}_l^T R_l \mathbf{u}_l \right) : \mathbf{u}_{k:(k+L)} \in \mathcal{U}(\gamma) \right\}$$

and define $f(\gamma)$ as

$$f(\gamma) = g(\gamma) + \sum_{\boldsymbol{\alpha}} Pr(\boldsymbol{\alpha}) \sum_{i,j=1}^{N} \gamma_{ij}^{\boldsymbol{\alpha}} Pr(\mathbf{x}_k \in \Lambda_{ij}^{\boldsymbol{\alpha}}|\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}).$$

The problem of selecting $\mathcal{U}$ is then equivalent to

$$\min_{\gamma} f(\gamma). \quad (10)$$

The following theorem leads to efficient algorithms for solving (10).

Theorem 1: The function $f(\gamma)$ is convex in γ.

Proof: The term

$$\sum_{\boldsymbol{\alpha}} Pr(\boldsymbol{\alpha}) \sum_{i,j=1}^{N} \gamma_{ij}^{\boldsymbol{\alpha}} Pr(\mathbf{x}_k \in \Lambda_{ij}^{\boldsymbol{\alpha}}|\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}})$$

is linear and hence convex in γ. It remains to show convexity of $g(\gamma)$.

Define $h(\mathbf{u}_{k:(k+L)})$ and $h_{ij}(\mathbf{u}_{k:(k+L)})$ by

$$\begin{aligned} h(\mathbf{u}_{k:(k+L)}) &= \mathbf{x}_k^T \overline{Q} \mathbf{x}_k + \mathbf{u}_{k:(k+L)}^T \overline{R} \mathbf{u}_{k:(k+L)} \\ &\quad + \mathbf{u}_{k:(k+L)}^T \overline{S} \mathbf{x}_k \\ h_{ij}^{\boldsymbol{\alpha}}(\mathbf{u}_{k:(k+L)}) &= \max_{\mathbf{x}_k \in \Lambda_{ij}^{\boldsymbol{\alpha}}} \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}) | \mathbf{x}_k) \end{aligned}$$

The value of $g(\gamma)$ is therefore equivalent to

$$\begin{array}{ll} \min & h(\mathbf{u}_{k:(k+L)}) \\ \mathbf{u}_{k:(k+L)} & \\ \text{s.t.} & h_{ij}^{\boldsymbol{\alpha}}(\mathbf{u}_{k:(k+L)}) \leq \gamma_{ij}^{\boldsymbol{\alpha}}, \ \forall i, j, \boldsymbol{\alpha} \end{array} \tag{11}$$

Since each h_{ij} is a pointwise maximum of convex functions of $\mathbf{u}_{k:(k+L)}$, and hence is convex, $g(\gamma)$ is the solution to a convex program. By Slater's Criterion [23], strong duality holds and hence

$$\begin{aligned} g(\gamma) = \max_{\lambda \geq 0} \inf_{\mathbf{u}_{k:(k+L)}} \{ & h(\mathbf{u}_{k:(k+L)}) \\ & + \sum_{\boldsymbol{\alpha}, i, j} Pr(\boldsymbol{\alpha}) \lambda_{ij}^{\boldsymbol{\alpha}} (h_{ij}^{\boldsymbol{\alpha}}(\mathbf{u}_{k:(k+L)}) - \gamma_{ij}^{\boldsymbol{\alpha})} \} \end{aligned}$$

The function within the infimum is jointly convex in $\mathbf{u}$ and γ, and hence is convex in γ. The function $g(\gamma)$ is therefore a pointwise maximum (over λ) of convex functions of γ, and hence is convex. ■

In Section V, we will present an algorithm that exploits convexity to compute an optimal $\mathcal{U}(\gamma)$.

C. Selecting a Control Action at Each Time Step

After the set of control actions $\mathcal{U}(\gamma)$ is chosen, the second stage of our approach selects a control action based on the observed sensor inputs in order to minimize the cost function. This problem can be stated as

$$\begin{array}{ll} \text{minimize} & \mathbf{E}\left\{ \sum_{l=k}^{k+L} (\mathbf{x}_l^T Q_l \mathbf{x}_l + \mathbf{u}_l^T R_l \mathbf{u}_l) | I(k), \bar{\boldsymbol{\alpha}} \right\} \\ \text{s.t.} & \max_{\mathbf{x}_k \in \Lambda_{ij}^{\boldsymbol{\alpha}}} \left\{ \delta_j + \mathbf{u}_{k:(k+L)}^T \overline{R}_{ij} \mathbf{u}_{k:(k+L)} \right. \\ & \left. + \mathbf{x}_k^T \overline{S}_{ij} \mathbf{u}_{k:(k+L)} \right\} \leq \gamma_{ij}^{\boldsymbol{\alpha}} \ \forall i, j, \boldsymbol{\alpha} \end{array} \tag{12}$$

We adopt a certainty-equivalent approach [24], in which we assume that $\mathbf{x}_k$ is equal to its maximum-likelihood value and all subsequent values of $\mathbf{x}_l$ are equal to their maximum-likelihood values. This leads to a refined formulation

$$\begin{array}{ll} \text{minimize} & \hat{\mathbf{x}}_k^T \overline{Q} \hat{\mathbf{x}}_k + \mathbf{u}_{k:(k+L)}^T R \mathbf{u}_{k:(k+L)} + \hat{\mathbf{x}}_k^T \overline{S} \mathbf{u}_{k:(k+L)} \\ \text{s.t.} & \max_{\mathbf{x}_k \in \Lambda_{ij}^{\boldsymbol{\alpha}}} \left\{ \delta_j + \mathbf{u}_{k:(k+L)}^T \overline{R}_{ij} \mathbf{u}_{k:(k+L)} \right. \\ & \left. + \mathbf{x}_k^T \overline{S}_{ij} \mathbf{u}_{k:(k+L)} \right\} \leq \gamma_{ij}^{\boldsymbol{\alpha}} \ \forall i, j, \boldsymbol{\alpha} \end{array} \tag{13}$$

Eq. (13) is a convex optimization problem that can be solved in polynomial time. Algorithms for both stages of our proposed approach are described in the following section.

V. Algorithms for Choosing Control Policy

In this section, we present algorithms for selecting the optimal control policy, including choosing the set of admissible actions and choosing an action at each stage. We first consider the problem of selecting the optimal action at each stage, which will then act as a subroutine. The problem (13) can be solved using a barrier function method, where the distance from each constraint set

$$\max_{\mathbf{x}_k \in \Lambda_{ij}^{\boldsymbol{\alpha}}} \delta_j + \mathbf{u}_{k:(k+L)}^T \overline{R}_{ij} \mathbf{u}_{k:(k+L)} + \mathbf{x}_k^T \overline{S}_{ij} \mathbf{u}_{k:(k+L)} \leq \gamma_{ij}^{\boldsymbol{\alpha}}$$

can be computed in polynomial time by solving a quadratic optimization problem.

The problem of choosing the optimal set of admissible control actions $\mathcal{U}(\gamma)$ can be solved via a subgradient approach. We have the following preliminary result.

Proposition 1: Let λ^* be a dual solution to (11). Then $-\lambda^*$ is a subgradient of $g(\gamma_0)$ at γ_0.

Proof: By the Lagrangian dual theorem, we have that

$$\begin{aligned} & g(\gamma) \\ & \leq \max_{\lambda} \left\{ \inf_{\mathbf{u}} \mathbf{x}^T \overline{Q} \mathbf{x} + \mathbf{u}^T \overline{R} \mathbf{u} + \mathbf{u}^T \overline{S} \mathbf{x} + \lambda^T (f_{ij}(\mathbf{u}) - \gamma) \right\}. \end{aligned}$$

The maximum is achieved at the dual optimal solution λ^*. Hence for any γ and γ_0,

$$\begin{aligned} g(\gamma) - g(\gamma_0) &= \max_{\lambda} \left\{ \inf_{\mathbf{u}} \{ f(\mathbf{u}) + \lambda^T (h(\mathbf{u}) - \gamma) \} \right\} \\ &\quad - \max_{\lambda} \left\{ \inf_{\mathbf{u}} \{ f(\mathbf{u}) + \lambda^T (h(\mathbf{u}) - \gamma_0) \} \right\} \\ &= \max_{\lambda} \left\{ \inf_{\mathbf{u}} \{ f(\mathbf{u}) + \lambda^T (h(\mathbf{u}) - \gamma) \} \right\} \\ &\quad - \inf_{\mathbf{u}} \{ f(\mathbf{u}) + (\lambda^*)^T (h(\mathbf{u}) - \gamma_0) \} \\ &\geq \inf_{\mathbf{u}} \{ f(\mathbf{u}) + (\lambda^*)^T (h(\mathbf{u}) - \gamma_) \} \\ &\quad - \inf_{\mathbf{u}} \{ f(\mathbf{u}) + (\lambda^*)^T (h(\mathbf{u}) - \gamma_0) \} \\ &= -(\lambda^*)^T (\gamma - \gamma_0), \end{aligned}$$

implying that $-\lambda^*$ is a subgradient. ■

Proposition 1 implies that the following algorithm can be used to solve the unconstrained optimization problem [23]. The policy γ is initialized to be any feasible value γ_0. At each iteration m, the subgradient of f at point γ_m is computed as $y_m = -\lambda^*(\gamma_m) + \sum_{\boldsymbol{\alpha}, i, j=1}^{n} c_{ij}^{\boldsymbol{\alpha}}$, where $-\lambda^*(\gamma_m)$ is the subgradient of g at γ_m defined in Proposition 1 and

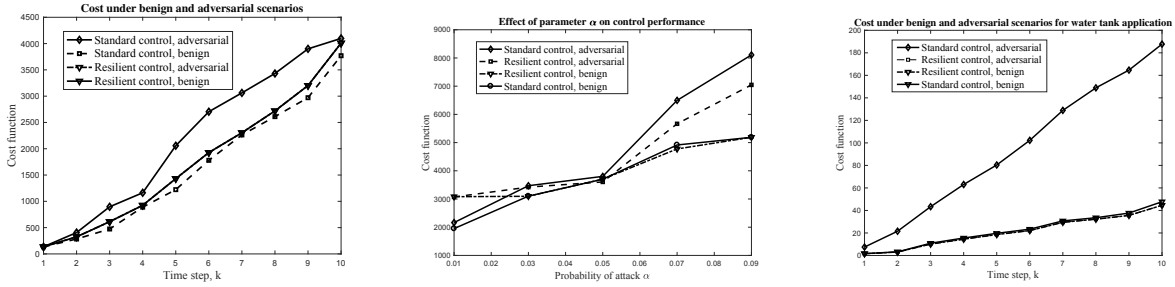
$$\begin{aligned} & c_{ij} = Pr(\boldsymbol{\alpha}) \\ & \cdot Pr((\mathbf{x} - \hat{\mathbf{x}}_k^{\boldsymbol{\alpha}})^T \Sigma_{k,\boldsymbol{\alpha}}^{-1} (\mathbf{x} - \hat{\mathbf{x}}_k^{\boldsymbol{\alpha}}) \in [\sigma_{i+1}, \sigma_i], \mathbf{x}^T \overline{Q} \mathbf{x} \in [\delta_i, \delta_{i+1}]). \end{aligned}$$

The value of $\gamma_{m+1} = \gamma_m - \epsilon_m y_m$, where $\epsilon_m > 0$ and $\sum_{k=1}^{\infty} \epsilon_m^2 < \infty$. The optimal solution γ^* is equal to

$$\gamma^* = \arg\min \{ f(\gamma) : \gamma \in \{ \gamma_0, \gamma_1, \ldots, \} \}.$$

A key step in this algorithm is computation of the coefficients $c_{ij}^{\boldsymbol{\alpha}}$. These coefficients can be computed by taking random samples of $\mathbf{x}$ conditioned on $\hat{\mathbf{z}}_{0:k}^{\boldsymbol{\alpha}}$.

The approach outlined in this section requires solving a convex program at each iteration in order to compute the sub-gradient $-\lambda^*$. While the computation time is polynomial in the number of states and inputs, as well as the number of discretized states N, the convergence rate may be slow. A faster heuristic with looser optimality bounds is described as follows.



(a) (b) (c)

Fig. 1. Numerical evaluation of proposed approach to resilient LQG control. (a) Comparison of the performance of our approach with standard LQG control in both benign and adversarial environments with Gaussian random system matrices and parameter $\boldsymbol{\alpha} = 0.01$. The resilient and standard LQG controllers provided the same cost function value under normal operating conditions, while the resilient controller provided significantly lower error compared to the standard LQG controller under false data injection attack. (b) Effect of parameter $Pr(\boldsymbol{\alpha})$, equal to probability of compromise, on the cost metric. The resilient control achieves the same performance as standard LQG control under benign operating conditions and outperforms the LQG control under adversarial conditions. (c) Effect of attack on a water tank filling case study. The resilient control provides comparable performance to the optimal control under benign operating conditions, and does not experience a loss of performance under false data injection attack.

Eq. (3) can be further bounded above by

$$J(\mu) \leq \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)})|\bar{\boldsymbol{\alpha}})Pr(\bar{\boldsymbol{\alpha}}) + \sum_{\boldsymbol{\alpha}} \big[Pr(\boldsymbol{\alpha}) \cdot \int_{\hat{\mathbf{z}}^{\boldsymbol{\alpha}}_{0:k}} \max_{\substack{\mathbf{u}_{k:(k+L)} \\ \in \mathcal{U}_\mu(\mathcal{I}^{\boldsymbol{\alpha}}_k)}} \int_{\mathbf{x}_k} \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)}))\ d\mathbf{x}_k\ d\hat{\mathbf{z}}^{\boldsymbol{\alpha}}_{0:k}\big]$$

Hence a sufficient condition is to select $\mathcal{U}_\mu(I^{\boldsymbol{\alpha}}_k)$ as

$$\{\mathbf{u}_{k:(k+L)} : \mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)})|\hat{\mathbf{z}}^{\boldsymbol{\alpha}}_{0:k}) \leq \gamma\}.$$

The constraint

$$\mathbf{E}(c(\mathbf{u}_{k:(k+L)}, \mathbf{x}_{k:(k+L)})|\hat{\mathbf{z}}^{\boldsymbol{\alpha}}_{0:k}) \leq \gamma$$

is quadratic in $\mathbf{u}_{k:(k+L)}$. Under this relaxation, the parameter γ is a scalar, and computing $g(\gamma)$ is equivalent to solving a quadratically-constrained quadratic program.

VI. Simulation

Our proposed approach was evaluated using Matlab. A system with $n = 10$ states, $m = 2$ inputs, and $p = 3$ outputs was simulated. The system matrices A, B, and C were generated randomly, with each entry selected as an independent $N(0, 1)$ Gaussian random variable. The matrix A was then normalized so that the set of maximum eigenvalue had magnitude 0.9. The matrices Q and R were also generated randomly with each entry as an independent $N(0, 1)$ Gaussian random variable. The same Q and R matrices were used at each time step. The noise vectors $\mathbf{w}_k$ and $\mathbf{v}_k$ had covariance matrices $\Sigma_w = \Sigma_v = I$. The set of compromised sensors $\boldsymbol{\alpha}$ was empty with probability 0.99 and equal to $\{3\}$ with probability 0.01.

Two control schemes were simulated. One scheme was our proposed resilient LQG control under the quadratic relaxation presented in Section V. The other simulated scheme was a standard LQG optimal control. Simulations occurred under both benign and adversarial settings. In the adversarial setting, the adversary followed a simple strategy of randomly generating a set of possible measurements to inject at each time step k, computing the control response to each measurement (via the adversary's knowledge of the control policy), and injecting the measurement that maximized the expected cost function $(\mathbf{x}_{k+1}Q\mathbf{x}_{k+1} + \mathbf{u}_{k+1}R\mathbf{u}_{k+1})$.

Figure 1(a) shows the system cost over time under the benign and adversarial scenarios as well as the standard LQG and resilient controllers. In order to ensure a fair comparison, the same $\mathbf{w}_k$ and $\mathbf{v}_k$ noise vectors were used at each time step. Each data point shows the summation

$$\sum_{l=0}^{k} (\mathbf{x}_l^T Q \mathbf{x}_l + \mathbf{u}_l^T R \mathbf{u}_l)$$

at time k. Under benign operating conditions, we found that our proposed approach provided roughly the same (within 1%) of the cost of the standard optimal LQG control. Under adversarial conditions, our proposed approach led to a 26% reduction in cost compared to optimal LQG control.

Figure 1(b) shows the impact of increasing $Pr(\boldsymbol{\alpha})$, equal to the probability of compromise, on the cost in both benign and adversarial environments. Increasing the probability of compromise increased the average cost for both the optimal LQG and resilient LQG methods, but the increase was slower for the resilient LQG algorithm.

We also evaluated our approach on an existing model of a set of interconnected water tanks, first proposed in [25]. The system is governed by the equations

$$\begin{aligned} \Delta\dot{L}1 &= -\frac{a_1}{A_1}\sqrt{\frac{g}{2L_{10}}}\Delta L_1 + \frac{K_p}{A_1}\Delta V_p \\ \Delta\dot{L}_2 &= \frac{a_1}{A_2}\sqrt{\frac{g}{2L_{10}}}\Delta L_1 - \frac{a_2}{A_2}\sqrt{\frac{g}{2L_{20}}}\Delta L_2 \end{aligned}$$

where ΔL_1 and ΔL_2 represent the deviation of the water tank levels from their equilibrium values, while ΔV_p is the input. The parameter values were chosen as $a_1 = a_2 = 0.178$, $A_1 = A_2 = 15.5$, $K_p = 2.775$, $g = 980$, $L_{20} = 1$, and $L_{10} = \frac{a_2^2}{a_1^2}L_{20}$, consistent with [25]. We discretized the system with sampling period $10ms$ and choose $Q = I_2$ and $R = 1$.

As in the randomly-generated systems, we found that the resilient approach provided similar performance with and without the false data injection attack, which was comparable to the standard LQG control when no attacker was present. In this case, the cost of the resilient control was one-fifth that of the standard control in the presence of an adversary.

VII. Conclusions and Future Work

This paper investigated Linear Quadratic Gaussian (LQG) control in the presence of false data injection attacks, in which the adversary has complete knowledge of the system state and control policy and attempts to maximize the quadratic cost function by introducing false inputs. We propose a two-stage solution approach. In the first stage, the system constructs a set of admissible control actions at each time step, in order to minimize the worst-case impact of an attack. The set of admissible actions is based on the prior sensor measurements. In the second stage, a single control action is chosen for each time step based on the current sensor measurement. We showed that both stages can be formulated as convex optimization problems and derived efficient controller design algorithms that exploit the convexity. Our approach was illustrated through a numerical study.

Future work will attempt to reduce the complexity of our approach to enable real-time control, as well as remove the discrete approximations required at the first stage, potentially using H_2 and H_∞ control techniques. We also plan to use the resilient LQG control as a basic building block for investigating more complex control problems, such as motion planning, in the presence of adversaries.

References

[1] J. Farrell, *Aided Navigation: GPS with High Rate Sensors*. McGraw-Hill, Inc., 2008.

[2] N. Diolaiti and C. Melchiorri, "Teleoperation of a mobile robot through haptic feedback," in *Haptic Virtual Environments and Their Applications, IEEE International Workshop 2002 HAVE*. IEEE, 2002, pp. 67–72.

[3] M. Kezunovic, S. Meliopoulos, V. Venkatasubramanian, and V. Vittal, *Application of Time-Synchronized Measurements in Power System Transmission Networks*. Springer, 2014.

[4] A. Perrig, J. Stankovic, and D. Wagner, "Security in wireless sensor networks," *Communications of the ACM*, vol. 47, no. 6, pp. 53–57, 2004.

[5] N. O. Tippenhauer, C. Pöpper, K. B. Rasmussen, and S. Capkun, "On the requirements for successful GPS spoofing attacks," in *Proceedings of the 18th ACM conference on Computer and communications security*. ACM, 2011, pp. 75–86.

[6] "Drone hijacking? that's just the start of GPS troubles," https://www.wired.com/2012/07/drone-hijacking/.

[7] D. P. Shepard, T. E. Humphreys, and A. A. Fansler, "Evaluation of the vulnerability of phasor measurement units to GPS spoofing attacks," *International Journal of Critical Infrastructure Protection*, vol. 5, no. 3, pp. 146–153, 2012.

[8] S. Amin, X. Litrico, S. Sastry, and A. M. Bayen, "Cyber security of water SCADA systemsPart I: Analysis and experimentation of stealthy deception attacks," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1963–1970, 2013.

[9] Y. Mo, E. Garone, A. Casavola, and B. Sinopoli, "False data injection attacks against state estimation in wireless sensor networks," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 5967–5972.

[10] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks against process control systems: risk assessment, detection, and response," in *Proceedings of the 6th ACM symposium on information, computer and communications security*. ACM, 2011, pp. 355–366.

[11] Y. Liu, P. Ning, and M. K. Reiter, "False data injection attacks against state estimation in electric power grids," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 1, p. 13, 2011.

[12] M. Pajic, J. Weimer, N. Bezzo, P. Tabuada, O. Sokolsky, I. Lee, and G. J. Pappas, "Robustness of attack-resilient state estimators," in *ICCPS'14: ACM/IEEE 5th International Conference on Cyber-Physical Systems (with CPS Week 2014)*. IEEE Computer Society, 2014, pp. 163–174.

[13] Y. Chen, S. Kar, and J. M. Moura, "Cyber physical attacks with control objectives," *IEEE Transactions on Automatic Control*, 2017.

[14] F. Pasqualetti, F. Dörfler, and F. Bullo, "Attack detection and identification in cyber-physical systems," *IEEE Transactions on Automatic Control*, vol. 58, no. 11, pp. 2715–2729, 2013.

[15] Y. Shoukry, P. Nuzzo, A. Puggelli, A. L. Sangiovanni-Vincentelli, S. A. Seshia, and P. Tabuada, "Secure state estimation for cyber physical systems under sensor attacks: a satisfiability modulo theory approach," *IEEE Transactions on Automatic Control*, 2017.

[16] D. P. Bertsekas, *Dynamic Programming and Optimal Control*. Athena Scientific Belmont, MA, 1995, vol. 1, no. 2.

[17] J.-Y. Keller and D. Sauter, "Monitoring of stealthy attack in networked control systems," in *IEEE Conference on Control and Fault-Tolerant Systems (SysTol)*, 2013, pp. 462–467.

[18] S. Mishra, Y. Shoukry, N. Karamchandani, S. Diggavi, and P. Tabuada, "Secure state estimation: optimal guarantees against sensor attacks in the presence of noise," in *IEEE International Symposium on Information Theory (ISIT)*, 2015, pp. 2929–2933.

[19] S. Z. Yong, M. Zhu, and E. Frazzoli, "Resilient state estimation against switching attacks on stochastic cyber-physical systems," in *IEEE 54th Annual Conference on Decision and Control (CDC)*, 2015, pp. 5162–5169.

[20] V. Ugrinovskii and C. Langbort, "Control over adversarial packet-dropping communication networks revisited," in *American Control Conference (ACC), 2014*. IEEE, 2014, pp. 3305–3309.

[21] S. Amin, A. A. Cárdenas, and S. Sastry, "Safe and secure networked control systems under denial-of-service attacks." in *Hybrid Systems: Computation and Control (HSCC)*, vol. 5469. Springer, 2009, pp. 31–45.

[22] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *47th Annual Allerton Conference on Communication, Control, and Computing*. IEEE, 2009, pp. 911–918.

[23] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[24] M. Morari and J. H. Lee, "Model predictive control: past, present and future," *Computers & Chemical Engineering*, vol. 23, no. 4, pp. 667–682, 1999.

[25] J. Araújo, A. Anta, M. Mazo, J. Faria, A. Hernandez, P. Tabuada, and K. H. Johansson, "Self-triggered control over wireless sensor and actuator networks," in *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*. IEEE, 2011, pp. 1–9.