Secure Sensor Design for Cyber-Physical Systems Against Advanced Persistent Threats

Muhammed O. Sayin(✉) and Tamer Başar

Coordinated Science Laboratory, University of Illinois at Urbana-Champaign,
Urbana, IL 61801, USA
{sayin2,basar1}@illinois.edu

Abstract. We introduce a new paradigm to the field of control theory: "secure sensor design". Particularly, we design sensor outputs cautiously against advanced persistent threats that can intervene in cyber-physical systems. Such threats are designed for the very specific target systems and seeking to achieve their malicious goals in the long term while avoiding intrusion detection. Since such attacks can avoid detection mechanisms, the controller of the system could have already been intervened in by an adversary. Disregarding such a possibility and disclosing information without caution can have severe consequences. Therefore, through secure sensor design, we seek to minimize the damage of such undetected attacks in cyber-physical systems while impacting the ordinary operations of the system at minimum. We, specifically, consider a controlled Markov-Gaussian process, where a sensor observes the state of the system and discloses information to a controller that can have friendly or adversarial intentions. We show that sensor outputs that are memoryless and linear in the state of the system can be optimal, in the sense of game-theoretic hierarchical equilibrium, within the general class of strategies. We also provide a semi-definite programming based algorithm to design the secure sensor outputs numerically.

Keywords: Stackelberg games · Stochastic control · Cyber-physical systems · Security · Advanced persistent threats · Sensor design · Semi-definite programming

1 Introduction

A cyber-physical system can be considered as a system equipped with sensing and actuation capabilities in the physical part, and monitoring or controlling capabilities using computer-based algorithms in the cyber part, e.g., process control systems, robotics, smart grid, and autonomous vehicles [9]. However, due to the cyber part, such systems are very prone to cyber-attacks. Reference [10] reveals such vulnerabilities of the inner vehicle networks to cyber attacks

This research was supported by the U.S. Office of Naval Research (ONR) MURI grant N00014-16-1-2710.

S. Rass et al. (Eds.): GameSec 2017, LNCS 10575, pp. 91–111, 2017.
DOI: 10.1007/978-3-319-68711-7_6

experimentally, e.g., an attacker has been able to control the brake system of a moving vehicle remotely. In 2010, StuxNet worm targeted very specifically certain supervisory control and data acquisition (SCADA) systems and managed to cause substantial damage, which was an eye-opener pointing to insufficiency of the existing, isolation based, security mechanisms for such systems [8]. Recently in 2014, Dragonfly Malware infiltrated into the cyber-physical systems across the energy and pharmaceutical industries and intervened in the systems over a long period of time stealthily [16]. In a nutshell, those experiences show that once an adversarial attacker infiltrates into the cyber part of the system, he/she can monitor and control the physical processes away from the system's desired target, which can lead to severe consequences. Therefore, developing novel formal security mechanisms plays a vital role in the security of these systems.

Existing studies mainly focus on characterizing the vulnerabilities of cyber-physical systems against various attack models. Reference [14] formulates necessary conditions for an undetected attack that can cause unbounded error in the state estimation. In [18], the authors characterize necessary and sufficient conditions for an undetected attack when the system does not have any sensor and process noises. In [5,6], the authors formulate the optimal cyber-attacks with control objectives, where the attacker both seeks to be undetected and drive the state of the systems according to his/her adversarial goals by manipulating sensor outputs and control inputs together. Recently, Reference [20] has analyzed the optimal attack strategies seeking to increase the quadratic cost of a system with linear Gaussian dynamics, while maintaining certain degree of stealthiness.

There are also studies that aim to provide formal security guarantees against false data injection attacks, where attackers infiltrate into a subset of multiple sensors and report false outputs into the system. In order to detect and recover from such attacks, Reference [7] provides a security mechanism for estimation and control based applications, and in [13], the authors propose a coding scheme for the outputs of multiple sensors. Apart from these two separate approaches, i.e., analyzing optimal attacks with control objectives and encoding outputs of multiple sensors against false data injection attacks, we aim to combine them together in the secure sensor design framework. Particularly, closed-loop control is essential in cyber-physical systems due to the uncertainty of the state noise, i.e., a controller needs the sensor outputs to be able to drive the state toward his/her desired path [11]. By designing sensor outputs in advance, we seek to provide security against the attacks with control objectives.

Economics also plays an essential role while developing defense strategies for cyber-security of systems [4]. As an example, investment on security measures should not exceed the value of the protected asset. Furthermore, adversarial attacks are also costly and an attack would be feasible, therefore expected, if the attack costs the attacker less than the damage at the target. Therefore reducing the damage that can be caused by such threats as much as possible is crucial to reduce the feasibility, therefore the likelihood, of such attacks. To this end, in the secure sensor design framework, we seek to minimize the damage by the attacks, with minimum impact on the ordinary operations of the system.

We propose a new approach for the security of cyber-physical systems by minimizing the damage of cyber-attacks on the system. We focus on undetectable, or difficult to detect, attacks, which we call "advanced persistent threats". These attacks are advanced by targeting very specific systems with knowledge about the underlying dynamics, and persistent by attacking stealthily, i.e., avoiding detection mechanisms. Since such attackers can intervene in the system for a long period of time without being detected, this rises the possibility of adversarial intervention in cyber part of the systems at any time. Therefore, the system designer should take such possibilities into consideration. However, the designer should also not take precautions as if the cyber part of the system is compromised due to such a possibility since that would impact the intended operations of the system substantially. In particular, there is a trade-off between securing the system and maintaining a certain performance in the system.

In this paper, to obtain explicit results, we specifically consider systems with linear quadratic Gaussian dynamics and control objectives, which have various applications in industry [20] from manufacturing processes to aerospace control. We consider the possibility for adversarial interventions in the controller by advanced persistent threats, and seek to design sensor outputs cautiously in advance. Therefore, there is a hierarchical structure between the sensor and the controller of the system. The controller constructs a closed-loop control input based on the sensor output, knowing the relationship between the sensor output and the state. Furthermore, if the controller is an adversary, then the objectives of the sensor and the controller mismatch. Therefore, we can analyze the interactions between the sensor and the controller through a game-theoretic hierarchical equilibrium, which implies that, as a sensor designer, we should anticipate the controller's reaction by also taking into account that the controller can have both friendly or adversarial objectives. We show that for controlled Markov-Gaussian processes, the equilibrium achieving sensor outputs are memoryless and linear in the underlying state of the system. Additionally, we provide a semi-definite programming (SDP) based algorithm to design secure sensor outputs numerically.

The main contributions of this paper are as follows:

- This appears to be the first work in the literature to study sensor design against advanced persistent threats that can infiltrate into the controller of a cyber-physical system.
- We provide a formal problem formulation from a game-theoretical perspective to design sensor outputs cautiously due to the possibility of undetected interventions in the controllers.
- Given any sensor strategies, we compute the optimal control strategies for both friendly and adversarial objectives. Note that the adversary seeks to construct control inputs that are close to the control inputs that would have been constructed if he/she had a friendly objective in order to avoid detection and accomplish his/her malicious goals in the long term over the time horizon by exploiting the uncertainties in the system.
- We show that the optimal sensor strategies in the sense of game-theoretic hierarchical equilibrium are memoryless and linear in the underlying state.

Correspondingly, friendly as well as adversarial control strategies are linear in the sensor outputs.
- We also provide a practical algorithm to design secure sensors numerically.

The paper is organized as follows: In Sect. 2, we provide the secure sensor design framework. In Sect. 3, we formulate the associated multi-stage static Bayesian Stackelberg game. In Sect. 4, we characterize the optimal controller response strategies for given sensor strategies. We compute the corresponding optimal sensor strategies in Sect. 5. We conclude the paper in Sect. 6 with several remarks and possible research directions. An Appendix A includes proof of a technical result.

Notations: For an index-ordered set of variables, e.g., $x_1, \cdots, x_n$, we define $x_{[k,l]} := x_k, \cdots, x_l$, where $1 \leq k \leq l \leq n$. $\mathbb{N}(0, .)$ denotes the multivariate Gaussian distribution with zero mean and designated covariance. We denote random variables by bold lower case letters, e.g., $\boldsymbol{x}$. For a vector x and a matrix A, x' and A' denote their transposes, respectively, and $\|x\|$ denotes the Euclidean (L^2) norm of the vector x. For a matrix A, $\text{tr}\{A\}$ denotes its trace. We denote the identity and zero matrices with the associated dimensions by I and O, respectively. For positive semi-definite matrices A and B, $A \succeq B$ means that $A - B$ is also a positive semi-definite matrix.

2 Problem Formulation

Consider a controlled stochastic system [11] described by the following state equation:

$$\boldsymbol{x}_{k+1} = A\boldsymbol{x}_k + B\boldsymbol{u}_k + \boldsymbol{v}_k,\ k = 1, 2, \ldots, n, \tag{1}$$

where[1] $A \in \mathbb{R}^{m\times m}$, $B \in \mathbb{R}^{m\times r}$, $\boldsymbol{x}_1 \sim \mathbb{N}(0, \Sigma_1)$. The additive noise sequence $\{\boldsymbol{v}_k\}$ is a white Gaussian vector process, i.e., $\boldsymbol{v}_k \sim \mathbb{N}(0, \Sigma_v)$, and is independent of the initial state $\boldsymbol{x}_1$. The closed loop control vector $\boldsymbol{u}_k \in \mathbb{R}^r$ is given by

$$\boldsymbol{u}_k = \gamma_k(\boldsymbol{s}_{[1,k]}), \tag{2}$$

where $\gamma_k(\cdot)$ can also be any Borel measurable function from $\mathbb{R}^{mk}$ to $\mathbb{R}^r$, and $\boldsymbol{s}_k \in \mathbb{R}^m$ is the sensor output, which is given by

$$\boldsymbol{s}_k = \eta_k(\boldsymbol{x}_{[1,k]}), \tag{3}$$

where $\eta_k(\cdot)$ can be any Borel measurable function from $\mathbb{R}^{mk}$ to $\mathbb{R}^m$.

As seen in Fig. 1, we have two non-cooperating agents: Sensor (S) and Controller (C). C can be a friend or an adversary while S does not know C's type. Only S has access to the state $\boldsymbol{x}_k$ and can construct sensor output $\boldsymbol{s}_k$. C observes $\boldsymbol{s}_k$, knows S's strategy $\eta_k(\cdot)$ due to a hierarchy between the agents, and, by using $\boldsymbol{s}_{[1,k]}$, can construct a closed loop control input $\boldsymbol{u}_k$, which cannot be monitored by the system.

[1] Even though we consider time invariant matrices A and B for notational simplicity, the provided results could also be extended to time-variant cases.

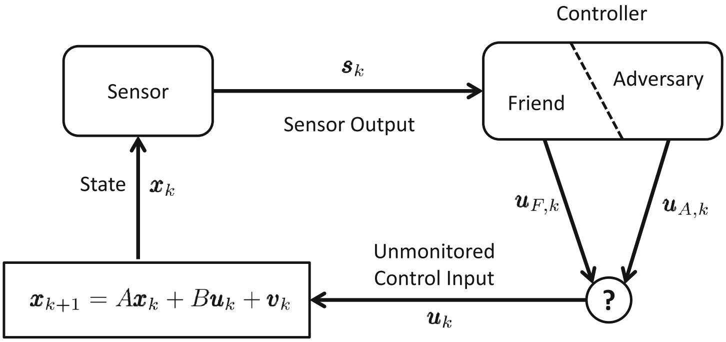

Fig. 1. Cyber physical system including a sensor and a controller.

Remark 1. A hierarchy between the agents is a reasonable assumption in control system design since sensors are designed and implemented in advance, and system engineers design the controllers knowing the relation between the sensor output and the underlying state.

The agents S and C construct $\boldsymbol{s}_k$ and $\boldsymbol{u}_k$ according to their own objectives. In particular, S chooses $\eta_k(\cdot)$ from the strategy space Υ_k, which, for each k, is the set of all Borel measurable functions from $\mathbb{R}^{mk}$ to $\mathbb{R}^m$, i.e., $\eta_k \in \Upsilon_k$ and $\boldsymbol{s}_k = \eta_k(\boldsymbol{x}_{[1,k]})$. C chooses $\gamma_k(\cdot)$ from the strategy space Γ_k, which is the set of all Borel measurable functions from $\mathbb{R}^{mk}$ to $\mathbb{R}^r$, i.e., $\gamma_k \in \Gamma_k$ and $\boldsymbol{u}_k = \gamma_k(\boldsymbol{s}_{[1,k]})$.

Normally, in a stochastic control scenario [11], S and C would have a common finite horizon[2] quadratic loss function

$$L(\boldsymbol{x}_{[2,n+1]}, \boldsymbol{s}_{[1,n]}, \boldsymbol{u}_{[1,n]}) = \sum_{k=1}^{n} \|\boldsymbol{x}_{k+1}\|^2_{Q_{k+1}} + \|\boldsymbol{u}_k\|^2_{R_k}, \tag{4}$$

where $Q_{k+1} \in \mathbb{R}^{m\times m}$ is positive semi-definite and $R_k \in \mathbb{R}^{r\times r}$ is positive definite. Then, S would disclose the state directly so that C could drive the state in their commonly desired path [11,12]. However, in a cyber physical system, the system is vulnerable against adversarial attacks that seek to drive the state of the system away from the system's desired target. We call such attacks "advanced persistent threats", which are advanced by being designed very specifically for the targeted system, i.e., the attacker knows, or can learn stealthily, the underlying state recursion, and persistent by avoiding intrusion detection. Therefore, S, i.e., the sensor designer, should anticipate the likelihood of adversarial intrusions into C, i.e., the possibility that C can be an adversary, and construct $\boldsymbol{s}_k$ accordingly.

We denote the set of all adversarial objectives by Ω, the appropriate σ-algebra on Ω by F, and the probability distribution over Ω by $\mathbf{P}$. In particular, we have the probability space $(\Omega, \mathsf{F}, \mathbf{P})$. And for a point $\omega \in \Omega$ drawn from Ω according

[2] E.g., horizon length is n.

to $\mathbf{P}$, the adversarial loss function is given by

$$L_A(\omega, \boldsymbol{x}_{[2,n+1]}, \boldsymbol{s}_{[1,n]}, \boldsymbol{u}_{[1,n]}) = \sum_{k=1}^{n} \|\boldsymbol{x}_{k+1} - z(\omega)\|^2_{Q_{A,k+1}(\omega)} + \|\boldsymbol{u}_{A,k} - \boldsymbol{u}^*_{F,k}\|^2_{R_{A,k}(\omega)}, \tag{5}$$

where $\boldsymbol{u}_{A,k}$, $k = 1, \ldots, n$, denotes the adversarial action, $z : (\Omega, \mathsf{F}) \to (\mathbb{R}^m, \mathsf{B}^m)$ is an $(\mathsf{F}, \mathsf{B}^m)$ measurable function[3], $Q_{A,k+1} : (\Omega, \mathsf{F}) \to (\mathbb{R}^{m\times m}, \mathsf{B}^{m\times m})$ is an $(\mathsf{F}, \mathsf{B}^{m\times m})$ measurable function such that $Q_{A,k+1}(\omega) \in \mathbb{R}^{m\times m}$ is positive semi-definite, and $R_{A,k} : (\Omega, \mathsf{F}) \to (\mathbb{R}^{r\times r}, \mathsf{B}^{r\times r})$ is an $(\mathsf{F}, \mathsf{B}^{r\times r})$ measurable function such that $R_{A,k}(\omega) \in \mathbb{R}^{r\times r}$ is positive definite. Here, for each $\omega \in \Omega$, $z(\omega)$ denotes the desired state that the adversary seeks to drive the system to, and $\boldsymbol{u}^*_{F,k}$ is the optimal action that would have been taken if C was a friend so that the adversary can avoid intrusion detection by being close to $\boldsymbol{u}^*_{F,k}$. We further assume that $z(\omega)$ is a second-order random vector.

Remark 2. We note that if the control inputs could have been monitored, then any deviation of the control input from the optimal control input of a friend type C could have been detected instantly.

3 A Multi-stage Static Bayesian Stackelberg Game

In order to model undetected adversarial interventions, let $\boldsymbol{\theta}$ be a Bernoulli random variable, with a commonly known p, corresponding to the likelihood of C being an adversary, i.e., $\mathbb{P}\{\boldsymbol{\theta} = 1\} = p$, and $\boldsymbol{\theta} = 1$ if C is an adversary. Since the type of C is not known by S, we can consider this incomplete information scenario as an imperfect information scenario [15]; in which Nature moves first, draws a realization of $\boldsymbol{\theta}$, then if the realization $\theta = 1$, also draws $\omega \in \Omega$, and reveals these only to C.

Furthermore, the multiple interactions between non-cooperating S and C can be considered as a *multi-stage* game [1]. Since S's actions $\boldsymbol{s}_{[1,n]}$ do not depend on C's actions $\boldsymbol{u}_{[1,n]}$, i.e., S cannot update his/her strategies after observing $\boldsymbol{u}_{[1,n]}$, this is a multi-stage *static* game. The underlying state recursion is common knowledge of both S and C (even if C can be an adversary). The type of C and, if C is an adversary, his/her objective are not known by S. However, S knows the probability space $(\Omega, \mathsf{F}, \mathbf{P})$ and p, which implies that this is a multi-stage static *Bayesian* game. There is also a hierarchy [1,17] between the agents in the announcement of the strategies such that S leads the game by announcing and sticking to his/her strategies in advance, i.e., C knows $\eta_{[1,n]}$ in advance. Therefore, we can model such a scheme as a multi-stage static Bayesian *Stackelberg* game, in which S is the leader.

Remark 3. Once any adversarial intrusion has been detected due to C's anomalous behavior through external defense mechanisms, this multi-stage static

[3] B^m denotes the Borel σ-algebra on $\mathbb{R}^m$.

Bayesian Stackelberg game terminates since the uncertainty about C's type is removed. The reaction of the system after the detection is beyond this paper's scope. Therefore, we consider that the game continues over the horizon and continuation of the game implies that any adversarial intervention has not been detected while the possibility of undetected adversarial intervention still exists.

Remark 4. Even though the attacker can also inject false data into the sensor outputs in order to avoid detection as in integrity attacks, e.g., [5,6], the attacker still needs the actual sensor outputs, which are designed by the system designer in advance, in order to construct the optimal control input according to his/her objective. Therefore, secure sensor design framework also plays a crucial role for the security of the systems against integrity attacks.

S and C aim to minimize their expected loss functions through the actions $\boldsymbol{s}_{[1,n]}$ and $\boldsymbol{u}_{[1,n]}$ by choosing the strategies $\eta_{[1,n]}$ and $\gamma_{[1,n]}$ accordingly. Given the realizations of S's actions, i.e., $s_{[1,k]}$, C constructs the control input $u_{F,k}$ or $u_{A,k}$ depending on his/her type, which not only depends on $s_{[1,k]}$, but also the associated strategies $\eta_{[1,k]}$. In order to show this dependence explicitly, we denote C's strategies by $\boldsymbol{u}_{F,k} = \gamma_{F,k}(\boldsymbol{s}_{[1,k]};\eta_{[1,k]})$ instead of $\gamma_{F,k}(\boldsymbol{s}_{[1,k]})$ if C is a friend, or $\boldsymbol{u}_{A,k} = \gamma_{A,k}(\omega,\boldsymbol{s}_{[1,k]};\eta_{[1,k]})$ instead of $\gamma_{A,k}(\omega,\boldsymbol{s}_{[1,k]})$ if C is an adversary. Furthermore, given S's strategies $\eta_{[1,n]}$, we let $\Pi_F(\eta_{[1,n]}), \Pi_A(\omega,\eta_{[1,n]}) \subset \mathbb{R}^{r\times n}$ be C's reaction set. And these reaction sets are given by:

$$\Pi_F(\eta_{[1,n]}) := \underset{\substack{u_{F,k}\in\mathbb{R}^r \\ k=1,\ldots,n}}{\operatorname{argmin}} \ \mathbb{E}\{L(\boldsymbol{x}_{[2,n+1]},\boldsymbol{s}_{[1,n]},\boldsymbol{u}_{F,[1,n]})\},$$

$$\Pi_A(\omega,\eta_{[1,n]}) := \underset{\substack{u_{A,k}\in\mathbb{R}^r \\ k=1,\ldots,n}}{\operatorname{argmin}} \ \mathbb{E}\{L_A(\omega,\boldsymbol{x}_{[2,n+1]},\boldsymbol{s}_{[1,n]},\boldsymbol{u}_{A,[1,n]})\},$$

where $\mathbb{E}$ denotes the expectation taken over $\{\boldsymbol{x}_1,\boldsymbol{v}_{[1,n]}\}$. Due to the positive definiteness assumptions on R_k and $R_{A,k}(\omega)$, for all $\omega\in\Omega$, L and L_A are strictly convex in C's actions $\boldsymbol{u}_{F,[1,n]},\boldsymbol{u}_{A,[1,n]}$. This implies that the corresponding reaction sets Π_F and Π_A are singletons and the best C actions $\boldsymbol{u}^*_{F,k},\boldsymbol{u}^*_{A,k}$ are unique.

Corresponding to the loss functions L and L_A, depending on the agents' actions $\boldsymbol{s}_k$ and $\boldsymbol{u}_k$, there exist certain cost functions depending on the agents' strategies: $J(\eta_{[1,n]},\gamma_{[1,n]})$ and $J_A(\omega,\eta_{[1,n]},\gamma_{[1,n]})$, while each strategy implicitly depends on the other. Therefore let $\tilde{\Pi}_F$ and $\tilde{\Pi}_A$ be the sets of best C strategies, as subsets of $\times_{k=1}^n \Gamma_k$:

$$\tilde{\Pi}_F(\eta_{[1,n]}) := \underset{\substack{\gamma_{F,k}\in\Gamma_k \\ k=1,\ldots,n}}{\operatorname{argmin}} \ J(\eta_{[1,n]},\gamma_{F,[1,n]}),$$

$$\tilde{\Pi}_A(\omega,\eta_{[1,n]}) := \underset{\substack{\gamma_{A,k}(\omega,\cdot)\in\Gamma_k \\ k=1,\ldots,n}}{\operatorname{argmin}} \ J_A(\omega,\eta_{[1,n]},\gamma_{A,[1,n]}),$$

which are equivalence classes such that $\forall\ \gamma^*_{F,[1,n]} \in \tilde{\Pi}_F$ (or $\forall\ \gamma^*_{A,[1,n]} \in \tilde{\Pi}_A$), we have $\boldsymbol{u}^*_{F,k} = \gamma^*_{F,k}(\boldsymbol{s}_{[1,k]}; \eta_{[1;k]})$ (or $\boldsymbol{u}^*_{A,k} = \gamma^*_{A,k}(\omega, \boldsymbol{s}_{[1,k]}; \eta_{[1;k]})$). Therefore, the pair of strategies $\left[\eta^*_{[1,n]}; (\gamma^*_{F,[1,n]}, \gamma^*_{A,[1,n]})\right]$ attains the Stackelberg equilibrium provided that

$$\eta^*_{[1,n]} = \underset{\substack{\eta_k \in \Upsilon_k,\\ k=1,\dots,n}}{\operatorname{argmin}}\ (1-p) J\big(\eta_{[1,n]}, \gamma^*_{F,[1,n]}(\cdot; \eta_{[1,n]})\big)$$
$$+\ p \int_{\Omega} J\big(\eta_{[1,n]}, \gamma^*_{A,[1,n]}(\omega, \cdot; \eta_{[1,n]})\big) \mathbf{P}(d\omega) \tag{6a}$$

$$\gamma^*_{F,[1,n]}(\cdot; \eta_{[1,n]}) = \underset{\substack{\gamma_{F,k} \in \Gamma_k,\\ k=1,\dots,n}}{\operatorname{argmin}}\ J\big(\eta_{[1,n]}, \gamma_{F,[1,n]}(\cdot; \eta_{[1,n]})\big), \tag{6b}$$

$$\gamma^*_{A,[1,n]}(\omega, \cdot; \eta_{[1,n]}) = \underset{\substack{\gamma_{A,k}(\omega,\cdot) \in \Gamma_k,\\ k=1,\dots,n}}{\operatorname{argmin}}\ J_A\big(\omega, \eta_{[1,n]}, \gamma_{A,[1,n]}(\omega, \cdot; \eta_{[1,n]})\big). \tag{6c}$$

In the following sections, we analyze these equilibrium achieving strategies, i.e., $\left[\eta^*_{[1,n]}; (\gamma^*_{F,[1,n]}, \gamma^*_{A,[1,n]})\right]$.

4 Optimal Follower (Controller) Reactions

By (4), for a given $\boldsymbol{s}_{[1,n]}$, the friendly C also seeks to minimize

$$\sum_{k=1}^{n} \mathbb{E}\left\{ \|\boldsymbol{x}_{k+1}\|^2_{Q_{k+1}} + \|\boldsymbol{u}_k\|^2_{R_k} \right\}, \tag{7}$$

over $\gamma_{F,k} \in \Gamma_k$, $k = 1, \dots, n$, such that $\boldsymbol{u}_{F,k} = \gamma_{F,k}(\boldsymbol{s}_{[1,k]})$ subject to (1)–(3). In order to facilitate the subsequent analysis, in the following, we rewrite the state equations (1)–(2) and the expected loss function (7) without altering the optimization problem.

Lemma 1. *The friendly objective* (7) *is equivalent to:*

$$\min_{\substack{\gamma_{F,k} \in \Gamma_k\\ k=1,\dots,n}} \sum_{k=1}^{n} \mathbb{E}\|\boldsymbol{u}_{F,k} + K_k \boldsymbol{x}_k\|^2_{\Delta_k} + G, \tag{8}$$

where

$$K_k = \Delta_k^{-1} B'_k \tilde{Q}_{k+1} A \tag{9a}$$

$$\Delta_k = B' \tilde{Q}_{k+1} B + R_k \tag{9b}$$

$$G = \operatorname{tr}\{\Sigma_1 \tilde{Q}_1\} + \sum_{k=1}^{n} \operatorname{tr}\{\Sigma_v \tilde{Q}_{k+1}\} \tag{9c}$$

and $\{\tilde{Q}_k\}$ *is a sequence defined through the following discrete-time Riccati equation:*

$$\tilde{Q}_{k+1} = Q_k + A'\left(\tilde{Q}_{k+1} - \tilde{Q}_{k+1}B\Delta_k^{-1}B'\tilde{Q}_{k+1}\right)A, \tag{10a}$$

$$\tilde{Q}_{n+1} = Q_{n+1} \text{ and } Q_1 = O. \tag{10b}$$

Proof. This follows from the extensively used "completing the squares" technique [2,11]. □

Note that in (8), $\boldsymbol{x}_k$ depends on the previous control inputs $\boldsymbol{u}_{[1,k-1]}$. Through a change of variables [2], the friendly C's objective (8) can be written as

$$\min_{\substack{\gamma_{F,k}\in\Gamma_k\\ k=1,\dots,n}} \sum_{k=1}^{n} \mathbb{E}\|\boldsymbol{u}_{F,k}^o + K_k\boldsymbol{x}_k^o\|_{\Delta_k}^2 + G \tag{11}$$

subject to (9)–(10) and

$$\boldsymbol{x}_{k+1}^o = A\boldsymbol{x}_k^o + \boldsymbol{v}_k,\ k = 1,\dots,n, \text{ and } \boldsymbol{x}_1^o = \boldsymbol{x}_1, \tag{12a}$$

$$\boldsymbol{u}_{F,k}^o = \boldsymbol{u}_{F,k} + K_kB\boldsymbol{u}_{F,k-1} + K_kAB\boldsymbol{u}_{F,k-2} + \cdots + K_kA^{k-2}B\boldsymbol{u}_{F,1}. \tag{12b}$$

Note also that, now, the process $\{\boldsymbol{x}_k^o\}$ is independent of the control inputs $\boldsymbol{u}_{F,k}$ (and $\boldsymbol{u}_{F,k}^o$). Therefore, by (11), given the sensor outputs $\boldsymbol{s}_{[1,k]} = s_{[1,k]}$, the optimal *transformed* control input $u_{F,k}^{o*}$ (12b) is given by

$$u_{F,k}^{o*} = -K_k\mathbb{E}\{\boldsymbol{x}_k^o|\boldsymbol{s}_{[1,k]} = s_{[1,k]}\},$$

which implies

$$\boldsymbol{u}_{F,k}^{o*} = -K_k\mathbb{E}\{\boldsymbol{x}_k^o|\boldsymbol{s}_{[1,k]}\} \tag{13}$$

almost everywhere on $\mathbb{R}^r$. By (12b), we have

$$\underbrace{\begin{bmatrix} \boldsymbol{u}_{F,n}^o \\ \boldsymbol{u}_{F,n-1}^o \\ \vdots \\ \boldsymbol{u}_{F,1}^o \end{bmatrix}}_{=:\,\boldsymbol{u}^o} = \underbrace{\begin{bmatrix} I & K_nB & \cdots & K_nA^{n-2}B \\ & I & \cdots & K_{n-1}A^{n-3}B \\ & & \ddots & \vdots \\ & & & I \end{bmatrix}}_{=:\,\Phi} \underbrace{\begin{bmatrix} \boldsymbol{u}_{F,n} \\ \boldsymbol{u}_{F,n-1} \\ \vdots \\ \boldsymbol{u}_{F,1} \end{bmatrix}}_{=:\,\boldsymbol{u}},$$

which can also be written as $\boldsymbol{u}_F^o = \Phi\boldsymbol{u}_F$. And (13) leads to

$$\boldsymbol{u}_F^{o*} = -\underbrace{\begin{bmatrix} K_n & & \\ & \ddots & \\ & & K_1 \end{bmatrix}}_{=:\,K} \underbrace{\begin{bmatrix} \mathbb{E}\{\boldsymbol{x}_n^o|\boldsymbol{s}_{[1,n]}\} \\ \vdots \\ \mathbb{E}\{\boldsymbol{x}_1^o|\boldsymbol{s}_1\} \end{bmatrix}}_{=:\,\underline{\hat{\boldsymbol{x}}}^o}, \tag{14}$$

which yields that the actual optimal control inputs are given by

$$\boxed{\boldsymbol{u}_F^* = -\Phi^{-1}K\ \underline{\hat{\boldsymbol{x}}}^o.} \tag{15}$$

While the friendly C has the same objective (4) with S, by (5), for each $\omega \in \Omega$, the adversarial C's objective is to minimize

$$\sum_{k=1}^{n} \mathbb{E}\left\{\|\boldsymbol{x}_{k+1} - z(\omega)\|^2_{Q_{A,k+1}(\omega)} + \|\boldsymbol{u}_{A,k} - \boldsymbol{u}^*_{F,k}\|^2_{R_{A,k}(\omega)}\right\}, \tag{16}$$

over $\gamma_{A,k}(\omega,\cdot) \in \Gamma_k$, $k = 1,\dots,n$, such that $\boldsymbol{u}_{A,k} = \gamma_{A,k}(\omega, \boldsymbol{s}_{[1,k]})$ subject to (1)–(3). Next, we aim to rewrite the state equations and the expected loss functions as in Lemma 1 and (11) for the minimization of the adversarial objective.

Let $\delta\boldsymbol{u}_k := \boldsymbol{u}_{A,k} - \boldsymbol{u}^*_{F,k}$ and instead of (1), consider the following recursion:

$$\begin{bmatrix} \boldsymbol{x}_{k+1} \\ \hdashline \boldsymbol{u}^*_F \\ z(\omega) \end{bmatrix} = \underbrace{\left[\begin{array}{c;{2pt/2pt}c} A & \cdots B \cdots \\ \hdashline O & I \end{array}\right]}_{=:\, \bar{A}} \underbrace{\begin{bmatrix} \boldsymbol{x}_k \\ \hdashline \boldsymbol{u}^*_F \\ z(\omega) \end{bmatrix}}_{= \bar{\boldsymbol{x}}_k} + \underbrace{\begin{bmatrix} B \\ \hdashline O \end{bmatrix}}_{=:\, \bar{B}} \delta\boldsymbol{u}_k + \underbrace{\begin{bmatrix} I \\ \hdashline O \end{bmatrix}}_{=:\, E} \boldsymbol{v}_k,$$

which can also be written as

$$\bar{\boldsymbol{x}}_{k+1} = \bar{A}\bar{\boldsymbol{x}}_k + \bar{B}\,\delta\boldsymbol{u}_k + E\boldsymbol{v}_k. \tag{17}$$

Correspondingly, the objective can be rewritten as

$$\sum_{k=1}^{n} \mathbb{E}\left\{\|\bar{\boldsymbol{x}}_{k+1}\|^2_{\bar{Q}_{A,k+1}(\omega)} + \|\delta\boldsymbol{u}_k\|^2_{R_{A,k}(\omega)}\right\}, \tag{18}$$

where

$$\begin{aligned} \bar{Q}_{A,k+1}(\omega) &:= \left[\begin{smallmatrix} I \\ O \\ -I \end{smallmatrix}\right] Q_{A,k+1}(\omega) \left[\begin{smallmatrix} I & O & -I \end{smallmatrix}\right] \\ &= \left[\begin{smallmatrix} Q_{A,k+1}(\omega) & O & -Q_{A,k+1}(\omega) \\ O & O & O \\ -Q_{A,k+1}(\omega) & O & Q_{A,k+1}(\omega) \end{smallmatrix}\right]. \end{aligned}$$

We point out the resemblance between (7) and (18). Therefore, by Lemma 1 and (11), we have the following transformations:

Lemma 2. *The adversary's objective* (18) *is equivalent to:*

$$\min_{\substack{\gamma_{A,k}(\omega,\cdot)\in\Gamma_k \\ k=1,\dots,n}} \sum_{k=1}^{n} \mathbb{E}\|\delta\boldsymbol{u}_k + K_{A,k}(\omega)\bar{\boldsymbol{x}}_k\|^2_{\Delta_{A,k}(\omega)} + G_A(\omega), \tag{19}$$

where

$$K_{A,k}(\omega) = \Delta_{A,k}(\omega)^{-1}\bar{B}'\tilde{Q}_{A,k+1}(\omega)\bar{A} \tag{20a}$$

$$\Delta_{A,k}(\omega) = \bar{B}'\tilde{Q}_{A,k+1}(\omega)\bar{B} + R_{A,k}(\omega) \tag{20b}$$

$$G_A(\omega) = \mathrm{tr}\{\bar{\Sigma}_1\tilde{Q}_{A,1}(\omega)\} + \sum_{k=1}^{n} \mathrm{tr}\{\bar{\Sigma}_v\tilde{Q}_{A,k+1}(\omega)\}, \tag{20c}$$

$$\bar{\Sigma}_1 := \left[\begin{smallmatrix} \Sigma_1 & \mathbb{E}\{\boldsymbol{x}^o_1(\boldsymbol{u}^*_F)'\} & O \\ \mathbb{E}\{\boldsymbol{u}^*_F(\boldsymbol{x}^o_1)'\} & \mathbb{E}\{\boldsymbol{u}^*_F(\boldsymbol{u}^*_F)'\} & O \\ O & O & z(\omega)z(\omega)' \end{smallmatrix}\right] \textit{ and } \bar{\Sigma}_v := \left[\begin{smallmatrix} \Sigma_v & O \\ O & O \end{smallmatrix}\right],$$

and $\{\tilde{Q}_{A,k}(\omega)\}$ *for each* $\omega \in \Omega$ *is a sequence defined through the following discrete-time Riccati equation:*

$$\tilde{Q}_{A,k+1}(\omega) = Q_{A,k}(\omega) + \bar{A}'\Big(\tilde{Q}_{A,k+1}(\omega) - \tilde{Q}_{A,k+1}(\omega)\bar{B}\Delta_{A,k}(\omega)^{-1}\bar{B}'\tilde{Q}_{A,k+1}(\omega)\Big)\bar{A}, \tag{21a}$$

$$\tilde{Q}_{A,n+1}(\omega) = Q_{A,n+1}(\omega) \text{ and } Q_{A,1}(\omega) = O. \tag{21b}$$

And corresponding to (11), the adversarial objective (19) can be written as

$$\min_{\substack{\gamma_{A,k}(\omega,\cdot)\in\Gamma_k\\ k=1,\dots,n}} \sum_{k=1}^{n} \mathbb{E}\|\,\delta\boldsymbol{u}_{\,k}^{\,o} + K_{A,k}(\omega)\bar{\boldsymbol{x}}_k^o\|^2_{\Delta_{A,k}(\omega)} + G_A(\omega) \tag{22}$$

subject to (20)–(21) and

$$\bar{\boldsymbol{x}}_{k+1}^o = \bar{A}\bar{\boldsymbol{x}}_k^o + E\boldsymbol{v}_k,\ k = 1,\dots,n, \text{ and } \bar{\boldsymbol{x}}_1^o = \bar{\boldsymbol{x}}_1, \tag{23a}$$

$$\delta\boldsymbol{u}_{\,k}^{\,o} = \delta\boldsymbol{u}_{\,k} + K_{A,k}(\omega)\bar{B}\,\delta\boldsymbol{u}_{\,k-1} + K_{A,k}(\omega)\bar{A}\bar{B}\,\delta\boldsymbol{u}_{\,k-2} + \dots + K_{A,k}(\omega)\bar{A}^{k-2}\bar{B}\,\delta\boldsymbol{u}_{\,1}. \tag{23b}$$

Note that in (22), $C_A(\omega)$ is independent from the adversary's optimization arguments even though it depends on $\boldsymbol{u}_F^*$ due to $\bar{\Sigma}_1$ in (20c). Furthermore, given the sensor outputs $\boldsymbol{s}_{[1,k]} = s_{[1,k]}$, the optimal *transformed* adversary action $\delta u_{A,k}^{o*}$ of (23b) is given by

$$\delta u_{A,k}^{o*} = -K_{A,k}(\omega)\mathbb{E}\{\bar{\boldsymbol{x}}_k^o|\boldsymbol{s}_{[1,k]} = s_{[1,k]}\},$$

which also implies

$$\delta\boldsymbol{u}_{\,k}^{\,o*} = -K_{A,k}(\omega)\mathbb{E}\{\bar{\boldsymbol{x}}_k^o|\boldsymbol{s}_{[1,k]}\} \tag{24}$$

almost everywhere on $\mathbb{R}^r$. By (23b), we have

$$\underbrace{\begin{bmatrix} \delta\boldsymbol{u}_{\,n}^{\,o} \\ \delta\boldsymbol{u}_{\,n-1}^{\,o} \\ \vdots \\ \delta\boldsymbol{u}_{\,1}^{\,o} \end{bmatrix}}_{=:\ \delta\boldsymbol{u}^{\,o}} = \underbrace{\begin{bmatrix} I\ K_{A,n}(\omega)\bar{B}_{n-1} & \cdots & K_{A,n}(\omega)\bar{A}^{n-2}\bar{B} \\ I & \cdots & K_{A,n-1}(\omega)\bar{A}^{n-3}\bar{B} \\ & \ddots & \vdots \\ & & I \end{bmatrix}}_{=:\ \Phi_A(\omega)} \underbrace{\begin{bmatrix} \delta\boldsymbol{u}_{\,n} \\ \delta\boldsymbol{u}_{\,n-1} \\ \vdots \\ \delta\boldsymbol{u}_{\,1} \end{bmatrix}}_{=:\ \delta\boldsymbol{u}},$$

which can also be written as $\delta\boldsymbol{u}^{\,o} = \Phi_A(\omega)\,\delta\boldsymbol{u}$. And (24) leads to

$$\delta\boldsymbol{u}^{o*} = -\underbrace{\begin{bmatrix} K_{A,n}(\omega) & & \\ & \ddots & \\ & & K_{A,1}(\omega) \end{bmatrix}}_{=:\ K_A(\omega)} \begin{bmatrix} \mathbb{E}\{\bar{\boldsymbol{x}}_n^o|\boldsymbol{s}_{[1,n]}\} \\ \vdots \\ \mathbb{E}\{\bar{\boldsymbol{x}}_1^o|\boldsymbol{s}_1\} \end{bmatrix}. \tag{25}$$

Next, we seek to compute $\mathbb{E}\{\bar{\boldsymbol{x}}_k^o|\boldsymbol{s}_{[1,k]}\}$ in (24). To this end, let us take a closer look at (23a):

$$\begin{bmatrix} \check{\boldsymbol{x}}_{k+1} \\ \hdashline \boldsymbol{u}_F^* \\ z(\omega) \end{bmatrix} = \left[\begin{array}{c:c} A & \cdots\ B\ \cdots \\ \hdashline O & I \end{array}\right] \begin{bmatrix} \check{\boldsymbol{x}}_{k} \\ \hdashline \boldsymbol{u}_F^* \\ z(\omega) \end{bmatrix} + \begin{bmatrix} I \\ \hdashline O \end{bmatrix} \boldsymbol{v}_k,$$

where we introduce $\check{\boldsymbol{x}}_k$, which is given by

$$\begin{aligned}
&\check{\boldsymbol{x}}_1 = \boldsymbol{x}_1 = \boldsymbol{x}_1^o\\
&\check{\boldsymbol{x}}_2 = A\check{\boldsymbol{x}}_1 + B\boldsymbol{u}_{F,1}^* + \boldsymbol{v}_1 = \boldsymbol{x}_2^o + B\boldsymbol{u}_{F,1}^*\\
&\check{\boldsymbol{x}}_3 = A\check{\boldsymbol{x}}_2 + B\boldsymbol{u}_{F,2}^* + \boldsymbol{v}_2 = A(\boldsymbol{x}_2^o + B\boldsymbol{u}_{F,1}) + B\boldsymbol{u}_{F,2}^* + \boldsymbol{v}_2 = \boldsymbol{x}_3^o + AB\boldsymbol{u}_{F,1}^* + B\boldsymbol{u}_{F,2}^*\\
&\quad\vdots\\
&\check{\boldsymbol{x}}_k = \boldsymbol{x}_k^o + B\boldsymbol{u}_{F,k-1}^* + AB\boldsymbol{u}_{F,k-2}^* + \cdots + A^{k-2}B\boldsymbol{u}_{F,1}^*.
\end{aligned} \tag{26}$$

Then, we have

$$\begin{bmatrix} \check{\boldsymbol{x}}_n \\ \check{\boldsymbol{x}}_{n-1} \\ \vdots \\ \check{\boldsymbol{x}}_1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_n^o \\ \boldsymbol{x}_{n-1}^o \\ \vdots \\ \boldsymbol{x}_1^o \end{bmatrix} + \underbrace{\begin{bmatrix} O & B & AB & \cdots & A^{n-2}B \\ O & O & B & \cdots & A^{n-3}B \\ \vdots & & & & \vdots \\ O & O & \cdots & \cdots & O \end{bmatrix}}_{=:\,D} \begin{bmatrix} \boldsymbol{u}_{F,n}^* \\ \boldsymbol{u}_{F,n-1}^* \\ \vdots \\ \boldsymbol{u}_{F,1}^* \end{bmatrix}.$$

Let D be partitioned as $D = [D_n' \cdots D_1']'$ such that

$$\check{\boldsymbol{x}}_k = \boldsymbol{x}_k^o + D_k\boldsymbol{u}_F^*. \tag{27}$$

Therefore, $\mathbb{E}\{\bar{\boldsymbol{x}}_k^o|\boldsymbol{s}_{[1,k]}\}$ can be written as

$$\mathbb{E}\{\bar{\boldsymbol{x}}_k^o|\boldsymbol{s}_{[1,k]}\} = \begin{bmatrix} \mathbb{E}\{\boldsymbol{x}_k^o|\boldsymbol{s}_{[1,k]}\} + D_k\mathbb{E}\{\boldsymbol{u}_F^*|\boldsymbol{s}_{[1,k]}\} \\ \mathbb{E}\{\boldsymbol{u}_F^*|\boldsymbol{s}_{[1,k]}\} \\ z(\omega) \end{bmatrix}. \tag{28}$$

Furthermore, (14) and (15) lead to

$$\mathbb{E}\{\boldsymbol{u}_F^*|\boldsymbol{s}_{[1,k]}\} = -\Phi^{-1}K \begin{bmatrix} \mathbb{E}\{\mathbb{E}\{\boldsymbol{x}_n^o|\boldsymbol{s}_{[1,n]}\}|\boldsymbol{s}_{[1,k]}\} \\ \vdots \\ \mathbb{E}\{\mathbb{E}\{\boldsymbol{x}_1^o|\boldsymbol{s}_1\}|\boldsymbol{s}_{[1,k]}\} \end{bmatrix}. \tag{29}$$

Note that we have

$$\mathbb{E}\{\mathbb{E}\{\boldsymbol{x}_l^o|\boldsymbol{s}_{[1,l]}\}|\boldsymbol{s}_{[1,k]}\} = \begin{cases} \mathbb{E}\{\boldsymbol{x}_l^o|\boldsymbol{s}_{[1,k]}\} \text{ if } l \geq k \\ \mathbb{E}\{\boldsymbol{x}_l^o|\boldsymbol{s}_{[1,l]}\} \text{ if } l < k \end{cases},$$

where the first case, i.e., $l \geq k$, follows due to the iterated expectations with nested conditioning sets, i.e., $\{\boldsymbol{s}_{[1,l]}\} \supseteq \{\boldsymbol{s}_{[1,k]}\}$ if $l \geq k$, and the second case, i.e., $l < k$, follows since $\mathbb{E}\{\boldsymbol{x}_l^o|\boldsymbol{s}_{[1,l]}\}$ is σ-$\boldsymbol{s}_{[1,k]}$ measurable if $l < k$. Therefore, (29) can be written as

$$\mathbb{E}\{\boldsymbol{u}_F^*|\boldsymbol{s}_{[1,k]}\} = -\Phi^{-1}K \underbrace{\left[\begin{array}{c:c:c} & A^{n-k} & \\ & \vdots & \\ O & & O \\ & A & \\ & I & \\ \hdashline O & O & I \end{array}\right]}_{=:\,L_k} \underline{\hat{\boldsymbol{x}}}^o, \tag{30}$$

where the middle block is the kth block column. Hence, we can rewrite (28) as

$$\mathbb{E}\{\bar{\boldsymbol{x}}_k^o|\boldsymbol{s}_{[1,k]}\} = \underbrace{\begin{bmatrix} E_k - D_k \varPhi^{-1} K L_k \\ -\varPhi^{-1} K L_k \\ O \end{bmatrix}}_{=:\, F_k} \hat{\underline{\boldsymbol{x}}}^o + \underbrace{\begin{bmatrix} O \\ O \\ z(\omega) \end{bmatrix}}_{=:\, \underline{z}(\omega)}, \tag{31}$$

where E_k is the indicator matrix such that $\mathbb{E}\{\boldsymbol{x}_k^o|\boldsymbol{s}_{[1,k]}\} = E_k \hat{\underline{\boldsymbol{x}}}^o$, $k = 1, \ldots, n$. Then, by (24), (25), and (31), we have

$$\delta \boldsymbol{u}^{\,o*}(\omega) = -K_A(\omega) \underbrace{\begin{bmatrix} F_n \\ \vdots \\ F_1 \end{bmatrix}}_{=:\, F} \hat{\underline{\boldsymbol{x}}}^o - K_A(\omega)\underline{z}(\omega).$$

Therefore, the actual optimal adversarial actions are given by

$$\boxed{\boldsymbol{u}_A^*(\omega) = \boldsymbol{u}_F^* - \varPhi_A(\omega)^{-1} K_A(\omega) \big[F \hat{\underline{\boldsymbol{x}}}^o + \underline{z}(\omega)\big].} \tag{32}$$

In the following theorem, we recap the results.

Theorem 1. *Given S's strategies $\boldsymbol{s}_k = \eta_k(\boldsymbol{x}_{[1,k]})$, $k = 1, \ldots, n$, C's optimal reactions $\boldsymbol{u}_{F,k}$ and $\boldsymbol{u}_{A,k}(\omega)$ are given by (15) or (32) depending on whether C is a friend or an adversary, respectively.*

In the following section, we formulate S's optimal strategies.

5 Optimal Leader (Sensor) Actions

By Theorem 1, S's objective can be written as

$$\min_{\substack{\eta_k \in \varUpsilon_k,\\ k=1,\ldots,n}} (1-p) \sum_{k=1}^{n} \mathbb{E}\left\{\|\boldsymbol{x}_{k+1}\|^2_{Q_{k+1}} + \|\boldsymbol{u}^*_{F,k}\|^2_{R_k}\right\}$$
$$+ p \int_{\varOmega} \sum_{k=1}^{n} \mathbb{E}\left\{\|\boldsymbol{x}_{k+1}\|^2_{Q_{k+1}} + \|\boldsymbol{u}^*_{A,k}(\omega)\|^2_{R_k}\right\} \mathbf{P}(d\omega).$$

However, we should also take into account that $\boldsymbol{x}_k$ evolves according to (1), which implies that the state $\boldsymbol{x}_k$ depends on the control input, and therefore C's type. In order to show this explicit dependence, henceforth, we will denote the state by $\boldsymbol{x}_{F,k}$ when C is a friend or by $\boldsymbol{x}_{A,k}$ when C is an adversary. Correspondingly, the sensor outputs are denoted by $\boldsymbol{s}_{F,k}$ and $\boldsymbol{s}_{A,k}$, respectively. Therefore, an explicit representation for S's objective is given by

$$\min_{\substack{\eta_k \in \varUpsilon_k,\\ k=1,\ldots,n}} (1-p) \sum_{k=1}^{n} \mathbb{E}\left\{\|\boldsymbol{x}_{F,k+1}\|^2_{Q_{k+1}} + \|\boldsymbol{u}^*_{F,k}\|^2_{R_k}\right\}$$
$$+ p \int_{\varOmega} \sum_{k=1}^{n} \mathbb{E}\left\{\|\boldsymbol{x}_{A,k+1}(\omega)\|^2_{Q_{k+1}} + \|\boldsymbol{u}^*_{A,k}(\omega)\|^2_{R_k}\right\} \mathbf{P}(d\omega). \tag{33}$$

Even though S constructs a single set of strategies $\{\eta_k \in \Upsilon_k\}$ without knowing C's type, the resulting sensor outputs $\{\boldsymbol{s}_k = \eta_k(\boldsymbol{x}_{[1,k]})\}$ depend on the states, $\boldsymbol{x}_{[1,k]}$'s, hence C's type, i.e., $\boldsymbol{x}_k = \boldsymbol{x}_{F,k}$ if C is a friend or $\boldsymbol{x}_k = \boldsymbol{x}_{A,k}$ if C is an adversary.

Let $T := \Phi^{-1}K$,

$$\begin{aligned} T_A(\omega) &:= \Phi^{-1}K + \Phi_A(\omega)^{-1}K_A(\omega)F \\ \xi(\omega) &:= \Phi_A(\omega)^{-1}K_A(\omega)\underline{z}(\omega) \end{aligned}$$

such that $\boldsymbol{u}_F^* = -T\,\underline{\hat{\boldsymbol{x}}}_F^o$ and $\boldsymbol{u}_A^*(\omega) = -T_A(\omega)\,\underline{\hat{\boldsymbol{x}}}_A - \xi(\omega)$, where $\underline{\hat{\boldsymbol{x}}}_\iota^o := \left[(\hat{\boldsymbol{x}}_{\iota,n}^o)' \cdots (\hat{\boldsymbol{x}}_{\iota,1}^o)'\right]'$ and $\hat{\boldsymbol{x}}_{\iota,k}^o := \mathbb{E}\{\boldsymbol{x}_k^o|\boldsymbol{s}_{\iota,[1,k]}\}$, for $\iota = \{F, A\}$. Note that the matrices T and $T_A(\omega)$, for each $\omega \in \Omega$, are block upper triangular. Furthermore, let $\underline{\hat{\boldsymbol{x}}}_{\iota,k}^o := \left[(\hat{\boldsymbol{x}}_{\iota,k}^o)' \cdots (\hat{\boldsymbol{x}}_{\iota,1}^o)'\right]'$, $\xi(\omega)$ be partitioned into $\xi(\omega) = [\xi_n(\omega)' \cdots \xi_1(\omega)']'$, and the block upper triangular matrices T and $T_A(\omega)$ be partitioned into the block matrices as

$$T = \begin{bmatrix} T_{n,n} & T_{n,n-1} & \cdots & T_{n,1} \\ & T_{n-1,n-1} & \cdots & T_{n-1,1} \\ & & \ddots & \\ & & & T_{1,1} \end{bmatrix}, T_A = \begin{bmatrix} T_{A,n,n} & T_{A,n,n-1} & \cdots & T_{A,n,1} \\ & T_{A,n-1,n-1} & \cdots & T_{A,n-1,1} \\ & & \ddots & \\ & & & T_{A,1,1} \end{bmatrix},$$

where we have dropped the argument ω for notational simplicity, and $\bar{T}_k := [T_{k,k} \ \cdots \ T_{k,1}]$, $\bar{T}_{A,k}(\omega) := [T_{A,k,k}(\omega) \ \cdots \ T_{A,k,1}(\omega)]$. Then, by Lemma 1 and (11), (33) is equivalent to

$$\begin{aligned} \min_{\substack{\eta_k \in \Upsilon_k, \\ k=1,\dots,n}} \ (1-p) \ &\sum_{k=1}^n \mathbb{E}\|K_k\boldsymbol{x}_k^o - \bar{T}_k\,\underline{\hat{\boldsymbol{x}}}_{F,k}^o\|_{\Delta_k}^2 \\ &+ p\int_\Omega \sum_{k=1}^n \mathbb{E}\|K_k\boldsymbol{x}_k^o - \bar{T}_{A,k}(\omega)\,\underline{\hat{\boldsymbol{x}}}_{A,k}^o(\omega) - \xi_k(\omega)\|_{\Delta_k}^2 \mathbf{P}(d\omega) + G. \end{aligned} \tag{34}$$

The first summation in (34) can be written as

$$\begin{aligned} \sum_{k=1}^n \operatorname{tr}\big\{\mathbb{E}\{\boldsymbol{x}_k^o(\boldsymbol{x}_k^o)'\}K_k'\Delta_k K_k\big\} - 2\operatorname{tr}\big\{\mathbb{E}\{\,\underline{\hat{\boldsymbol{x}}}_{F,k}^o(\boldsymbol{x}_k^o)'\}K_k'\Delta_k\bar{T}_k\big\} \\ + \operatorname{tr}\big\{\mathbb{E}\{\,\underline{\hat{\boldsymbol{x}}}_{F,k}^o(\,\underline{\hat{\boldsymbol{x}}}_{F,k}^o)'\}\bar{T}_k'\Delta_k\bar{T}_k\big\} \end{aligned} \tag{35}$$

while the second summation can be written as

$$\begin{aligned}
&\sum_{k=1}^{n} \mathrm{tr}\{\mathbb{E}\{\boldsymbol{x}_k^o(\boldsymbol{x}_k^o)'\}K_k'\Delta_k K_k\} + \int_{\Omega} \xi_k(\omega)'\Delta_k \xi_k(\omega)\mathbf{P}(d\omega)\\
&+ \int_{\Omega} \mathrm{tr}\big\{\mathbb{E}\{\,\underline{\hat{\boldsymbol{x}}}_{A,k}^o(\omega)\,\underline{\hat{\boldsymbol{x}}}_{A,k}^o(\omega)'\}\bar{T}_{A,k}(\omega)'\Delta_k \bar{T}_{A,k}(\omega)\big\}\mathbf{P}(d\omega)\\
&- 2\int_{\Omega} \mathrm{tr}\big\{\mathbb{E}\{\,\underline{\hat{\boldsymbol{x}}}_{A,k}^o(\omega)(\boldsymbol{x}_k^o)'\}K_k'\Delta_k \bar{T}_{A,k}(\omega)\big\}\mathbf{P}(d\omega)\\
&+ 2\int_{\Omega} \mathrm{tr}\big\{\mathbb{E}\{\,\underline{\hat{\boldsymbol{x}}}_{A,k}^o(\omega)\}\xi_k(\omega)'\Delta_k \bar{T}_{A,k}(\omega)\big\}\mathbf{P}(d\omega)\\
&- 2\int_{\Omega} \mathrm{tr}\big\{\mathbb{E}\{\boldsymbol{x}_k^o\}\xi_k(\omega)'\Delta_k K_k\big\}\mathbf{P}(d\omega),
\end{aligned} \tag{36}$$

where the last term is zero since $\boldsymbol{x}_k^o$ is zero-mean. The following lemma says that the posterior covariances do not depend on ω.

Lemma 3. *The posterior $\hat{\boldsymbol{x}}_{A,k}^o(\omega)$ is independent of ω. Further, both posteriors $\hat{\boldsymbol{x}}_{F,k}^o$ and $\hat{\boldsymbol{x}}_{A,k}^o$ are equivalent and given by*

$$\boxed{\hat{\boldsymbol{x}}_k^o := \hat{\boldsymbol{x}}_{F,k}^o = \hat{\boldsymbol{x}}_{A,k}^o(\omega) = \mathbb{E}\left\{\boldsymbol{x}_k^o \,|\, \eta_1(\boldsymbol{x}_1^o), \dots, \eta_k(\boldsymbol{x}_{[1,k]}^o)\right\}.} \tag{37}$$

Proof. Consider the state recursion when C is a friend:

$$\boldsymbol{x}_{F,k+1} = A\boldsymbol{x}_{F,k} + B\boldsymbol{u}_{F,k}^* + \boldsymbol{v}_k,$$

which can also be written as[4]

$$\begin{aligned}
\boldsymbol{x}_{F,1} &= \boldsymbol{x}_1^o\\
\boldsymbol{x}_{F,2} &= A\boldsymbol{x}_{F,1} + B\boldsymbol{u}_{F,1}^* + \boldsymbol{v}_1 = \boldsymbol{x}_2^o + B\boldsymbol{u}_{F,1}^*\\
\boldsymbol{x}_{F,3} &= A\boldsymbol{x}_{F,2} + B\boldsymbol{u}_{F,2}^* + \boldsymbol{v}_2 = A(\boldsymbol{x}_2^o + B\boldsymbol{u}_{F,1}^*) + B\boldsymbol{u}_{F,2}^* + \boldsymbol{v}_2\\
&= \boldsymbol{x}_3^o + AB\boldsymbol{u}_{F,1}^* + B\boldsymbol{u}_{F,2}^*\\
&\;\;\vdots\\
\boldsymbol{x}_{F,k} &= \boldsymbol{x}_k^o + B\boldsymbol{u}_{F,k-1}^* + AB\boldsymbol{u}_{F,k-2}^* + \dots + A^{k-2}B\boldsymbol{u}_{F,1}^*.
\end{aligned}$$

Let $M_k := [B \; AB \; \cdots \; A^{k-2}B]$ and $\underline{\boldsymbol{u}}_{F,k} := [\boldsymbol{u}_{F,k}' \; \cdots \; \boldsymbol{u}_{F,1}']'$. Then, for $k > 1$, we have

$$\boldsymbol{x}_{F,k} := \boldsymbol{x}_k^o + M_{k-1}\underline{\boldsymbol{u}}_{F,k-1}. \tag{38}$$

Furthermore, let

$$T_k := \begin{bmatrix} T_{k,k} & \cdots & T_{k,1}\\ & \ddots & \vdots\\ & & T_{1,1} \end{bmatrix}$$

[4] Note the resemblance to (26).

such that $\underline{\boldsymbol{u}}_{F,k} = -T_k\,\underline{\hat{\boldsymbol{x}}}^o_{F,k}$ and (38) can be written as

$$\boldsymbol{x}_{F,k} = \boldsymbol{x}^o_k - M_{k-1}T_{k-1}\,\underline{\hat{\boldsymbol{x}}}^o_{F,k-1}. \tag{39}$$

Therefore, we have $\hat{\boldsymbol{x}}^o_{F,k} = \mathbb{E}\{\boldsymbol{x}^o_k|\eta_1(\boldsymbol{x}^o_1),\ldots,\eta_k(\boldsymbol{x}^o_1,\ldots,\boldsymbol{x}^o_k - c_{k,k})\}$, for certain deterministic $c_{i,j} \in \mathbb{R}^m$, $i,j = 1,\ldots,k$, since $\hat{\boldsymbol{x}}^o_{F,j}$ is $\sigma - \boldsymbol{x}^o_{[1,j]}$ measurable. Correspondingly, we have

$$\boldsymbol{x}_{A,k}(\omega) = \boldsymbol{x}^o_k - M_{k-1}T_{A,k-1}(\omega)\,\underline{\hat{\boldsymbol{x}}}^o_{A,k-1} - M_{k-1}\underline{\xi}_{k-1}(\omega), \tag{40}$$

where

$$T_{A,k}(\omega) := \begin{bmatrix} T_{A,k,k}(\omega) & \cdots & T_{A,k,1}(\omega) \\ & \ddots & \vdots \\ & & T_{A,1,1}(\omega) \end{bmatrix} \text{ and } \underline{\xi}_k(\omega) := \begin{bmatrix} \xi_k(\omega) \\ \vdots \\ \xi_1(\omega) \end{bmatrix},$$

which leads to $\hat{\boldsymbol{x}}^o_{A,k}(\omega) = \mathbb{E}\{\boldsymbol{x}^o_k|\eta_1(\boldsymbol{x}^o_1),\ldots,\eta_k(\boldsymbol{x}^o_1,\ldots,\boldsymbol{x}^o_k - d_{k,k}(\omega))\}$, for certain other deterministic $d_{i,j}(\omega) \in \mathbb{R}^m$, $i,j = 1,\ldots,k$, since $\hat{\boldsymbol{x}}^o_{A,j}(\omega)$ is $\sigma - \boldsymbol{x}^o_{[1,j]}$ measurable.

Next, we employ the following lemma about shifting of random variables in order to compute $\hat{\boldsymbol{x}}^o_{F,k}$'s and $\hat{\boldsymbol{x}}^o_{A,k}(\omega)$'s.

Lemma 4. *Let* $(\Omega, \mathsf{F}, \mathbf{P})$ *be a probability space, where* Ω *is the outcome space with an appropriate* σ*-algebra* F*, and* $\mathbf{P}$ *is a distribution over* Ω*. Let also* $\boldsymbol{x} : (\Omega, \mathsf{F}) \rightarrow (\mathbb{R}^m, \mathsf{B}^m)$ *be a random variable,* $h : (\mathbb{R}^m, \mathsf{B}^m) \rightarrow (\mathbb{R}^m, \mathsf{B}^m)$ *be a Borel measurable function, and* $c \in \mathbb{R}^m$ *be a deterministic vector. Then, we have*

$$\mathbb{E}\{\boldsymbol{x}|h(\boldsymbol{x})\} = \mathbb{E}\{\boldsymbol{x}|h(\boldsymbol{x}+c)\}. \tag{41}$$

Proof. The proof is provided in the Appendix A. ■

Therefore, Lemma 4 and (51) imply (37) and the proof is concluded. □

Next, by (35), (36), and Lemma 3, (34) can be written as

$$\begin{aligned}\min_{\substack{\eta_k \in \Upsilon_k,\\ k=1,\ldots,n}} \sum_{k=1}^{n} & \operatorname{tr}\{\Sigma_k K_k' \Delta_k K_k\} + p\,\mathbb{E}_\Omega\{\xi_k(\omega)'\Delta_k \xi_k(\omega)\} \\ & - 2\operatorname{tr}\Big\{\mathbb{E}\{\,\underline{\hat{\boldsymbol{x}}}^o_k(\boldsymbol{x}^o_k)'\}\,K_k'\Delta_k\big((1-p)\bar{T}_k + p\,\mathbb{E}_\Omega\{\bar{T}_{A,k}(\omega)\}\big)\Big\} \\ & + p\operatorname{tr}\Big\{\mathbb{E}\{\,\underline{\hat{\boldsymbol{x}}}^o_k(\,\underline{\hat{\boldsymbol{x}}}^o_k)'\}\,\mathbb{E}_\Omega\{\bar{T}_{A,k}(\omega)'\Delta_k\bar{T}_{A,k}(\omega)\}\Big\} \\ & + (1-p)\operatorname{tr}\Big\{\mathbb{E}\{\,\underline{\hat{\boldsymbol{x}}}^o_k(\,\underline{\hat{\boldsymbol{x}}}^o_k)'\}\,\bar{T}_k'\Delta_k\bar{T}_k\Big\} + G,\end{aligned} \tag{42}$$

where $\mathbb{E}_\Omega$ denotes the expectation taken over Ω with respect to the distribution $\mathbf{P}$ and $\Sigma_k := \mathbb{E}\{\boldsymbol{x}^o_k(\boldsymbol{x}^o_k)'\}$.

We note that for $l \leq k$, $\mathbb{E}\{\,\hat{\boldsymbol{x}}^o_l(\boldsymbol{x}^o_k)'\} = \mathbb{E}\{\,\hat{\boldsymbol{x}}^o_l(\boldsymbol{x}^o_l)'\}(A')^{k-l}$ since $\boldsymbol{v}_j$, $j > l$, and $\hat{\boldsymbol{x}}^o_l$, which is σ-$\boldsymbol{s}_{[1,l]}$ measurable, are independent of each other and $\{\boldsymbol{v}_k\}$ is a zero-mean white noise process. Furthermore, we have

$$\begin{aligned}\mathbb{E}\{\,\hat{\boldsymbol{x}}^o_l(\boldsymbol{x}^o_l)'\} &= \mathbb{E}\{\mathbb{E}\{\,\hat{\boldsymbol{x}}^o_l(\boldsymbol{x}^o_l)'|\boldsymbol{s}_{[1,l]}\}\} \\ &= \mathbb{E}\{\,\hat{\boldsymbol{x}}^o_l(\,\hat{\boldsymbol{x}}^o_l)'\}\end{aligned} \tag{43}$$

due to the law of iterated expectations. Let $H_k := \mathbb{E}\{\hat{\boldsymbol{x}}_k^o(\hat{\boldsymbol{x}}_k^o)'\}$. Then, we have

$$\mathbb{E}\{\underline{\hat{\boldsymbol{x}}}_k^o(\boldsymbol{x}_k^o)'\} = \begin{bmatrix} H_k \\ H_{k-1}A' \\ \vdots \\ H_1(A')^{k-1} \end{bmatrix}, \mathbb{E}\{\underline{\hat{\boldsymbol{x}}}_{k-1}^o(\boldsymbol{x}_k^o)'\} = \begin{bmatrix} H_{k-1}A' \\ \vdots \\ H_1(A')^{k-1} \end{bmatrix}$$

and

$$\begin{aligned}\mathbb{E}\{\underline{\hat{\boldsymbol{x}}}_k^o(\underline{\hat{\boldsymbol{x}}}_k^o)'\} &= \begin{bmatrix} \mathbb{E}\{\hat{\boldsymbol{x}}_k^o(\hat{\boldsymbol{x}}_k^o)'\} & \cdots & \mathbb{E}\{\hat{\boldsymbol{x}}_k^o(\hat{\boldsymbol{x}}_1^o)'\} \\ \vdots & & \vdots \\ \mathbb{E}\{\hat{\boldsymbol{x}}_1^o(\hat{\boldsymbol{x}}_k^o)'\} & \cdots & \mathbb{E}\{\hat{\boldsymbol{x}}_1^o(\hat{\boldsymbol{x}}_1^o)'\} \end{bmatrix} \\ &= \begin{bmatrix} H_k & AH_{k-1} & \cdots & A^{k-1}H_1 \\ H_{k-1}A' & H_{k-1} & \cdots & A^{k-2}H_1 \\ \vdots & \vdots & \ddots & \vdots \\ H_1(A')^{k-1} & H_1(A')^{k-2} & \cdots & H_1 \end{bmatrix} \end{aligned} \tag{44}$$

since for $l < k$, we have

$$\begin{aligned}\mathbb{E}\{\hat{\boldsymbol{x}}_l^o(\hat{\boldsymbol{x}}_k^o)'\} &= \mathbb{E}\{\mathbb{E}\{\hat{\boldsymbol{x}}_l^o(\hat{\boldsymbol{x}}_k^o)'|\boldsymbol{s}_{[1,l]}\}\} \\ &\overset{(a)}{=} \mathbb{E}\{\hat{\boldsymbol{x}}_l^o\mathbb{E}\{\hat{\boldsymbol{x}}_k^o|\boldsymbol{s}_{[1,l]}\}'\} \\ &\overset{(b)}{=} \mathbb{E}\{\hat{\boldsymbol{x}}_l^o(\hat{\boldsymbol{x}}_l^o)'\}(A')^{k-l},\end{aligned}$$

where (a) holds since $\hat{\boldsymbol{x}}_l^o$ is σ-$\boldsymbol{s}_{[1,l]}$ measurable, and (b) follows due to the iterated expectations with nested conditioning sets, i.e., $\{\boldsymbol{s}_{[1,l]}\} \subseteq \{\boldsymbol{s}_{[1,k]}\}$.

Next, we can rewrite (42) as

$$\min_{\substack{\eta_k\in\Upsilon_k,\\ k=1,\ldots,n}} \sum_{k=1}^n \Xi_k^o + \operatorname{tr}\left\{\begin{bmatrix} H_k \\ H_{k-1}A' \\ \vdots \\ H_1(A')^{k-1} \end{bmatrix} \Xi_k\right\} + \operatorname{tr}\left\{\begin{bmatrix} H_k & AH_{k-1} & \cdots & A^{k-1}H_1 \\ H_{k-1}A' & H_{k-1} & \cdots & A^{k-2}H_1 \\ \vdots & \vdots & \ddots & \vdots \\ H_1(A')^{k-1} & H_1(A')^{k-2} & \cdots & H_1 \end{bmatrix} \bar{\Xi}_k\right\}, \tag{45}$$

where

$$\begin{aligned}\Xi_k^o &:= \operatorname{tr}\{\Sigma_k K_k' \Delta_k K_k\} + p\,\mathbb{E}_\Omega\{\xi_k(\omega)'\Delta_k\xi_k(\omega)\} + \frac{1}{n}G \\ \Xi_k &:= -2K_k'\Delta_k\big((1-p)\bar{T}_k + p\,\mathbb{E}_\Omega\{\bar{T}_{A,k}(\omega)\}\big) \\ \bar{\Xi}_k &:= p\mathbb{E}_\Omega\{\bar{T}_{A,k}(\omega)'\Delta_k\bar{T}_{A,k}(\omega)\} + (1-p)\bar{T}_k'\Delta_k\bar{T}_k,\end{aligned}$$

which are independent of the optimization arguments. Hence, the optimization problem (42) faced by S can be written as an affine function of H_k's as follows:

$$\min_{\substack{\eta_k\in\Upsilon_k,\\ k=1,\ldots,n}} \sum_{k=1}^n \operatorname{tr}\{V_k H_k\} + \Xi^o, \tag{46}$$

for certain symmetric deterministic matrices $V_k \in \mathbb{R}^{m\times m}$, $k = 1,\ldots,n$, where $\Xi^o := \sum_{k=1}^n \Xi_k^o$. Note that as a sensor designer, we seek to solve this infinite-dimensional optimization problem (46) within the general class of strategies.

To this end, we employ the approach in [19], which considers a finite-dimensional optimization problem that bounds the original infinite dimensional one from below, and then, compute strategies for the original problem, which optimizes the lower bound. Based on this, the following theorem characterizes equilibrium achieving strategies of both agents S and C.

Theorem 2. *The multi-stage static Bayesian Stackelberg equilibrium between S and C, i.e., (6), can be attained through linear strategies, i.e., the secure sensor outputs $s_{[1,n]}$ are linear in the state $x_{[1,n]}$ and the corresponding, friendly or adversarial, control inputs, $u_{F,[1,n]}$ or $u_{A,[1,n]}$, are linear in the sensor outputs $s_{[1,n]}$.*

Proof. Based on Lemma 1 in [19], by characterizing necessary conditions on H_k's, we have

$$\begin{aligned} &\min_{\substack{S_k\in\mathbb{S}^m,\\ k=1,\ldots,n}} \sum_{k=1}^n \mathrm{tr}\{V_k S_k\} \;\leq \min_{\substack{\eta_k\in\Upsilon_k,\\ k=1,\ldots,n}} \sum_{k=1}^n \mathrm{tr}\{V_k H_k\},\\ &\text{s.t.}\quad \Sigma_j \succeq S_j \succeq A S_{j-1} A' \;\forall j \end{aligned} \tag{47}$$

where $\Sigma_j := \mathbb{E}\{\boldsymbol{x}_j^o(\boldsymbol{x}_j^o)'\}$ and $\mathbb{S}^m$ denotes the set of $m\times m$ symmetric matrices. Note that the left hand side of (47) is a finite-dimensional optimization, indeed an SDP, problem. By invoking Theorem 3 in [19], we can characterize the solutions of this SDP problem, $S_1^*,\ldots,S_n^*$, as

$$S_k^* = A S_{k-1}^* A' + (\Sigma_k - A S_{k-1}^* A')^{1/2} P_k (\Sigma_k - A S_{k-1} A')^{1/2}, \tag{48}$$

for $k=1,\ldots,n$, where $S_0^* = O$ and P_k's are certain symmetric idempotent matrices. Note that by solving the SDP problem numerically, we can compute the corresponding P_k's.

Next, say that S employs memoryless linear policies $\boldsymbol{s}_k = \eta_k(\boldsymbol{x}_{F,k}) = C_k'\boldsymbol{x}_{F,k}$ if C is friendly or $\boldsymbol{s}_k = \eta_k(\boldsymbol{x}_{A,k}(\omega)) = C_k'\boldsymbol{x}_{A,k}(\omega)$. Then, by Lemma 3, we have

$$\hat{\boldsymbol{x}}_k^o = \mathbb{E}\{\boldsymbol{x}_k^o | C_1'\boldsymbol{x}_1^o,\ldots,C_k'\boldsymbol{x}_k^o\}.$$

which can also be written as

$$\hat{\boldsymbol{x}}_k^o = A\hat{\boldsymbol{x}}_{k-1}^o + (\Sigma_k - AH_{k-1}A')C_k(C_k'(\Sigma_k - AH_{k-1}A')C_k)^+ C_k'(\boldsymbol{x}_k^o - A\hat{\boldsymbol{x}}_{k-1}^o),$$

for $k=1,\ldots,n$, $\hat{\boldsymbol{x}}_{-1}^o := 0$ and $H_0 := O$. Therefore, $H_k = \mathbb{E}\{\hat{\boldsymbol{x}}_k^o(\hat{\boldsymbol{x}}_k^o)'\}$ is given by

$$H_k = AH_{k-1}A' + (\Sigma_k - AH_{k-1}A')C_k(C_k'(\Sigma_k - AH_{k-1}A')C_k)^+ C_k'(\Sigma_k - AH_{k-1}A'). \tag{49}$$

We emphasize the resemblance between (48) and (49). In particular, if we set $\bar{C}_k := (\Sigma_k - AH_{k-1}A')^{1/2}C_k$, $k=1,\ldots,n$, (49) yields

$$H_k = AH_{k-1}A' + (\Sigma_k - AH_{k-1}A')^{1/2}\bar{C}_k(\bar{C}_k'\bar{C}_k)^+\bar{C}_k'(\Sigma_k - AH_{k-1}A')^{1/2},$$

where $\bar{C}_k(\bar{C}_k'\bar{C}_k)^+\bar{C}_k'$ is also a symmetric idempotent matrix just like P_k in (48).

Therefore, given P_k's, let $P_k = U_k \Lambda_k U_k'$ be the eigen decomposition and set $\bar{C}_k = U_k \Lambda_k$, i.e., set

$$C_k = (\Sigma_k - A S_{k-1}^* A')^{-1/2} U_k \Lambda_k. \tag{50}$$

Then, we obtain $H_k = S_k^*$, which implies that S's optimal strategies are memoryless and linear in the underlying state. Correspondingly, the optimal control inputs for both friendly and adversarial C are linear in the sensor outputs by (15) or (32). □

In Table 1, we provide a numerical algorithm to design secure sensors in advance.

Table 1. Computation of equilibrium achieving sender policies.

```
Algorithm: Secure Sensor Design

SDP Problem:
    Compute $V_k$, for $k = 1, \ldots, n$, by (7)-(45).
    Solve the SDP problem on the left hand side of (47) through a numerical toolbox
        and obtain the solutions $S_k^*$, for $k = 1, \ldots, n$.
    Set $S_0^* = O$.
Equilibrium achieving sensor strategies:
    Compute the corresponding idempotent matrices $P_k, \forall k$, by using $S_k^*$, $\forall k$, and (48).
    Compute the eigen decompositions: $P_k = U_k \Lambda_k U_k'$.
    Compute $C_k$, $\forall k$, by using $S_{k-1}^*, U_k, \Lambda_k$, and (50).
```

6 Conclusion

In this paper, we have proposed and addressed secure sensor design problem for cyber-physical systems with linear quadratic Gaussian dynamics against the advanced persistent threats with control objectives. By designing sensor outputs cautiously in advance, we have sought to minimize the damage that can be caused by undetected target-specific threats. However, this is not an active defense strategy against a detected threat. Therefore, such a defense mechanism should also consider the maintenance of the ordinary operations of the system. To this end, we have modeled the problem formally in a game-theoretical setting. We have determined the optimal control inputs for both friendly and adversarial objectives. Then, we have characterized the secure sensor strategies, showing that the strategies that are memoryless and linear in the underlying state lead to the equilibrium. Finally, we have provided an algorithm to compute these strategies numerically.

Some future directions of research on this topic include secure sensor design when the sensor has access to the state only partially, e.g., noisy observation, or when the attackers infiltrate into the controller within the horizon. Note also that we have only considered the secure sensor design within optimal control framework. Formulations for, e.g., robust control or feedback stability of the systems, can also be interesting future research directions.

A Appendix: Proof of Lemma 4

Let $\boldsymbol{y}_1 = h(\boldsymbol{x})$ and $\boldsymbol{y}_2 = h(\boldsymbol{x} + c)$ be random variables, where c is a deterministic shift vector of the same dimension as $\boldsymbol{x}$. Then, for any $B \in \mathsf{B}^p$, we have $\boldsymbol{y}_1^{-1}(B) = \{\omega \in \Omega : \boldsymbol{y}_1(\omega) \in B\} = \{\omega \in \Omega : h(\boldsymbol{x})(\omega) \in B\} = \{\omega \in \Omega : \boldsymbol{x}(\omega) \in h^{-1}(B)\}$. Correspondingly, we also have $\boldsymbol{y}_2^{-1}(B) = \{\omega \in \Omega : \boldsymbol{y}_2(\omega) \in B\} = \{\omega \in \Omega : h(\boldsymbol{x} + c)(\omega) \in B\} = \{\omega \in \Omega : \boldsymbol{x}(\omega) \in h^{-1}(B) - c\}$. Note that the σ-algebras generated by the random variables $\boldsymbol{y}_1$ and $\boldsymbol{y}_2$ are given by $\sigma(\boldsymbol{y}_i) = \{\boldsymbol{y}_i^{-1}(B) : B \in \mathsf{B}^p\}$, for $i = 1, 2$ [3]. This implies that $\sigma(\boldsymbol{y}_1) = \{\{\omega \in \Omega : \boldsymbol{x}(\omega) \in h^{-1}(B)\} : B \in \mathsf{B}^p\}$ and $\sigma(\boldsymbol{y}_2) = \{\{\omega \in \Omega : \boldsymbol{x}(\omega) \in h^{-1}(B) - c\} : B \in \mathsf{B}^p\}$. Furthermore, for each $B \in \mathsf{B}^p$, there exists $B_2 \in \mathsf{B}^p$ such that

$$h^{-1}(B) = h^{-1}(B_2) - c \in \mathsf{B}^p$$

since Borel sets are shift invariant [3]. Therefore, we have

$$\sigma(\boldsymbol{y}_1) = \sigma(\boldsymbol{y}_2) \tag{51}$$

and correspondingly, we obtain (41).

References

1. Başar, T., Olsder, G.: Dynamic Noncoopertative Game Theory. Society for Industrial Mathematics (SIAM) Series in Classics in Applied Mathematics. SIAM, Philadelphia (1999)
2. Bansal, R., Başar, T.: Simultaneous design of measurement and control strategies for stochastic systems with feedback. Automatica **25**(5), 679–694 (1989)
3. Billingsley, P.: Probability and Measure. Wiley, New Jersey (2012)
4. Brangetto, P., Aubyn, M.K.-S.: Economic aspects of national cyber security strategies. Technical report, NATO Cooperative Cyber Defense Centre of Excellence Tallinn, Estonia (2015)
5. Chen, Y., Kar, S., Moura, J.M.F.: Cyber physical attacks constrained by control objectives. In: Proceedings of American Control Conference (ACC), pp. 1185–1190 (2016)
6. Chen, Y., Kar, S., Moura, J.M.F.: Cyber physical attacks with control objectives and detection constraints. In: Proceedings of the 55th IEEE Conference on Decision and Control (CDC), pp. 1125–1130 (2016)

7. Fawzi, H., Tauada, P., Diggavi, S.: Secure estimation and control for cyber physical systems under adversarial attacks. IEEE Trans. Autom. Control **59**(6), 1454–1467 (2014)
8. Karnouskos, S.: Stuxnet worm impact on industrial cyber-physical system security. In: Proceedings of IEEE Industrial Electronics Society (IECON) (2011)
9. Khaitan, S.K., McCalley, J.D.: Design techniques and applications of cyberphysical systems: a survey. IEEE Syst. J. **9**(2), 350–365 (2014)
10. Koscher, K., Czeskis, A., Roesner, F., Patel, S., Kohno, T., Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., Savage, S.: Experimental security analysis of a modern automobile. In: Proceedings of IEEE Symposium on Security and Privacy, pp. 447–462, 2010
11. Kumar, P.R., Varaiya, P.: Stochastic Systems: Estimation, Identification and Adaptive Control. Prentice Hall, Englewood Cliffs (1986)
12. Liberzon, D.: Calculus of Variations and Optimal Control Theory: A Concise Introduction. Princeton University Press, Princeton (2011)
13. Miao, F., Zhu, Q., Pajic, M., Pappas, G.J.: Coding schemes for securing cyber-physical systems against stealthy data injection attacks. IEEE Trans. Autom. Control **4**, 106–117 (2017)
14. Mo, Y., Sinopoli, B.: Integrity attacks on cyber-physical systems. In: Proceedings of the 1st ACM International Conference on High Confidence Networked Systems, pp. 47–54, 2012
15. Myerson, R.B.: Game Theory: Analysis of Conflict. Harvard University Press, Cambridge (1997)
16. Nelson, N.: The impact of Dragonfly malware on industrial control systems. The SANS Institute (2016)
17. Paruchuri, P., Pearce, J.P., Marecki, J., Tambe, M., Ordonez, F., Karus, S.: Playing games for security: An efficient exact algorithm for solving Bayesian Stackelberg games. In: Proceedings of Autonomous Agents and Multiagent Systems (AAMAS) (2008)
18. Pasqualetti, F., Dorfler, F., Bullo, F.: Attack detection and identification in cyber-physical systems. IEEE Trans. Autom. Control **58**(11), 2715–2729 (2013)
19. Sayin, M.O., Akyol, E., Başar, T.: Hierarchical multi-stage Gaussian signaling games: strategic communication and control. Automatica, arXiv:1609.09448 (2017, submitted)
20. Zhang, R., Venkitasubramaniam, P.: Stealthy control signal attacks in linear quadratic Gaussian control systems: detectability reward tradeoff. IEEE Trans. Inf. Forensics Secur. **12**(7), 1555–1570 (2017)