Automatica 78 (2017) 194–201

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

A multi-channel transmission schedule for remote state estimation under DoS attacks[☆]

Kemi Ding[a], Yuzhe Li[a,1], Daniel E. Quevedo[b], Subhrakanti Dey[c], Ling Shi[a]

[a] Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong
[b] Department of Electrical Engineering, Paderborn University, Germany
[c] Department of Engineering Science, Uppsala University, Sweden

ARTICLE INFO

Article history:
Received 22 September 2015
Received in revised form
20 February 2016
Accepted 3 December 2016

Keywords:
Kalman filtering
Multi-channel networks
State estimation
DoS attack
Markov game
Dropout
Power control

ABSTRACT

This paper considers a cyber-physical system (CPS) under denial-of-service (DoS) attacks. The measurements of a sensor are transmitted to a remote estimator over a multi-channel network, which may be congested by a malicious attacker. Among these multiple communication paths with different characteristics and properties at each time step, the sensor needs to choose a single channel for sending data packets while reducing the probability of being attacked. In the meanwhile, the attacker needs to decide the target channel to jam under an energy budget constraint. To model this interactive decision-making process between the two sides, we formulate a two-player zero-sum stochastic game framework. A Nash Q-learning algorithm is proposed to tackle the computation complexity when solving the optimal strategies for both players. Numerical examples are provided to illustrate the obtained results.

1. Introduction

In order to enrich human–machine interactions in the physical/virtual world, various areas, such as aerospace, healthcare, transportation, civil infrastructure and smart grids, have increasingly adopted cyber-physical systems (CPSs) during recent years. A CPS is a system connecting the cyber world (e.g., information, communication) with the physical world by integrating computation, communication and control. In a CPS, multiple static/mobile sensors and actuators, which communicate with each other, interact with physical processes under the control of an intelligent decision system (Hespanha, Naghshtabrizi, & Xu, 2007). Traditionally, the sensing data is transmitted to the control centers through a single channel. However, this technique cannot provide reliable and timely communication in the competitive communication environment brought by distributed wireless sensor networks (WSNs) in CPSs as a result of limited bandwidth. A multi-channel network is an efficient technology to alleviate bursty communication traffic by transmitting the signals over several channels (Incel, 2011). Besides this, multi-channel networks can increase the resilience ability of a system under some rare events, e.g., severe weather or natural disasters (Proakis, 2001). Consequently, current MAC protocols, WSN hardware and commercially available radios for sensors support multi-channel communication. Hence, it is natural and important to consider multi-channel networks in the study of CPSs.

Although multi-channel techniques improve the throughput and capacity of the network, they cannot alleviate severe security challenges brought by the cyber components of the CPSs, especially vulnerabilities of wireless connection among sensors, estimators and actuators. Since CPSs are connected to many safety-critical infrastructure systems, attacks can lead to severe damage, as in the reported incident of the advanced computer worm Stuxnet infecting industrial control systems (Zhu & Bacsar, 2015). Therefore, careful defense-strategy design is essential for assuring the safe operation of CPSs in the face of adversarial attacks. In the present work, we mainly deal with DoS attacks (Li, Shi, Cheng, Chen, & Quevedo, 2015), which block the information flow

[☆] The work by L. Shi, Y. Li and K. Ding is supported by a HK RGC theme-based project T23-701/14N. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Valery Ugrinovskii under the direction of Editor Ian R. Petersen.

E-mail addresses: kdingaa@ust.hk (K. Ding), pku.ltracy@gmail.com (Y. Li), dquevedo@ieee.org (D.E. Quevedo), subhrakanti.dey@angstrom.uu.se (S. Dey), eesling@ust.hk (L. Shi).

[1] Fax: +852 2358 1485.

http://dx.doi.org/10.1016/j.automatica.2016.12.020
0005-1098/© 2016 Elsevier Ltd. All rights reserved.

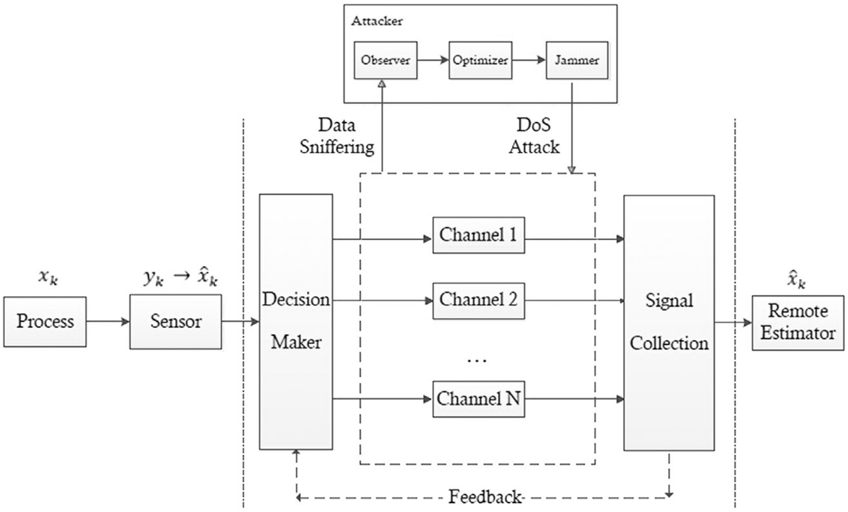


Fig. 1. Multi-channel remote estimation with an attacker.

between the sender (sensor) and the receiver to increase the packet-drop rate, and propose a defense-strategy design under DoS attacks for the multi-channel CPSs.

Energy limitation is inevitable (Shi, Cheng, & Chen, 2011) as sensing, computation and transmission power are restricted for sensor nodes and, moreover, replacing or recharging batteries may not always be possible in some WSNs. Consequently, a complicated security issue in CPSs (see Fig. 1) arises. To be more specific, facing DoS attacks launched by an intelligent attacker, the receiver (estimator) should choose a channel which is not just energy-efficient, but also dodges the attacks. Simultaneously, the smart attacker may notice the evasive actions taken by the sensor, and then modify its attack mode. Thus, an interaction between the sensor and the attacker occurs incessantly. The interaction can be captured by a game-theoretical approach, under which several recent studies about CPS security have been carried out, see Agah, Das, and Basu (2004), Li, Quevedo, Dey, and Shi (in press-a), Li, Quevedo, Dey, and Shi (in press-b), Liu, Liu, and El Saddik (2014), Song, Willett, Zhou, and Luh (2012) and Zhu and Bacsar (2015).

Based on cooperative game theory, the survey conducted by Agah et al. (2004) has proposed a new method for clustering sensors to provide a higher level of security. Instead of adopting model-based attacks to networks, in the formulation of Liu et al. (2014), an intelligent attacker with a dynamic and random attack strategy was considered in developing a two-player zero-sum stochastic game. Our recent work (Li et al., 2015) studied an interactive power scheduling between a sensor and a jammer, and used Markov chain theory to obtain the equilibrium solution. The recent work by Zhu and Bacsar (2015) introduced a cross-layer system design problem which results in solving a zero-sum differential game for robust control, coupled with a zero-sum stochastic game for a security policy in the absent of power constraints. In Li, Lai, and Qiu (2011), Li et al. used an ideal multi-channel system between a sensor and a remote estimator to avoid DoS attacks in a smart grid.

Compared with previous works, the main contributions of our current work are summarized as follows:

(1) With few studies undertaking an analysis of multi-channel networks for the secure state estimation problem in CPSs, this work sheds new light on the practical *multi-channel-based estimation issue* under malicious DoS attacks. In communication theory, the transmitted signal is typically assumed to be independent and identically distributed or at least stationary and ergodic; however, in this paper we consider a dynamical process and focus on improving estimation quality in an effective and defensive way.

(2) We investigate the iterative process between the sensor and the attacker by developing *a two-player zero-sum stochastic game*. By involving an elastic data arrival rate matrix to mathematically represent the relationship between the mixed strategies and the average packet loss rate, we obtain the reward function of the game. Moreover, by developing a Nash Q-learning algorithm, we acquire the optimal rational strategies for the sensor and the attacker.

(3) The stochastic game presented can be extended to include the channel choices and the power level selections, providing an important opportunity for the analysis of power allocation in multi-channel networks.

The remainder of the paper is organized as follows. Section 2 contains the mathematical models of the system. Section 3 demonstrates the framework of the stochastic game between the sensor and the attacker, while an algorithm is provided to obtain the rational strategies in Section 4. Some examples and concluding remarks are presented in Section 5 and Section 6.

Notations: $\mathbb{S}^n_+$ (or $\mathbb{S}^n_{++}$) is the set of n by n positive semi-definite matrices (or positive definite matrices). When $X \in \mathbb{S}^n_+$ (or $X \in \mathbb{S}^n_{++}$), we write $X \geq 0$ (or $X > 0$). $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. For functions h, g, $h \circ g$ is defined as the function composition $h(g(\cdot))$. $\mathbb{E}[\cdot]$ is the expectation of a random variable and $\Pr(\cdot)$ refers to the probability. The notation $\overrightarrow{\cdot}$ is the representation of vectors. The superscripts $'$ and $\star$ stand for the transposition and the optimal solution. The subscripts s and a refer to the sensor and the attacker, respectively.

2. Problem setup

In this section, we will introduce the mathematical models of the multi-channel structure depicted in Fig. 1, which is comprised of sensing, communication and estimation parts.

2.1. Process and sensor model

Consider the following linear time-invariant system:

$$x_{k+1} = Ax_k + w_k, \qquad y_k = Cx_k + v_k, \tag{1}$$

where $x_k \in \mathbb{R}^n$ is the system state vector, $y_k \in \mathbb{R}^m$ is the measurement taken by the sensor at time k, $w_k \in \mathbb{R}^n$ and $v_k \in \mathbb{R}^m$ are zero-mean i.i.d. Gaussian random noises with $\mathbb{E}[w_k w_j'] = \delta_{kj} Q$ ($Q \geq 0$), $\mathbb{E}[v_k v_j'] = \delta_{kj} R$ ($R > 0$), and $\mathbb{E}[w_k v_j'] = 0$ $\forall j, k$. The initial state x_0 is a zero-mean Gaussian random vector with covariance

$\Pi_0 \geq 0$, and which is uncorrelated with w_k and v_k. The pair (A, C) is assumed to be detectable and $(A, \sqrt{Q})$ is stabilizable.

In current CPSs, sensors are usually designed to be "smart" (Hovareshti, Gupta, & Baras, 2007) to improve the estimation/control performance of the system. Advanced embedded systems-on-chip render the improvement possible as they equip sensors with storage and computing capabilities, and they also allow sensors to process the collected information by executing some simple recursive algorithms. Thus, the sensor in Fig. 1, after taking measurements at time step k, runs a Kalman filter to estimate the state x_k of the process locally based on the overall collected measurements, instead of transmitting them to the remote estimator directly. The local minimum mean-squared error (MMSE) estimate of the process x_k is denoted by $\hat{x}_k^s = \mathbb{E}[x_k|y_0, \ldots, y_k]$. The corresponding estimation error e_k^s and the error covariance matrix P_k^s are defined as $e_k^s \triangleq x_k - \hat{x}_k^s$ and $P_k^s \triangleq \mathbb{E}[(e_k^s)(e_k^s)'|y_0, \ldots, y_k]$. These terms are computed via the Kalman filter and the iteration starts from $\hat{x}_0^s = 0$ and $P_0^s = \Pi_0$. For convenience, we define the Lyapunov and Riccati operators h and $\tilde{g} : \mathbb{S}_+^n \to \mathbb{S}_+^n$ as $h(X) \triangleq AXA' + Q$ and $\tilde{g}(X) \triangleq X - XC'[CXC' + R]^{-1}CX$. The error covariance P_k^s converges exponentially to a unique fixed point $\overline{P}$ of $h \circ \tilde{g}$ (Anderson & Moore, 1979). For simplicity, we ignore the transient periods and assume that the Kalman filter at the sensor has entered steady state; i.e., we assume that:

$$P_k^s = \overline{P}, \quad k \geq 1. \tag{2}$$

2.2. Multi-channel communication and attack model

After obtaining the state estimate $\hat{x}_k^s$, the sensor transmits this value as a data packet to the remote estimator through an N-channel communication network. We assume that all N channels have independent Additive White Gaussian Noise. In practice, the data packets may arrive unsuccessfully to the remote estimator because of noise, signal fading, etc. The packet arrival reliability measured in packet-error-rate (PER) is closely related to the signal-to-noise ratio (SNR) and the intrinsic channel characteristics (such as the channel coding method, the design of the receiver, etc.). Considering a DoS attack for a channel (in which an adversary can congest the communication network), the corresponding $SNR_i = \frac{p_s}{\sigma_i}$ is revised to be SINR (signal-to-interference-and-noise-ratio):

$$SINR_i = \frac{p_s}{p_a + \sigma_i}, \quad \text{if interfered,} \tag{3}$$

where the notations p_s and p_a correspond to the transmission power of the sensor and the interference power of the attacker, respectively, and σ_i is the additive white noise power of the ith channel. We assume that the channel gains are taken to be unity, therefore the definition of received SINR can be defined based on transmission powers instead of the actual received power.

In practice, different channel coding methods (such as cyclic redundancy check) will be used to detect the symbol error of the data before the decoder uploads the data to the receiver. As a result, some packets will be dropped if they contain errors. Furthermore, packet dropouts can also be caused by network congestion, timeout and other channel characteristics. This leads to the general form:

$$PER_i \triangleq \begin{cases} F(SINR_i), & \text{if the } i\text{th channel is attacked,} \\ F(SNR_i), & \text{otherwise.} \end{cases}$$
$$\beta_i = 1 - PER_i, \tag{4}$$

where β_i represents the packet arrival rate for the ith channel, and $F(\cdot)$ is a non-increasing function.

2.3. Remote state estimation

Let $\hat{x}_k$ be the MMSE estimate of the process x_k at the remote estimator, with corresponding error covariance P_k. The remote estimator obtains the state estimate as follows (Wu, Yuan, Zhang, & Shi, 2013): once receiving $\hat{x}_k^s$ successfully, the estimator synchronizes its estimate $\hat{x}_k$ with it; otherwise, the estimator simply predicts the estimate based on its previous estimate using the system model (1). To be precise,

$$\hat{x}_k = \begin{cases} \hat{x}_k^s, & \text{if data is received successfully,} \\ A\hat{x}_{k-1}, & \text{otherwise.} \end{cases} \tag{5}$$

As a result, the error covariance P_k at time k obeys

$$\begin{aligned} P_k &\triangleq \mathbb{E}[(x_k - \hat{x}_k)(x_k - \hat{x}_k)'] \\ &= \begin{cases} \overline{P}, & \text{if data is received successfully,} \\ h(P_{k-1}), & \text{otherwise.} \end{cases} \end{aligned} \tag{6}$$

Assume that the initial value of the error covariance at the remote estimator also starts from $\overline{P}$, that is, $P_0 = \overline{P}$. This corresponds to receiving a local state estimate (with estimation error covariance $\overline{P}$) at time $k = 0$. According to (6), P_k can only take values in the finite set $\{\overline{P}, h(\overline{P}), h^2(\overline{P}), \ldots, h^k(\overline{P})\}$ at a given time k.

2.4. Problem of interest

Given a tight budget of transmission and congestion power, the strategy design of the sensor or the attacker is important for efficient operation. In general, the task of the sensor is to make sure that its end user (i.e., the remote estimator) is sufficiently informed of the process, without wasting energy; and as for the attacker, it tends to disrupt the reliable communication between the sensor and the user, also without expending more effort than required. With opposite goals, the decision-making procedures of the sensor and the attacker are interactively linked. In Section 3, we will investigate the decision-making procedures for the defending sensor and the malicious attacker in a multi-channel network with power constraints and develop a game played over time to model this interactive process.

3. Stochastic game framework

Depending on the targets of the attacker, the information available to the defending sensor and the security mechanism, the adopted game may take different forms, such as a static game, Bayesian game, etc. (Leyton-Brown & Shoham, 2008). In order to improve their performance, it is more meaningful for both players to adopt dynamic cost-effective strategies based on the real-time information of the process rather than adopting static "offline" schemes. Motivated by this, we introduce a two-player stochastic game in this section to design the defense-attack strategies.

3.1. Stochastic game formulation

Define a stochastic game as $\mathcal{G} = \langle \mathcal{I}, \mathcal{S}, \mathcal{M}, \mathcal{P}, \mathcal{J} \rangle$, where $\mathcal{I}$ is the set of two players; $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ represents the action set for each player in $\mathcal{I}$; the term $\mathcal{M}$ denotes the state space of the game; the transition probability $p(m_{k+1}|m_k, s) \in \mathcal{P} : \mathcal{M}_k \times \mathcal{S} \to \mathcal{M}_{k+1}$ represents the probability of the next state being m_{k+1} given the current state m_k and the current action profile s; and the function $\mathcal{J} : \mathcal{M} \times \mathcal{S} \to \mathbb{R}^{\mathcal{I}}$ denotes the payoff of each player $i \in \mathcal{I}$. At time k, an action pair $s = (s_1, s_2)$ is chosen by the attacker and the sensor according to a mixed strategy pair (Leyton-Brown & Shoham, 2008). The joint actions affect the transition probability $p(m_{k+1}|m_k, s)$ for the next states and also incur a cost. In our problem setup, we have the following specifications.

Player The sensor $\mathcal{I}_s$ and the attacker $\mathcal{I}_a$ are assumed to be rational players; i.e., each of them makes the best choice in terms of their own preference among all available actions for them.

Strategy The game is played over time; namely, at every time k, the sensor needs to choose through which channel to send the data packets; the attacker faces the same situation, except with the hostile motivation to block a channel. Hence, the strategies of the sensor and the attacker, denoted by $(\overrightarrow{\gamma_k}, \overrightarrow{\lambda_k}) \in \mathcal{S}$, are composed of different probabilities of choosing corresponding channels. That is, $\overrightarrow{\gamma_k} = [Pr_0, Pr_1, \ldots, Pr_N]'$ and $\overrightarrow{\lambda_k} = [\widetilde{Pr_0}, \widetilde{Pr_1}, \ldots, \widetilde{Pr_N}]'$. Note that the extra term Pr_0 describes the likelihood of choosing the inactive state (no data is sent) by the sensor for energy saving; similarly for the attacker, $\widetilde{Pr_0}$ stands for launching no attack with some probability. The two collections of strategies are denoted by $\theta_s \triangleq \{\overrightarrow{\gamma}_1, \overrightarrow{\gamma}_2, \ldots\}$ and $\theta_a \triangleq \{\overrightarrow{\lambda}_1, \overrightarrow{\lambda}_2, \ldots, \}$ for the sensor and the attacker, respectively.

Remark 3.1. The representation of the probability for channel selection consists of pure/deterministic strategies (a pure strategy specifies a choice of *action*) and mixed/randomized strategies (a probability distribution over pure strategies). We are interested in the design of the strategies $\overrightarrow{\gamma}_k, \overrightarrow{\lambda_k}$ (i.e., the probability distribution) for the two players following from the action history.

State Define the state as the error covariance, i.e., $m_k \triangleq P_k$. Generally, the network uses acknowledgment (ACK) mechanisms for reliable data transfer; i.e., a feedback loop containing the control message from the receiver to the sender (see Fig. 1). The estimator explicitly informs the sensor whether the data packet is received successfully or not in an ACK. Thus, the sensor has full knowledge of the state. In the current work, the attacker is assumed to infer the state s_k as well by eavesdropping the ACKs.

Transition probability Note that the transition probability depends on the packet arrival rate in (4). To analyze this relationship under the multi-channel network, we introduce a matrix describing the packet arrival rate under different actions for each player and then link the game concept with the communication mode. Consider a time-varying matrix

$$M_k = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ \beta'_{10} & \beta'_{11} & \cdots & \beta'_{N1} \\ \vdots & \vdots & \ddots & \cdots \\ \beta'_{N0} & \beta'_{N1} & \cdots & \beta'_{NN} \end{pmatrix} \in \mathbb{R}^{(N+1)\times(N+1)}, \tag{7}$$

where the element $\beta'_{nm}, n, m \in \{0, 1, \ldots, N\}$ denotes the successful packet arrival rate when the sensor transmits data through the n-channel, and the attacker interferes the m-channel. Note that if the sensor decides not to send data, then the arrival rate $\beta'_{0m} = 0$. Moreover, based on (4), only the diagonal elements of the matrix M_k are functions of the *SINR*, that is,

$$\beta'_{nm} = \begin{cases} \beta_n(SINR_n), & \text{if } m = n, \\ \beta_n(SNR_n), & \text{if } m \neq n, \end{cases} \tag{8}$$

where $n, m \in \{1, \ldots, N\}$. We can obtain the expected packet arrival rate at time k via:

$$R_k(M_k, \overrightarrow{\lambda_k}, \overrightarrow{\gamma_k}) = \overrightarrow{\gamma_k}' M_k \overrightarrow{\lambda_k}. \tag{9}$$

Last, we can derive the transition probability as follows:

$$p(P_{k+1} \mid P_k, \overrightarrow{\lambda_k}, \overrightarrow{\gamma_k}, M_k) = \begin{cases} R_k, & \text{if } P_{k+1} = \overline{P}, \\ 1 - R_k, & \text{if } P_{k+1} = h(P_k), \\ 0, & \text{otherwise}. \end{cases} \tag{10}$$

Energy constraints One of the most important goals of network design is to have effective operations with limited energy. For simplicity, we fix the transmission power level (i.e., $\overrightarrow{E_s}$) and attack power level (i.e., $\overrightarrow{E_a}$) with respect to different channels: $\overrightarrow{E_s} = [e_0, e_1, \ldots, e_N]$ and $\overrightarrow{E_a} = [\tilde{e_0}, \tilde{e_1}, \ldots, \tilde{e_N}]$ (an extended form can be found in Section 3.3). Then, the expected transmission energy is denoted by $\overline{E_s} = \overrightarrow{E_s} \cdot \overrightarrow{\gamma_k}$ for the sensor, and $\overline{E_a} = \overrightarrow{E_a} \cdot \overrightarrow{\lambda_k}$ applied by the attacker.

Payoff According to Section 2.3, the stochastic game starts at the initial state $P_0 = \overline{P}$. Consequently, the game flow is defined as $g = \{P_0, s_0, P_1, s_1, \ldots\}$. A one-stage reward can be defined for each player based on the tradeoff between the average performance of the system at the next state and the consumed energy. The average performance is quantified by the expectation $\mathbb{E}(P_{k+1})$. This formulation depends on historical states and needs additional mathematical tools, which will be provided later.

With the sensor aiming to maximize the system performance without too much energy expenditure, the one-stage reward function for the sensor is given by

$$r_k(P_{k+1}, \overrightarrow{\gamma_k}, \overrightarrow{\lambda_k}) \triangleq -\mathrm{Tr}\,[\mathbb{E}(P_{k+1})] - \delta_s \frac{\overline{E_s}}{R_k} + \delta_a \frac{\overline{E_a}}{1 - R_k}, \tag{11}$$

where the function $r_k : \mathcal{M}_{k+1} \times \mathcal{S} \to \mathbb{R}$ is in an average sense and the parameters δ_a, δ_s represent the proportions of the energy term in the reward. Similarly, the reward function for the attacker here is taken as $-r_k$. Note that the reward depends on the energy efficiency (i.e., the terms $\delta_s \frac{\overline{E_s}}{R_k}$ and $\delta_a \frac{\overline{E_a}}{1-R_k}$ for the sensor and the attacker) rather than the actual energy, as often used in communication systems (Incel, 2011).

After obtaining a stream of reward functions $\{r_0, r_1, \ldots\}$, the payoff function for the sensor is obtained through a generalization from a Markov decision processes (MDP) to the infinite horizon stochastic game:

$$\mathcal{J}_S(\theta_s, \theta_a, P_0) = \sum_{k=0}^{+\infty} \rho^k r_k(P_k, \theta_s, \theta_a), \tag{12}$$

where the parameter $\rho \in [0, 1)$ stands for the discount factor. The payoff function of the attacker is opposite to that of the sensor. Thus, we have derived a two-player zero-sum stochastic game to interpret the interactive process between the sensor and the attacker.

Assumption 3.2. The sensor and the attacker adopt stationary strategies. Moreover, the transition probability and the reward function are stationary; i.e., independent of time k.

Remark 3.3. In other words, the probabilities over actions for the stationary strategies only depend on the state, i.e., $\overrightarrow{\gamma}_k \triangleq \overrightarrow{\gamma}_k(s_k)$ and $\overrightarrow{\lambda}_k \triangleq \overrightarrow{\lambda}_k(s_k)$. With the evolution of the system state, the strategies are different in every stage and the rational actions determined by the strategies may likewise be time-varying.

3.2. Markov chain model

To explain the intrinsic properties of the dynamics of the receiver state P_k, we introduce a Markov decision process, which enables us to characterize $\mathbb{E}(P_{k+1})$. To be precise, we define the chain state of the MDP to be exactly the error covariance P_k. Hence, at time k, the possible values of the error covariance P_k denoted by the state set $\mathbf{Z}_k$ are represented as $\mathbf{Z}_k = \{P_k : \overline{P}, h(\overline{P}), \dots, h^k(\overline{P})\}$. From (10), we describe the transitions from elements in the state set $\mathbf{Z}_k$ to values in $\mathbf{Z}_{k+1}$ via the matrix $\mathbf{T}_k$ with elements:

$$\mathbf{T}_k(i,j) \triangleq \mathbb{P}[\mathbf{Z}_{k+1} \mid \mathbf{Z}_k] = \begin{cases} R_k & \text{if } j = 1, \\ 1 - R_k & \text{if } j = i+1, \\ 0 & \text{otherwise}, \end{cases}$$

where $i \in \{1, 2, \dots, k\}$ and $j \in \{1, 2, \dots, k+1\}$ stand for the index of the possible values of the states in $\mathbf{Z}_k$ and $\mathbf{Z}_{k+1}$. Define a matrix Π, in which $\Pi(i,j)$ measures the probability with the state P_j at time j equals the possible values $h^i(\overline{P})$, and based on $\Pi(\cdot, j) = \Pi(\cdot, j-1)'\mathbf{T}_j$, we have

$$\Pi = \begin{pmatrix} R_1 & \cdots & R_k \\ 1 - R_1 & \cdots & (1-R_k)R_{k-1} \\ 0 & \cdots & (1-R_k)(1-R_{k-1})R_{k-2} \\ \vdots & \ddots & \vdots \\ 0 & \cdots & (1-R_k)(1-R_{k-1})\dots(1-R_1) \end{pmatrix} \tag{13}$$

where $i \in \{0, 1, \dots, k\}$ and $j \in \{1, 2, \dots, k\}$. Considering the probability distribution of P_k, i.e., the kth column of the matrix Π, we can evaluate the expected error covariance at time k as follows:

$$\mathrm{Tr}\,[\mathbb{E}(P_k)] = \mathrm{Tr}\left[\sum_{i=1}^{k+1} \Pi(i,k) h^{i-1}(\overline{P})\right], \quad \forall k \in \mathbb{Z}^+. \tag{14}$$

By now, we have an explicit expression for all the game elements. The solution of the optimal strategies for each player will be given in Section 4.2.

3.3. Extension to power-level selection

In our original formulation, only one power level was utilized for each channel. The above strategies for the sensor and the attacker can be easily extended to the case of multiple power levels. For example, if there are $l_i, i \in 1, \dots, N$ transmission power levels e_i^j for each channel, then the whole power level set is denoted by $\overrightarrow{E_s} = 0 \cup \{e_i^j : j \in \{1, 2, \dots, l_i\}, i \in \{1, 2, \dots, N\}\}$, covering the no-data-transmission mode. Respectively, we have $\overrightarrow{\gamma_k} = [Pr_0, Pr_1^1, \dots, Pr_i^j, \dots, Pr_N^{l_N}]'$ and $\overrightarrow{\lambda_k} = [\widetilde{Pr}_0, \widetilde{Pr}_1^1, \dots, \widetilde{Pr}_i^j,, \dots, \widetilde{Pr}_N^{l_N}]'$, where Pr_i^j represents the probability for the sensor to choose channel i and transmit data packets with power e_i^j. Hence, the pure strategy s for the sensor considers the $1 + \sum_{i=1}^N l_i$ power-level choices, which covers the $1 + N$ channel selections. A simple example for this setting will be illustrated in Section 5.2.

4. Design of rational strategies

In this section, we analyze the Nash equilibrium (NE) (i.e., a pair of optimal strategies) of the stochastic game derived in Section 3 and design the NE through a Nash Q-learning algorithm.

4.1. Equilibrium analysis

To solve the strategy-design problem, we first introduce the equilibria of the stochastic game. Then, we prove the existence of the optimal strategies in Theorem 4.2. Eventually, the strategy-design problem is equivalent to finding the NE of the stochastic game.

Definition 4.1 (*Nash Equilibrium* Leyton-Brown & Shoham, 2008). In the two-player zero-sum stochastic game with the initial state P_0 between the sensor and the attacker, a strategy profile $\pi_s^\star$ (or $\pi_a^\star$) for the sensor (or attacker) is a Nash equilibrium if no player can benefit from changing strategies while the other keeps its own unchanged, i.e., $\mathcal{J}_S^\star(P_0) \doteq \mathcal{J}_S(P_0, \pi_s^\star, \pi_a^\star) \geq \mathcal{J}_S(P_0, \pi_s, \pi_a^\star), \forall \pi_s$ and $\mathcal{J}_A^\star(P_0) \doteq \mathcal{J}_A(P_0, \pi_s^\star, \pi_a^\star) \geq \mathcal{J}_A(P_0, \pi_s^\star, \pi_a), \forall \pi_a$.

Theorem 4.2. *This sensor–attacker stochastic game between the sensor and the attacker has at least one NE.*

Proof. This game with discounted rewards has two players and $N+1$ pure strategies for each player. Moreover, from Assumption 3.2, players all deploy stationary strategies. The result follows from Leyton-Brown and Shoham (2008, Theorem 6.3.5). ■

4.2. Learning methodology

We now present a method to obtain the NE of the two-player zero-sum stochastic game. Denote the expected future payoffs (in which the players adopt the NE strategies from time $k+1$ onwards) for the sensor and the attacker as $\mathcal{J}_S^\star(m')$ and $\mathcal{J}_A^\star(m')$, where the notation $\star$ represents the payoff derivation using the NE strategies and the term m' stands for the next state. Then, based on (12), each player will play the following zero-sum single-decision game at every time k:

Problem 4.3. For the sensor,

$$\begin{aligned} \mathcal{J}_S^\star(m) = \max_{\overrightarrow{\gamma_k} \in \mathcal{S}} \Big(& r_k(m, \overrightarrow{\gamma_k}, \overrightarrow{\lambda_k^\star}) \\ & + \rho \sum_{m \in \mathcal{M}} p(m' | m, \overrightarrow{\gamma_k}, \overrightarrow{\lambda_k^\star}) \times \mathcal{J}_S^\star(m') \Big) \end{aligned}$$

$$\text{s.t.} \sum_{j=1}^{N+1} \gamma_{k,j} = 1, \qquad \sum_{j=1}^{N+1} \lambda_{k,j} = 1, \qquad \gamma_{k,j} \geqslant 0, \qquad \lambda_{k,j} \geqslant 0.$$

Problem 4.4. For the attacker,

$$\begin{aligned} \mathcal{J}_A^\star(m) = \max_{\overrightarrow{\gamma_k} \in \mathcal{S}} - \Bigg(& r_k(m, \overrightarrow{\gamma_k^\star}, \overrightarrow{\lambda_k}) \\ & + \rho \sum_{m \in \mathcal{M}} p(m' | m, \overrightarrow{\gamma_k^\star}, \overrightarrow{\lambda_k}) \times \mathcal{J}_S^\star(m') \Bigg) \end{aligned}$$

$$\text{s.t.} \sum_{j=1}^{N+1} \gamma_{k,j} = 1, \qquad \sum_{j=1}^{N+1} \lambda_{k,j} = 1,$$

$$\gamma_{k,j} \geqslant 0, \qquad \lambda_{k,j} \geqslant 0.$$

In the above, notations $\overrightarrow{\gamma_k^\star}, \overrightarrow{\lambda_k^\star}$ represent the actions determined by the NE action rules $\pi_s^\star, \pi_a^\star$ and can be obtained by solving the joint optimization problems. Unfortunately, there is no straightforward method for solving these problems in general because of the tight coupling between Problems 4.3 and 4.4. Classic algorithmic techniques for solving the stochastic game include *value iteration* and *strategy improvement* (Bertsekas, 2005), quadratic programming, etc. However, these algorithms require information of the reward for all states, which may be inaccessible for the sensor and the attacker. In order to overcome these disadvantages, we use the following Nash Q-learning algorithm (Hu & Wellman, 2003) to obtain the NE solution. Executing this algorithm just requires knowledge of the system parameters.

First, based on the objective function in Problems 4.3 and 4.4, we define Q-values for each player and denote the Q-value for the sensor as follows (the Q-value of the attacker is negative to that of the sensor):

$$\mathcal{Q}^\star(m, \overrightarrow{\gamma}, \overrightarrow{\lambda}) = r(m, \overrightarrow{\gamma}, \overrightarrow{\lambda}) + \rho[R(m, \pi_s^\star, \pi_a^\star) \times \mathcal{J}_S^\star(\overline{P}) \\ + (1 - R(m, \pi_s^\star, \pi_a^\star)) \times \mathcal{J}_S^\star(h(m))], \qquad (15)$$

where r is the one-stage reward defined in (11) with the initial state m and the joint actions $(\overrightarrow{\gamma}, \overrightarrow{\lambda})$. Note that from Assumption 3.2, the subscript k of the reward and strategies can be omitted. The value $\mathcal{Q}^\star$ combines the current reward and future reward when the two players play specified rational actions $\overrightarrow{\gamma}$ and $\overrightarrow{\lambda}$ according to the NE strategies from the next period onwards. Comparing (15) with Problem 4.3, we have

$$\mathcal{J}_S^\star(m) = \underset{\overrightarrow{\gamma}, \overrightarrow{\lambda}}{Nash}\ \mathcal{Q}^\star(m, \overrightarrow{\gamma}, \overrightarrow{\lambda}), \qquad (16)$$

where the notation *Nash* represents the process of finding the NE of the one-stage game with reward $\mathcal{Q}^\star$. Therefore, searching the NE of the stochastic game is translated from solving the joint optimization problem defined in Problems 4.3 and 4.4 to finding the NE of a single stage game with reward $\mathcal{Q}^\star$.

Next, we develop a learning process to calculate $\mathcal{Q}^\star$ through repeated plays, in which each player updates its Q-value by using the Q-value information of itself and its opponent from an arbitrary guess at the beginning. The updated equation of the Q-value for the sensor is thus defined as follows:

$$\mathcal{Q}_{k+1}(m, \overrightarrow{\lambda_k}, \overrightarrow{\gamma_k}) = (1 - \alpha_k)\mathcal{Q}_k(m, \overrightarrow{\lambda_k}, \overrightarrow{\gamma_k}) \\ + \alpha_k(r_k + \rho Nash\ \mathcal{Q}_k(m')), \qquad (17)$$

where $Nash\ \mathcal{Q}_k(m')$ is as in (16), α_k (or $\alpha_k(m, \overrightarrow{\gamma_k}, \overrightarrow{\lambda_k})$) is the learning rate to be designed and it is related to the state m and the actions. From the standard formulation of a two-player zero-sum single stage game, $Nash\ \mathcal{Q}_k(m')$ can be represented in the following min–max form and can be easily solved using linear programming:

$$Nash\ \mathcal{Q}_k(m') \triangleq \mathcal{J}_S^\star(m') \\ = \max_{\pi_s} \min_{\pi_a} \sum_{\overrightarrow{\gamma_k}, \overrightarrow{\lambda_k}} \mathcal{Q}_k(m', \overrightarrow{\gamma_k}, \overrightarrow{\lambda_k})\pi_s(m', \overrightarrow{\gamma_k})\pi_a(m', \overrightarrow{\lambda_k}). \qquad (18)$$

Finally, with a large number of repeated plays and given the randomness of adopted actions in every step, the Q-value $\mathcal{Q}_k(m, \overrightarrow{\gamma_k}, \overrightarrow{\lambda_k})$ may converge to $\mathcal{Q}^\star$ (see Theorem 4.5). The optimal strategies for the stochastic game can be obtained by (16). The generalized version of the Nash Q-learning algorithm is given in **Algorithm 1**. Note that $\|\cdot\|$ represents the matrix norm and δ is the accuracy requirement.

Algorithm 1 Nash Q-learning algorithm

```
Initialization:
$k = 0$ and set the initial state $m \in \mathcal{M}$
Initialize the Q-value $\mathcal{Q}_k^i(m, \overrightarrow{\lambda_k}, \overrightarrow{\gamma_k})$ for all states $m$ and
  arbitrary $\overrightarrow{\lambda_k}, \overrightarrow{\gamma_k}$, where $i = 1, 2$ represents the two players
  respectively
While $||\mathcal{Q}_{k+1} - \mathcal{Q}_k|| < \delta$
Find the NE Q-value $Nash\ \mathcal{Q}(m')$ based on (18) and the
  corresponding optimal mixed strategies
Randomly select actions for the two players based on the
  optimal mixed strategy profiles
Observe the next state $m'$ and update the Q-value in (17)
$k := k + 1$
End
```

Theorem 4.5 (*Convergence Conditions*)**.** *The Nash Q-learning process (i.e., the iteration defined in (17)) will converge to $\mathcal{Q}^\star$ with probability* 1, *if*

(1) *Every state $m \in \mathcal{M}$ with different actions $(\overrightarrow{\gamma_k}, \overrightarrow{\lambda_k}) \in \mathcal{S}$ is visited infinitely often during the learning process.*
(2) *The learning rate α_k satisfies: $\alpha_k \in [0, 1)$, $\sum_{k=0}^{\infty} \alpha_k = \infty$ and $\sum_{k=0}^{\infty} \alpha_k^2 < \infty$; $\alpha_k(m, \overrightarrow{\gamma}, \overrightarrow{\lambda}) \neq 0$ if $(m, \overrightarrow{\gamma}, \overrightarrow{\lambda})$.*
(3) *A saddle point exists in every stage game for all times k and states m, and each player uses the NE payoffs to update their Q-values.*

The proof of Theorem 4.5 is similar to those in Hu and Wellman (2003). The two former conditions can be satisfied according to Remark 4.6. Due to the existence of a saddle point in any two-player zero-sum stage game, condition (3) is always satisfied.

Remark 4.6. Condition (1) indicates that as long as every state and action is successfully traversed, the convergence result is independent of the random actions of each player during the learning process. Through a large number of iterations and random actions in the learning process, Condition (1) can be satisfied. In Condition (2), the first term restricts the convergence of the learning rate, whereas the second states that the players only update the Q-values corresponding to the current time and state. To satisfy these conditions, we design the learning rate as a non-zero decreasing function of time k and the current state and actions, which is illustrated in the simulation part (see Section 5). It is reasonable to decrease the learning rate because the learning process starts with little information, while after many steps, sufficient information is available for the sensor. This consequently reduces the proportion of the learning term in the updated equation (17).

We now outline stability issues of the estimation process (when the game is in equilibrium) in terms of the asymptotic state estimation error covariance. Based on (6) and the stationarity assumption of the error covariance (which was proved in Kar, Sinopoli, & Moura, 2012), we have $\lim_{T\to\infty} \sup \frac{1}{T} \sum_{k=1}^{T} \mathrm{Tr}[\mathbb{E}(P_k)] = \lim_{k\to\infty} \sup \mathrm{Tr}[\mathbb{E}(P_k)] = \sum_{i=1}^{+\infty} \pi_i \mathrm{Tr}[h^{i-1}(\overline{P})]$, $\sum_{i=1}^{+\infty} \pi_i = 1$ where $\pi_i \triangleq \mathbb{P}r(P_{+\infty} = h^i(\overline{P})) = \prod_{j=0}^{i-1}(1 - \epsilon_j)\epsilon_i$ and $\epsilon_i = 1 - \overrightarrow{\gamma_k^\star}'(m = h^i(\overline{P}))M_k\overrightarrow{\lambda_k^\star}(m = h^i(\overline{P}))$ represents the arrival rate when the NE strategies are adopted. Based on properties of the Lyapunov operator $h(X) = AXA' + Q$ (Quevedo, Ahlén, Leong, & Dey, 2012), a sufficient condition for the estimation stability is $\min_{i=0,\ldots,K} \epsilon_i > \rho(A)^{-2}$, where $\rho(A)$ stands for the spectral radius of A. Note that we only consider the first $K + 1$ rates ϵ_i since there is a final state $h^K(\overline{P})$[2] which represents $h^i(\overline{P})$ with $i \geq K$ during the learning process and the strategies used for these states are the same as those of the state $h^K(\overline{P})$.

5. Simulations

In this section, we will illustrate the results using some examples. We consider the following vector system with parameters $A = \begin{pmatrix} 1 & 0.5 \\ 0 & 1.05 \end{pmatrix}$, $C = \begin{pmatrix} 1 & 0 \end{pmatrix}$, $Q = \begin{pmatrix} 0.5 & 0 \\ 0 & 0.5 \end{pmatrix}$ and $R = 0.5$, where the steady-state error covariance is $\overline{P} = \begin{pmatrix} 0.38 & 0.28 \\ 0.28 & 1.69 \end{pmatrix}$. Assuming that the channels in the examples are wireless fast-fading channels, we adopt the general form of $F(\cdot)$ in (4), i.e., $F(x) = cx^{-L_i}$, where c, L_i are constants related to the channel characteristics. In the examples, we have two channels with different characteristics, $L_1 = 2, L_2 = 1$, but with the same noise parameters, $\sigma_i = 0.1,\ i = 1, 2$.

[2] Asymptotically, this finite state approximation is valid as the probabilities of the remaining states decay exponentially fast from (13).

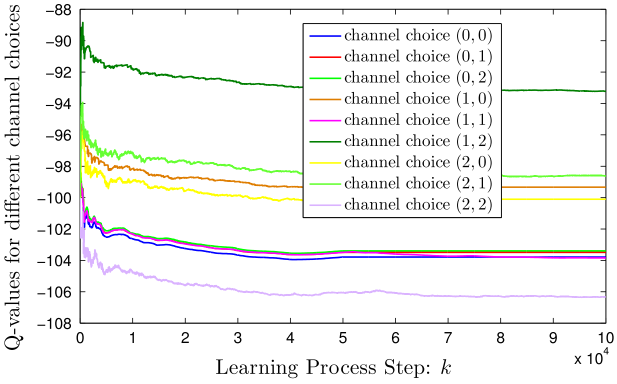


Fig. 2. Q-value in state $\overline{P}$ for all joint channel choices.

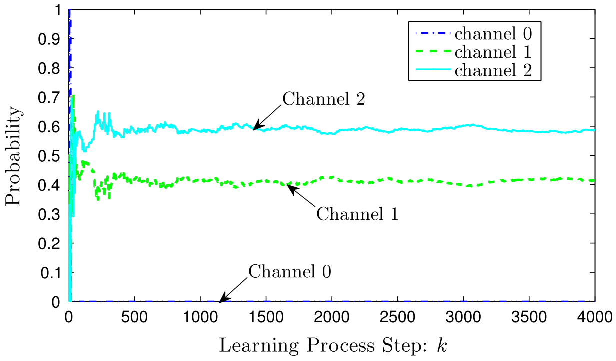


Fig. 3. Strategy learning process for the sensor in state $\overline{P}$.

5.1. Multi-channel strategies

For these two channels, we design the transmission energy levels $p_s = (0.5\ 0.6)'$ for the sensor and the attack energy levels $p_a = (0.3\ 0.3)'$ for the attacker, respectively. The other parameter settings for the reward function and the learning process are $\delta_s = 1$, $\delta_a = 1$, $\rho = 0.96$ and $\alpha_t = 10/[15 + count(m, \overrightarrow{\gamma_t}, \overrightarrow{\lambda_t})]$, where *count* is a function to calculate the occurrence of the pair $(m, \overrightarrow{\gamma_t}, \overrightarrow{\lambda_t})$. Before the learning process, to reduce the computational burden, we impose the restriction that the states set $\mathcal{M}$ is finite and equals $\{\overline{P}, \dots, h^4(\overline{P})\}$. We apply the Nash Q-learning algorithm 100 000 times (about 10 min using MATLAB) and obtain the following results (see Fig. 2):

- **Result 1** After 100 000 learning steps, we can see that the Q-values in different states all converge. Moreover, state $\overline{P}$ is more likely to be achieved. We take state $\overline{P}$ as an example, in which the converged Q-values are concluded in Fig. 2.
- **Result 2** We observe that not only do the Q-values converge, but the channel-choice probabilities also tend to be stable for each player under different states (for example, see Fig. 3). Hence, we can obtain the stationary strategies for the sensor and the attacker under different states. We summarize the optimal strategies of each player under states $\overline{P}$ and $h(\overline{P})$ in Table 1.
- **Result 3** Based on the NE strategies shown in Table 1, we compare $\mathrm{Tr}(P_k)$ and its averaged form $\frac{1}{k}\sum_{l=0}^{k}\mathrm{Tr}(P_l)$ under the defensive and the non-defensive case (in the latter the players choose channels following a uniform distribution). The corresponding simulation result is given in Fig. 4. It is easy to conclude that the defensive strategy provides a more accurate state estimate.

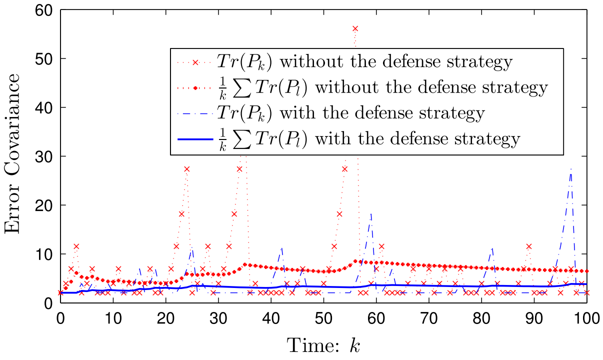


Fig. 4. Comparison between the defense and no-defense case.

Table 1
Optimal strategies in states $\overline{P}$ and $h(\overline{P})$.

State		Channel		
		0th	1th	2th
$\overline{P}$	Sensor	0	0.42	0.58
	Attacker	0	0.71	0.29
$h(\overline{P})$	Sensor	0	0.44	0.56
	Attacker	0	0.63	0.37

Table 2
Settings and the optimal strategies.

Channel		0-ch	1-ch		2-ch	
Sensor	Energy	0	0.2	0.5	0.2	0.6
	Probability	0	0.48	0.37	0.15	0
Attacker	Energy	0	0.1	0.3	0.05	0.3
	Probability	0.76	0.06	0.17	0	0.01

5.2. Multi-channel strategies and power-level selection

We consider a system with the same parameters as the previous one, but as foreshadowed in Section 3.3, add new energy levels for the two channels; i.e., the total transmission power levels deployed by the sensor are $[0.3\ 0.5]'$ and $[0.3\ 0.6]'$, respectively, and the attacker power levels are $[0.15\ 0.3]'$ and $[0.1\ 0.2]'$. With extra energy levels, the pure strategy set for each player becomes the combination of the channel choice and power assignment. Applying the Nash Q-learning algorithm, we have similar conclusions as in the previous example. The Q-values also converge and the optimal strategies for each player are illustrated in Table 2 (also taking state $\overline{P}$ as an example).

6. Conclusion

This work has investigated security issues in CPSs with a multi-channel network, where a malicious agent can launch DoS attacks by jamming communication channels between the sensor and the remote estimator. Taking energy limitations into consideration, a two-player zero-sum stochastic game model was formulated and a Nash Q-learning method was proposed to solve the Nash equilibrium. As this work is limited to the case of a game between one sensor and one attacker, more research involving multiple sensors or multiple attackers can be done.

References

Agah, A., Das, S. K., & Basu, K. (2004). A game theory based approach for security in wireless sensor networks. In *Proc. IEEE conf. performance, computing, and communications* (pp. 259–263).

Anderson, B., & Moore, J. B. (1979). *Optimal filtering, Vol. 1*. (p. 1979). Englewood Cliffs, NJ: Prentice-Hall.

Bertsekas, D. P. (2005). *Dynamic programming and optimal control, Vol. 1*. Athena Scientific.

Hespanha, J., Naghshtabrizi, P., & Xu, Y. (2007). A survey of recent results in networked control systems. *Proceedings of the IEEE*, *95*(1), 138–162.

Hovareshti, P., Gupta, V., & Baras, J. S. (2007). Sensor scheduling using smart sensors. In *Proc. IEEE conf. decision and control* (pp. 494–499).

Hu, J., & Wellman, M. P. (2003). Nash q-learning for general-sum stochastic games. *The Journal of Machine Learning Research*, *4*, 1039–1069.

Incel, O. D. (2011). A survey on multi-channel communication in wireless sensor networks. *Computer Networks*, *55*(13), 3081–3099.

Kar, S., Sinopoli, B., & Moura, J. M. (2012). Kalman filtering with intermittent observations: Weak convergence to a stationary distribution. *IEEE Transactions on Automatic Control*, *57*(2), 405–420.

Leyton-Brown, K., & Shoham, Y. (2008). *Essentials of game theory: A concise multidisciplinary introduction*. Morgan & Claypool Publishers.

Li, H., Lai, L., & Qiu, R. (2011). A denial-of-service jamming game for remote state monitoring in smart grid. In *Proc. IEEE conf. information sciences and systems* (pp. 1–6).

Li, Y., Quevedo, D. E., Dey, S., & Shi, L. (2016). A game-theoretic approach to fake-acknowledgment attack on cyber-physical systems. *IEEE Transactions on Signal and Information Processing Over Networks*, http://dx.doi.org/10.1109/TSIPN.2016.2611446, (in press-a).

Li, Y., Quevedo, D. E., Dey, S., & Shi, L. (2016). SINR-based DoS attack on remote state estimation: A game-theoretic approach. *IEEE Transactions on Control of Network Systems*, http://dx.doi.org/10.1109/TCNS.2016.2549640, (in press-b).

Li, Y., Shi, L., Cheng, P., Chen, J., & Quevedo, D. (2015). Jamming attacks on remote state estimation in cyber-physical systems: A game-theoretic approach. *IEEE Transactions on Automatic Control*, *60*(10), 2831–2836.

Liu, S., Liu, P. X., & El Saddik, A. (2014). A stochastic game approach to the security issue of networked control systems under jamming attacks. *Journal of the Franklin Institute*, *351*(9), 4570–4583.

Proakis, J. (2001). *Ser. McGraw-Hill series in electrical and computer engineering: communications and signal processing*, *Digital communications*. McGraw-Hill.

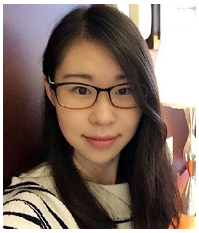
Quevedo, D. E., Ahlén, A., Leong, A. S., & Dey, S. (2012). On Kalman filtering over fading wireless channels with controlled transmission powers. *Automatica*, *48*(7), 1306–1316.

Shi, L., Cheng, P., & Chen, J. (2011). Sensor data scheduling for optimal state estimation with communication energy constraint. *Automatica*, *47*(8), 1693–1698.

Song, X., Willett, P., Zhou, S., & Luh, P. (2012). The MIMO radar and jammer games. *IEEE Transactions on Signal Processing*, *60*(2), 687–699.

Wu, J., Yuan, Y., Zhang, H., & Shi, L. (2013). How can online schedules improve communication and estimation tradeoff? *IEEE Transactions on Signal Processing*, *61*(7), 1625–1631.

Zhu, Q., & Bacsar, T. (2015). Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: Games-in-games principle for optimal cross-layer resilient control systems. *IEEE Transactions on Control Systems*, *35*(1), 46–65.



Kemi Ding received the B.S. degree in Electronic and Information Engineering from Huazhong University of Science and Technology, Wuhan, China, in 2014. She is currently pursuing a Ph.D. degree in the Department of Electronic and Computer Engineering at the Hong Kong University of Science and Technology, Hong Kong, China. From September 2016 to December 2016, she is a visiting student in the School of Engineering and Applied Sciences (SEAS) in Harvard University. Her current research interests include cyber-physical system security/privacy, networked state estimation and wireless sensor networks.

Yuzhe Li received the B.S. degree in Mechanics and B.A. degree in Economics from Peking University, China in 2011 and the Ph.D. degree in Electronic and Computer Engineering from the Hong Kong University of Science and Technology, Hong Kong in 2015. Between June 2013 and August 2013, he was a visiting scholar in the University of Newcastle, Australia. He is currently a Postdoctoral Fellow at the Department of Electrical and Computer Engineering, University of Alberta, Canada. His current research interests include cyber-physical system security, sensor power control and networked state estimation.

Daniel E. Quevedo holds the Chair of Automatic Control (Regelungs- und Automatisierungstechnik) at Paderborn University, Germany. He received Ingeniero Civil Electrónico and M.Sc. degrees from the Universidad Técnica Federico Santa María, Chile, in 2000. In 2005, he was awarded the Ph.D. degree from the University of Newcastle in Australia.

Dr. Quevedo was supported by a full scholarship from the alumni association during his time at the Universidad Técnica Federico Santa María and received several university-wide prizes upon graduating. He received the IEEE Conference on Decision and Control Best Student Paper Award in 2003 and was also a finalist in 2002. In 2009 he was awarded a five-year Research Fellowship from the Australian Research Council.

Prof. Quevedo is Associate Editor of the IEEE Control Systems Magazine, Editor of the International Journal of Robust and Nonlinear Control, Steering Committee Member of the IEEE Internet of Things Initiative, and serves as Chair of the IEEE Control Systems Society Technical Committee on Networks & Communication Systems. His research interests are in control of networked systems and of power converters.

Subhrakanti Dey received the B.Tech. and M.Tech. degrees from Indian Institute of Technology, Kharagpur, in 1991 and 1993, respectively, and the Ph.D. degree from Australian National University, Canberra, in 1996. He is currently a Professor with the Dept of Engineering Sciences in Uppsala University, Sweden. His current research interests include networked control systems, wireless communications and networks, signal processing for sensor networks, and stochastic and adaptive signal processing and control. Professor Dey currently serves on the Editorial Board of IEEE Transactions on Signal Processing and IEEE Transactions on Control of Network Systems. He was also an Associate Editor for the IEEE Transactions on Signal Processing during 2007–2010 and the IEEE Transactions on Automatic Control during 2004–2007, and Associate Editor for Elsevier Systems and Control Letters during 2003–2013.

Ling Shi received the B.S. degree in electrical and electronic engineering from Hong Kong University of Science and Technology, Kowloon, Hong Kong, in 2002 and the Ph.D. degree in Control and Dynamical Systems from California Institute of Technology, Pasadena, CA, USA, in 2008. He is currently an associate professor at the Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology. His research interests include cyber-physical systems security, networked control systems, sensor scheduling, and event-based state estimation. He has been serving as a subject editor for International Journal of Robust and Nonlinear Control from March 2015 and an associate editor for IEEE Transactions on Control of Network Systems from July 2016. He also served as an associate editor for a special issue on Secure Control of Cyber-physical Systems in the IEEE Transactions on Control of Network Systems in 2015–2016.