A Jointly Optimal Design of Control and Scheduling in Networked Systems under Denial-of-Service Attacks *

Jingyi Lu a, Daniel E. Quevedo b

a Department of Mechanical Engineering, University of Paderborn, Germany.
b School of Electrical Engineering & Robotics, Queensland University of Technology, Brisbane, Australia.

Abstract

We consider the joint design of control and scheduling under stochastic Denial-of-Service (DoS) attacks in the context of networked control systems. A sensor takes measurements of the system output and forwards its dynamic state estimates to a remote controller over a packet-dropping link. The controller determines the optimal control law for the process using the estimates it receives. An attacker aims at degrading the control performance by increasing the packet-dropout rate with a DoS attack towards the sensor-controller channel. We assume both the controller and the attacker are rational in a game-theoretic sense and establish a partially observable stochastic game to derive the optimal joint design of scheduling and control. Using dynamic programming we prove that the control and scheduling policies can be designed separately without sacrificing optimality, making the problem equivalent to a complete information game. We employ Nash Q-learning to solve the problem and prove that the solution is guaranteed to constitute an $\varepsilon$-Nash equilibrium. Numerical examples are provided to illustrate the tradeoffs between control performance and communication cost.

Key words: networked control, cyber attacks, Kalman filter, dynamic programming, stochastic game

1 Introduction

Cyber-physical systems (CPSs) have successfully integrated computing devices and networking infrastructure to remotely sense and control the physical world. This enables the development of exciting applications in manufacturing, transportation, and smart grid [13]. However, the wireless communication among components of CPSs introduces vulnerabilities against malicious adversaries [5]. For example, data integrity and availability can be easily deteriorated by replay attacks [23] and denial-of-service (DoS) attacks [29]. This weakness brings a great threat to the security and safety of CPSs, especially when operating in closed-loop. This raises the issue of how to design systematic prevention mechanisms to enhance system security.

Game theory is often employed as a tool for resilient design and analysis in the presence of adversaries [21]. Assuming the rationality of the adversary, game-theoretic frameworks can model the interactive decision-making process and yield non-conservative strategies. They have been widely applied to the secure design of control and estimation. For example, Zhu and Başar [31] formulated an infinite time horizon zero-sum game to derive a secure control policy that is resilient against various types of cyberattacks. Li et al. [20] presented a power control strategy for remote state estimation in the presence of DoS attacks through a stochastic game. The idea has been recently extended to address linear quadratic Gaussian control problems in [30]. Miao et al. [22] proposed a stochastic game with hybrid states for designing optimal switching control policies to counteract data injection attacks.

In this work, we employ the game-theoretic framework to study a secure co-design of control and scheduling to save communication costs and enhance CPS security. As shown in Fig. 1, we consider a remote control problem where the system state estimate is sent from the sensor to the controller across a TCP link affected by random packet dropouts. The communication channel suffers from potential DoS jamming attacks launched by an adversary. The attacker intends to degrade the control performance and increase the controller’s communica-
The contribution of this work is three-fold: (1) we establish a partially observable stochastic game with hybrid states to derive the jointly optimal scheduling and control strategy under a DoS attack in the context of remote control with random packet dropouts; (2) we prove that the design of control and scheduling can be separated without sacrificing optimality and show that the game can be reduced to a complete information stochastic game with countably infinite discrete states; (3) we show that Nash Q-learning can efficiently solve such problems and guarantee an $\epsilon$-equilibrium.

The remaining parts of this manuscript are organized as follows: Section 2 describes the model of the networked control system and the problem setup; Section 3 formulates the partially observable stochastic game and analyzes the optimality of the separate design; Section 4 presents a truncated policy derived from Nash Q-learning and proves the $\epsilon$-equilibrium of the resulting solution; Section 5 provides a numerical example to verify the theoretical results and illustrate the tradeoffs between the communication cost and the control performance; Section 6 draws conclusions.

2 PROBLEM FORMULATION

2.1 System model

Consider a linear dynamic system

$$x_{k+1} = Ax_k + Bu_k + w_k,$$  
\(y_k = Cx_k + v_k,$$  
where $x_k \in \mathbb{R}^{n_x}$, $u_k \in \mathbb{R}^{n_u}$ and $y_k \in \mathbb{R}^{n_y}$ denote the system state, input and output respectively. Without loss of generality, assume $x_0$ is a zero-mean Gaussian noise. $w_k \in \mathbb{R}^{n_w}$ and $v_k \in \mathbb{R}^{n_v}$ are i.i.d. Gaussian noises with zero mean, i.e., $\mathbb{E}(w_k) = 0$, $\mathbb{E}(v_k) = 0$, and $\mathbb{E}(w_i w_j) = \mathbb{E}(v_i v_j) = 0$ if $i \neq j$. Denote $\mathbb{E}(w_k v_k^T) = Q$ and $\mathbb{E}(v_k v_k^T) = R$, with $Q > 0$ and $R > 0$. Assume that the pair $(A, B)$ is stabilizable and $(A, C)$ is detectable.
As depicted in Fig. 1, a local sensor takes measurements of the system output \( y_k \) and generates a state estimate which is denoted as \( \hat{x}_k^* \). This estimate is wrapped as a packet and forwarded to the remote controller over a TCP link. The transmission commits to the command signal \( \nu_k \in \{0, 1\} \), which is determined by the controller and transmitted to the sensor via a reliable link with no packet dropouts. The optimization of \( \nu_k \) is detailed in Section 3. Assume the TCP link has i.i.d. packet dropouts and the probability of a successful reception is \( \lambda \). This set up is commonly seen in many remote control problems [8, 16, 17].

Define \( \gamma_k^{\epsilon} \) as an indicator variable of successful reception. We define

\[
\gamma_k^{\epsilon} = \begin{cases} 
0 & \text{if no signal is received}, \\
1 & \text{if } \hat{x}_k^* \text{ is successfully received}, 
\end{cases}
\]

and note that

\[
P(\gamma_k^{\epsilon} = 0 | \nu_k = 0) = 1, \\
P(\gamma_k^{\epsilon} = 0 | \nu_k = 1) = 1 - \lambda. 
\] (3)

The remote controller determines the input \( u_k \) and the scheduling command \( \nu_k \) based on the received information. If the packet is successfully received, then the TCP-link will send an acknowledgment back to the sensor, from which the sensor can infer the value of \( \gamma_k^{\epsilon} \) and compute its state estimates as described next.

### 2.2 State estimation

Denote the information set at the sensor and at the controller at time \( k \) as \( \mathcal{I}_k^{\epsilon} \) and \( \mathcal{I}_k \), respectively. According to the problem setup described above, we have

\[
\mathcal{I}_k^{\epsilon} = \{y_0, \ldots, y_k, \hat{x}_0^*, \ldots , \hat{x}_k^*, \nu_0, \ldots , \nu_k, \gamma_0^{\epsilon}, \ldots , \gamma_k^{\epsilon}\}, \\
\mathcal{I}_k = \{\gamma_0^{\epsilon} \hat{x}_0^*, \ldots , \gamma_k^{\epsilon} \hat{x}_k^*, \nu_0, \ldots , \nu_k, \gamma_0^{\epsilon}, \ldots , \gamma_k^{\epsilon}\}. 
\] (4)

It is easy to figure out that

\[
\mathcal{I}_k^{\epsilon} \subset \mathcal{I}_k. 
\] (5)

Denote the control policy adopted by the controller as \( \nu_k = \pi_k^{\epsilon}(\mathcal{I}_k^{\epsilon}) \). If the sensor knows the function \( \pi_k^{\epsilon} \), it can easily infer \( u_k \) from \( \mathcal{I}_k^{\epsilon} \) since \( \mathcal{I}_k^{\epsilon} \) is a subset of \( \mathcal{I}_k \) and the sensor knows all information to reconstruct \( u_k \). In this case, the separation principle holds [4]. The sensor can run a Kalman filter to obtain the optimal state estimate \( \hat{x}_k^* \), which follows \( \hat{x}_k^* = \mathbb{E}(x_k | \mathcal{I}_k^{\epsilon}, y_k, u_{k-1}) \). Specifically, we have

\[
\begin{align*}
\dot{\hat{x}}_k^* &= A \hat{x}_{k-1}^* + Bu_{k-1} + K_k \left( y_k - C (A \hat{x}_{k-1}^* + Bu_{k-1}) \right) \\
\hat{x}_k &= (I_n - K_k C) A \hat{x}_{k-1}^* + (I - K_k C) u_{k-1} - K_k v_k \\
P_k^* &= (I_n - K_k C) \left( A P_{k-1}^* A^\top + Q \right). 
\end{align*}
\] (6)

Here \( \hat{x}_k = x_k - \hat{x}_k^* \) denotes the state estimation error at the sensor [26]. \( P_k^* \) denotes the estimation error covariance defined as \( P_k^* = \mathbb{E} \left( \hat{x}_k \hat{x}_k^\top \right) \) and

\[
K_k \triangleq (AP_{k-1}^* A^\top + Q) C^\top \left( C (AP_{k-1}^* A^\top + Q) C^\top + R \right)^{-1}. 
\]

Initialize \( P_k^* \) with large values. According to [2], \( P_k^* \) converges exponentially fast. Let \( \hat{P} \) denote the steady-state error covariance after the Kalman filter reaches the steady state. In Remark 7, we analyze the rationality of assuming the sensor’s knowledge towards the control policy.

Taking into account the packet dropouts, the optimal state estimate at the controller, denoted as \( \hat{x}_k \), is derived in the following proposition.

**Proposition 1** The optimal state estimate follows the recursion

\[
\hat{x}_k = \begin{cases} 
\hat{x}_k^* & \text{if } \gamma_k^{\epsilon} = 1, \\
\hat{x}_k & \text{if } \gamma_k^{\epsilon} = 0, 
\end{cases} 
\] (7)

where

\[
\hat{x}_k = A \hat{x}_{k-1} + Bu_{k-1}. 
\] (8)

Accordingly, the expected error covariance at the controller, denoted as \( P_k \), follows

\[
P_k = \mathbb{E} ((x_k - \hat{x}_k) (x_k - \hat{x}_k)^\top | \mathcal{I}_k^{\epsilon}) = \begin{cases} 
P_k^* & \text{if } \gamma_k^{\epsilon} = 1, \\
f(P_{k-1}) & \text{if } \gamma_k^{\epsilon} = 0, 
\end{cases} 
\] (9)

where \( f(X) = AXA^\top + Q \).

**PROOF.** If \( \gamma_k^{\epsilon} = 1 \), \( \hat{x}_k^* \) is successfully received by the controller. Since \( \hat{x}_k^* \) is the optimal estimate of \( x_k \), it is easy to obtain that

\[
\hat{x}_k = \arg \min_{\hat{x}_k} \mathbb{E} ((x_k - \hat{x}_k) (x_k - \hat{x}_k)^\top | \mathcal{I}_k^{\epsilon}, u_{k-1} = \pi_{k-1}^{\epsilon}(\mathcal{I}_k^{\epsilon-1})) = \arg \min_{\hat{x}_k} \mathbb{E} ((x_k - \hat{x}_k) (x_k - \hat{x}_k)^\top | \hat{x}_k^*) = \hat{x}_k^*. 
\]

It can be further verified that the corresponding error covariance follows \( P_k = P_k^* \) [26]. Next, we detail the proof...
of the case $\gamma_k^c = 0$. Assume that the latest successful reception occurs at time $k_0$ with $k_0 < k$. Namely, $\gamma_{k_0}^c = 1$ and $\gamma_{k_0+1}, \ldots, \gamma_k$ all equal to 0. Since the command signal $\nu_k$ is determined by the controller at time $k$ before the transmission takes place. It only contains the information in $I_{k-1}^c$, rather than $I_{k}^c$ or $I_{k}^s$. Therefore, the calculation of $\hat{x}_k$ can be simplified as

$$\hat{x}_k = \arg \min_{\hat{x}_k} \mathbb{E}((x_k - \hat{x}_k)(x_k - \hat{x}_k)^\top | I_k^c)$$

$$= \arg \min_{\hat{x}_k} \mathbb{E}((x_k - \hat{x}_k)(x_k - \hat{x}_k)^\top | \hat{x}_{k_0}, u_{k_0}, \ldots, u_{k-1})$$

$$= A^{k-k_0} \hat{x}_{k_0} + \sum_{i=1}^{k-k_0} A^{k-1} Bu_{k-i}. \quad (10)$$

(10) can be written into a recursive form as given in (7). Accordingly, the evolution of the expected error covariance $P_k$ can be derived and provided in (9).

We initialize $P_0$ as $P_0^c$. Define $\tau_k$ as the holding time, which indicates the time steps since the last successful transmission, i.e.

$$\tau_k = \begin{cases} 0 & \text{if } \gamma_k^c = 1, \\ \tau_{k-1} + 1 & \text{if } \gamma_k^c = 0, \end{cases} \quad (11)$$

From (9) and (11), we can express $P_k$ in terms of $\tau_k$ as $P_k = f^{\tau_k}(P_{k-\tau_k}^c)$, which can be further simplified as $P_k = f^{\tau_k}(P_{\tau_k}^c)$ after the Kalman filter has reached the steady state. Our subsequent analysis uses the following properties of the state estimate and the estimation error.

**Lemma 2.** The following facts are true:

(a) $\mathbb{E}((x_k - \hat{x}_k)(x_k - \hat{x}_k)^\top | \hat{x}_k) = 0$;
(b) $\mathbb{E}((\hat{x}_k - \bar{x}_k)(\hat{x}_k - \bar{x}_k)^\top | \bar{x}_k) = 0$;
(c) $\mathbb{E}((\hat{x}_k - \bar{x}_k)(\hat{x}_k - \bar{x}_k)^\top | \bar{x}_k) = f(P_{k-1}) - P_k^c$.

**PROOF.** See Appendix A.

### 2.3 Model of the adversary

Consider an adversary that intends to degrade the control performance by deteriorating the availability of the communication channel. In particular, the adversary may launch a DoS attack, e.g., by increasing the noise at the receiver, and decreasing the successful transmission probability [20]. We model the DoS attack as a binary process where the successful transmission probability is lowered to $\lambda_a < \lambda$ when an attack is launched and remains unchanged otherwise.

Let $a_k$ be an indicator variable of the DoS attack, i.e.

$$a_k = \begin{cases} 1 & \text{if an attack is launched at time } k, \\ 0 & \text{if no attack is launched at time } k. \end{cases}$$

In view of (3), we have

$$\begin{align*}
\mathbb{P}(\gamma_k^c = 0 | (\nu_k, a_k) = (0, j)) &= 1, \quad j \in \{0, 1\}, \\
\mathbb{P}(\gamma_k^c = 0 | (\nu_k, a_k) = (1, 0)) &= 1 - \lambda, \\
\mathbb{P}(\gamma_k^c = 0 | (\nu_k, a_k) = (1, 1)) &= 1 - \lambda_a.
\end{align*} \quad (12)$$

From (3), (9) and (12), the evolution of the error covariance $P_k$ in the presence of an attacker can be modeled as a Markov decision process (MDP) where $P_k$ is taken as the state, and the pair $(\nu_k, a_k)$ is taken as the action. State transition probabilities are computed accordingly as

$$\begin{align*}
\mathbb{P}(P_{k} = f(P_{k-1}) | P_{k-1}, (\nu_k, a_k) = (0, j)) &= 1, \quad j \in \{0, 1\}, \\
\mathbb{P}(P_{k} = f(P_{k-1}) | P_{k-1}, (\nu_k, a_k) = (1, 0)) &= 1 - \lambda, \\
\mathbb{P}(P_{k} = f(P_{k-1}) | P_{k-1}, (\nu_k, a_k) = (1, 1)) &= 1 - \lambda_a, \\
\mathbb{P}(P_{k} = P_{k-1}^c | P_{k-1}, (\nu_k, a_k) = (1, 0)) &= \lambda, \\
\mathbb{P}(P_{k} = P_{k-1}^c | P_{k-1}, (\nu_k, a_k) = (1, 1)) &= \lambda_a.
\end{align*} \quad (13)$$

In addition, assume that the attacker can overhear the acknowledgments sent from the controller to the sensor and make use of them to efficiently schedule the attacks. Denote the information available to the attacker at time $k$ as $I_k^a$. We then have

$$I_k^a = \{\gamma_0, \ldots, \gamma_k\} \quad (14)$$

and, when comparing with (4), note that the attacker has significantly less information than the controller-scheduler.

### 3 Co-design of control and scheduling

In this section, we formulate the interactive decision making process as a partially observable stochastic game and show that, from the controller’s perspective, the design of the control and scheduling can be conducted separately without sacrificing optimality. This property simplifies the game into a complete information stochastic game.

#### 3.1 Design of the partially observable stochastic game

Consider that the legitimate controller aims at optimizing the control performance by using the information
in $T^c_{k-1}$ to determine the transmission command $\nu_k$ and adjust the input $u_k$. In particular, the controller intends to minimize a linear combination of a quadratic term of the state $x_k$ and the input $u_k$, as well as the controller’s transmission cost and the negative of the adversary’s attacking cost:

$$J = \frac{1}{K} \mathbb{E}\left( \sum_{k=0}^{K-1} x_k^\top W x_k + u_k^\top U u_k + c_s \nu_k - c_a a_k \right)$$  \hspace{1cm} (15)

Here $c_s$ denotes the cost of transmission generated by the controller and the sensor, $c_a$ denotes the cost of launching a jamming attack. $W$ and $U$ are positive definite matrices. Conversely, the adversary intends to maximize $J$ by smartly scheduling the attack $a_k$ according to its observations $T^c_{k-1}$.

Assume the two agents select their actions independently. The interaction can then be formulated as a zero-sum stochastic game. Specifically, we denote the stochastic game $G$ as a tuple $(I, S, \{\mu^0\}, \mathcal{A}_c, \mathcal{A}_a, \mathcal{O}_c, \mathcal{O}_a, \mathcal{P}, C)$, where

- $I$ denotes the number of agents. Here we have two agents, i.e., the controller and the attacker.
- $S$ denotes the set of states $s_k$. Here we define

$$s_k = [x_k, P_{k-1}]$$

where $x_k$ is the system state vector taking values from the continuous space $\mathbb{R}^{n_x}$ and the error covariance $P_{k-1}$ takes values from a countable set

$$I^c = \{ P^c_{k}, f(P^c_{k}), \ldots, f^{K-k}(P^c_{k}), \ldots \}$$  \hspace{1cm} (16)

with $k \in [0, K]$. See (9).

- $\{\mu^0\}$ denotes the initial value of the state.
- $\mathcal{A}_c$ denotes the action space of the controller. In our problem, the controller has two actors, i.e., the system input $u_k$ which is continuous and the scheduling command $\nu_k \in \{0, 1\}$.
- $\mathcal{A}_a$ denotes the action space of the attacker, which is $a_k \in \{0, 1\}$.
- $\mathcal{O}_c$ denotes the set of observations available to the controller. In accordance with the causality, the controller determines the transmission command first and update the system input $u_k$ based on the latest state estimate $\hat{x}_k$ if the packet is successfully received. Therefore, the information available to the determination of $\nu_k$ should be $I^c_{k-1}$, while the information available to the determination of $u_k$ is $I^c_k$. According to Proposition 1, the optimal estimate of $x_k$ from $T^c_{k-1}$ is $\hat{x}_k$ and the error covariance $P_{k-1}$ can be inferred from the sequence of $\gamma^c_k$. We define the observation of the act $\nu_k$ as

$$o^\nu_k = \{\hat{x}_k, P_{k-1}\}.$$  

Given $T^c_k$, $\gamma^c_k$ is available to the controller if $\gamma^c_k = 1$. Therefore, we define the observation of the act $u_k$ as

$$o^u_k = \{\hat{x}_k, \gamma^c_k \hat{x}_k, P_{k-1}\}.$$  

In addition, the error covariance $P^c_k$ is taken as common knowledge to all agents.

- $\mathcal{O}_a$ denotes the set of observations available to the attacker. As detailed in Section 2.3, the set $T^a_{k-1}$ collects all the information available to the adversary, from which only $P_{k-1}$ can be inferred according to (9). Therefore, we define the observation of the adversary as

$$o^a_k = \{P_{k-1}\}.$$  

- $\mathcal{P}$ is the transition probabilities. From (1), we may notice that the transition probability of $x_k$ depends on $u_{k-1}$ and the distribution of $u_{k-1}$:

$$\mathbb{P}(x_k \mid x_{k-1}, u_{k-1}) \sim \mathcal{N}(Ax_{k-1} + Bu_{k-1}, Q).$$

In addition, the evolution of $P_k$ depends on the actions $\nu_k$ and $a_k$ as given in (13).

- $C$ denotes the cost function. From (15), we have the immediate cost at time $k$ given as

$$c_k = x_k^\top W x_k + u_k^\top U u_k + c_s \nu_k - c_a a_k.$$  

The controller aims at minimizing the cost while the attacker aims at maximizing the cost.

**Remark 3** Note that the controller can infer $\hat{x}_k$ and $P_{k-1}$ from $T^c_{k-1}$ according to (4) and (8). After the transmission action $\nu_k$ is determined, the controller has a stochastic access to $\hat{x}_k$ such that $\hat{x}_k$ is available. Beyond that, the attacker can infer $P_{k-1}$ from $T^a_{k-1}$. Since neither of the two agents have complete information of the state $s_k$, the game is a partially observable stochastic game with a hybrid of discrete and continuous states [22]. In the following section, we employ dynamic programming to show that the design of control and scheduling can be conducted separately without sacrificing optimality. More importantly, this enables us to simplify the problem into a complete information game with countably infinite discrete states that can be efficiently solved.

### 3.2 Separation of scheduling and control

In this section, we first consider the case where the time horizon $K$ in (15) is finite, and then extend the conclusions to infinite horizons, i.e. $K \to \infty$. To state our first result, we define

$$S_K = W, \quad F_k = (B^\top S_{k+1} B + U)^{-1} B^\top S_{k+1} A,$$

$$S_k = A^\top S_{k+1} A + W - A^\top S_{k+1} B F_k,$$  \hspace{1cm} (17)

for $k = K - 1, \ldots, 0$. 

5
Theorem 4 For the stochastic game $G$ with a finite $K$, the stationary Nash equilibrium exists and the design of the control and scheduling can be conducted separately. In particular, the optimal control policy is in the form of
\[ u_k^* = -F_k \bar{x}_k, \]
and the optimal transmission policy and the attack policy, denoted as $\pi_k^*$ and $\pi_k^a$, respectively, can be derived by solving the following minimax optimization problem,
\[
\min \{ \nu_k = \pi_k^*(I_{K-1}) \} \{ a_k = \pi_k^a(I_{K-1}) \} \sum_{k=0}^{K-1} \left( \text{tr}(\sum \nu_k S_{k+1} A + W - S_k) \times \mathbb{E}(P_k) \right) + c_o \nu_k - c_o a_k \}
\]
Moreover, the resulting optimal policies depend on the triplet $(P_{k-1}, P_k^c, k)$, such that the game $G$ is equivalent to a complete-information stochastic game with finite discrete states.

PROOF. From the controller’s perspective, the game can be formulated in a recursive form as
\[
J_k^c = x_k^T S_k x_k, \quad J_k^c = \min_{\pi_k^a(o_k^a)} \mathbb{E}\left( \max_{\pi_k^a(o_k^a)} \mathbb{E}(c_k + J_{k+1}^c) \right),
\]
with $k = K - 1, \ldots, 0$. Similarly, from the adversary’s perspective, the game can be expressed as
\[
J_k^o = x_k^T S_k x_k, \quad J_k^o = \max_{\pi_k^a(o_k^a)} \mathbb{E}\left( \min_{\pi_k^a(o_k^a)} \mathbb{E}(c_k + J_{k+1}^o) \right).
\]
First, set $k = K - 1$. Denote $\pi_k^a(o_k^a)$ as the optimal policy adopted by the adversary, (20) can be equivalently expressed as
\[
J_k^o = \min_{\nu_k} \mathbb{E} \left( \min_{u_k} \mathbb{E} \left( \mathbb{E}(x_k^T W x_k + u_k^T U u_k + c_o \nu_k - c_o a_k + x_{k+1}^T S_{k+1} x_{k+1} | a_k = \pi_k^a(o_k^a) | o_k^a) | o_k^a \right) \right)
\]
\[
= \min_{\nu_k} \mathbb{E} \left( \min_{u_k} \mathbb{E} \left( \mathbb{E}(x_k^T W x_k + u_k^T U u_k + c_o \nu_k - c_o a_k + x_{k+1}^T S_{k+1} x_{k+1} | \gamma_k^c, \bar{x}_k, \gamma_{k+1}^c, \bar{x}_{k+1} | a_k = \pi_k^a(P_{k-1}), \bar{x}_k, P_{k-1}) \right) \right).
\]
(23) holds since the attacker’s action $a_k$ only has an impact on $\gamma_k^c$ as shown in (12).

According to Proposition 1 in [9], the optimal input is given as
\[
u_k = \begin{cases} -F_k \bar{x} & \gamma_k^c = 0 \\ -F_k \bar{x}_k & \gamma_k^c = 1 \end{cases},
\]
where the optimal feedback gain is independent of packet arrivals.

Substitution of (24) into (23) provides
\[
J_k^o = \min_{\nu_k} \mathbb{E} \left( (1 - \gamma_k^c) J_k^c | \gamma_k^c = 0 + \gamma_k^c J_k^c | \gamma_k^c = 1 \right),
\]
with $a_k = \pi_k^a(P_{k-1})$,
\[
J_k^c | \gamma_k^c = 0 = \bar{x}_k^T S_k \bar{x}_k + \text{tr}(W + A^T S_{k+1} A) P_k + \nu_k (1 - a_k) + c_o \nu_k - c_o a_k,
\]
\[
J_k^c | \gamma_k^c = 1 = \bar{x}_k^T S_k \bar{x}_k + \text{tr}(W + A^T S_{k+1} A) P_k^c + \nu_k (1 - a_k) + c_o \nu_k - c_o a_k.
\]
In view of the relationship between $\gamma_k^c$ and $(\nu_k, a_k)$ given in (12), $J_k^o$ is written as
\[
J_k^o = \min_{\nu_k} \mathbb{E} \left( (1 - \nu_k) \gamma_k^c | \bar{x}_k^T S_k \bar{x}_k + \text{tr}(W + A^T S_{k+1} A) P_k + \nu_k (1 - a_k) + c_o \nu_k - c_o a_k, \gamma_k^c | \bar{x}_k^T S_k \bar{x}_k + \text{tr}(W + A^T S_{k+1} A) P_k^c + \nu_k (1 - a_k) + c_o \nu_k - c_o a_k \right).
\]
We can further prove with Lemma 2 (b) and (c) that
\[
\mathbb{E}(\bar{x}_k^T S_k \bar{x}_k | x_k, P_{k-1}) - \mathbb{E}(\bar{x}_k^T S_k | x_k, P_{k-1}) \times (x_k - \bar{x}_k) \times \mathbb{E}(\bar{x}_k^T S_k | x_k, P_{k-1}) \times \mathbb{E}(\bar{x}_k^T S_k | x_k, P_{k-1}) + \text{tr}(S_k f(P_{k-1})),
\]
and
\[
\mathbb{E}(\bar{x}_k^T S_k \bar{x}_k | x_k) = \mathbb{E}(\bar{x}_k^T S_k \bar{x}_k | x_k) + \text{tr}(S_k f(P_{k-1})),
\]

\[
= \mathbb{E}(\bar{x}_k^T S_k \bar{x}_k | x_k) + \text{tr}(S_k f(P_{k-1})),
\]

\[
= \mathbb{E}(\bar{x}_k^T S_k \bar{x}_k | x_k) + \text{tr}(S_k f(P_{k-1})),
\]
Substitution of (26) and (27) into (25) yields that
\begin{align*}
J_k^* &= E(x_k^T S_k x_k + \bar{\nu}_k + \nu_k \lambda_k - c_a a_k + \max_{u_k} E \left[ (x_{k+1}^T S_{k+1} x_{k+1} + \nu_{k+1} - c_a a_{k+1} + \nu_k \lambda_{k+1} - \nu_{k+1} \lambda_k ) f(P_{k+1}) - (1 - \nu_k - \lambda_k) \right] \\
&= E(x_k^T S_k x_k + \nu_k \lambda_k - c_a a_k + \max_{u_k} E \left[ (x_{k+1}^T S_{k+1} x_{k+1} + \nu_{k+1} - c_a a_{k+1} + \nu_k \lambda_{k+1} - \nu_{k+1} \lambda_k ) f(P_{k+1}) - (1 - \nu_k - \lambda_k) \right]
\end{align*}
Grouping all the terms that contain \( \nu_k \), we may notice that the coefficients of \( \nu_k \) are independent of the state estimate \( \bar{x}_k \). This important property indicates that the optimal scheduling command \( \nu_k \) is independent of \( \bar{x}_k \). Next, we look at \( J_k^* \). Similar to the derivation of \( J_k^* \), we have
\begin{align*}
J_k^* &= \max_{u_k} E \left[ (x_k^T W x_k + u_k^T U u_k + c_a a_k + \nu_k \lambda_k - \nu_k \lambda_k ) f(P_{k+1}) - (1 - \nu_k - \lambda_k) \right]
\end{align*}

Compare (28) and (29). \( J_k^* \) minimizes a convex function and \( J_k^* \) maximizes a concave function. According to Kakutani fixed-point theorem [12], a Nash-equilibrium exists. For \( k \in [0, K-1] \), the existence of the Nash-equilibrium can be proved in the same way.

To complete the proof, note that in (28) and (29), the decision variables both depend on \( P_{k-1} \) and the matrix \( W + A^T S_{k+1} A \), which is a function of time \( k \). Beyond that, when \( k \in [0, K-1] \), \( P_{k-1} \) takes values from a finite set according to (6) and (9) and its value is available in both \( \nu_k^* \) and \( \nu_k^* \). Hence, the game is a complete-information game with finite discrete states. Since the schedule commands \( \nu_k \) and \( a_k \) are both independent of the state estimate \( \bar{x}_k \), from [18, Theorem 1], the optimal cost is in the form of
\begin{align*}
E(x_0^T S_0 x_0) + \sum_{k=0}^{K-1} \text{tr} (S_{k+1} Q) + \sum_{k=0}^{K-1} \left( \text{tr} (A^T S_{k+1} A + W - S_k) E(P_k) \right) + c_a a_k - c_a a_k
\end{align*}

Since the first two terms are irrelevant to \{\( \nu_k \)\} and \{\( a_k \)\}, the solution to (19) is the same as that to (30).

**Remark 5** Theorem 4 establishes that in the present setting, the optimal scheduling command \( \nu_k \) is independent of the state estimate. This fundamental property is often assumed without detailed justification in co-design works, e.g., [18].

Our result guarantees that the separate design does not sacrifice optimality when the controller works as the scheduler. However, if the sensor schedules, such as considered in [14, 30], then this separation property does not hold. In fact, consider that \( \nu_k \) is determined by the sensor, so that \( \bar{x}_k \) and \( \bar{x}_k \) are available before the decision is made. \( \nu_k^* \) contains information, and therefore the expectations in (23) should be computed conditioned on both \( \bar{x}_k \) and \( \bar{x}_k \). Since \( \nu_k \) contains information of \( \bar{x}_k^* \), the optimal estimate at the controller does not follow (7), but becomes a nonlinear estimation problem, as analyzed in [19, 28], and will further complicate the design of the feedback gain.

Theorem 4 can be easily extended to infinite horizons provided that the value of \( J \) is bounded.

**Corollary 6** Suppose that
\begin{align*}
\lambda_a > 1 - \frac{1}{|\sigma_{\max}(A)|^2},
\end{align*}
where \( |\sigma_{\max}(A)| \) denotes the spectral radius of \( A \). Then the infinite horizon stochastic game \( G \) has a Nash equilibrium. The optimal control policy is given as
\begin{align*}
\nu_k^* = -(B^T S_{\infty} A + U)^{-1} B^T S_{\infty} A \bar{x}_k,
\end{align*}
where, cf (17),
\begin{align*}
S_{\infty} = A^T S_{\infty} A + W - A^T S_{\infty} B (B^T S_{\infty} A + U)^{-1} B^T S_{\infty} A.
\end{align*}
The optimal scheduling policies can be derived from
\begin{align*}
\min_{\{v_k = \pi^*(P_{k-1})\}} \max_{\{a_k = \pi^*(P_{k-1})\}} \tilde{J}(\pi^*, \pi^*) = \lim_{K \to \infty} \frac{1}{K} \sum_{k=0}^{K-1} \left( c_a \nu_k - c_a a_k + \text{tr} (M_\infty \text{E}(P_k)) \right),
\end{align*}
where \( M_\infty = A^T S_{\infty} A + W - S_{\infty} \) and \( P_k \) takes values from the countably infinite set \{\( P, f(P), f^2(P), \ldots \)\} cf (16).

In Corollary 6, (31) gives a sufficient condition to ensure the boundedness of (33), since even if the attacker launches the DoS attack at all time instances, the expected estimation error covariances are ensured to be bounded. Given that the Kalman filter converges exponentially fast, we can study the infinite horizon optimization problem in (33) by assuming the steady state of the Kalman filter, i.e., \( P_k^* = \bar{P} \). Moreover, since the
feedback gain in (32) also converges to a constant matrix, the scheduling policies become time-independent. Therefore, the infinite horizon stochastic game \( G \) is equivalent to a complete-information game with countably infinite states.

**Remark 7** We may notice that the control policy \( \pi_k^\tau_\bar{\tau} \) is a standard state feedback strategy with time dependent feedback gain (the finite horizon case) or constant feedback gain (the infinite horizon case) according to Theorem 4 and Corollary 6. The gain is independent of the packet dropout \( \gamma_k \) as well as the attacker’s policy. Therefore, the controller can synchronize the control policy with the sensor before the process starts such that the separation principle holds at the sensor.

4 Nash Q-learning

In this section, we detail the algorithms to solve the stochastic game \( G \) in practice. If \( K \) is finite, the problem can be easily solved using the idea of dynamic programming. We focus on the case where \( K \) is infinite.

Nash Q-learning is a well known multi-agent reinforcement learning algorithm for stochastic games [11, 20]. However, it is only applicable to MDPs with finite states and discounted costs. To apply Nash Q-learning to the current setup, we have to do two modifications to the original problem in (33): (1) aggregate the states of the MDP to make the states finite; (2) approximate the average cost with a discounted cost. In particular, we consider a stochastic game, denoted as \( G^N \), in the form of

\[
J = \min_{\{\nu_k = \pi(\bar{\tau}_{k-1})\}} \max_{\{a_k = \pi(a_{k-1})\}} \lim_{K \to \infty} \sum_{k=0}^{K-1} \eta^k \mathbb{E} \left[ c_{\nu_k} - c_{a_k} \epsilon_k + \text{tr} \left( M_{\infty} f^{\bar{\tau}_k}(\mathbf{P}) \right) \right],
\]

where the discount factor \( 0 < \eta < 1 \) and \( \bar{\tau}_k \) follows, cf. (11),

\[
\bar{\tau}_k = \begin{cases} 0 & \nu_k \gamma_k = 1 \\ \max \{\bar{\tau}_{k-1} + 1, N - 1\} & \nu_k \gamma_k = 0 \end{cases}.
\]

In (34), we replace \( \mathbb{E}(P_k) \) with \( \mathbb{E}(f^{\bar{\tau}_k}(\mathbf{P})) \) since there is a one-to-one correspondence between \( P_k \) and the holding time \( \bar{\tau}_k \) according to (11). Compare (35) and (11).

In (35), the states with \( \gamma_k \geq N - 1 \) are aggregated as a single state. In this way, we keep states in the MDP to facilitate the application of Nash Q-learning. In particular, we learn the Q-function, denoted as \( Q(\bar{\tau}, \nu_k, a_k) \), for \( \bar{\tau} \in [0, N - 2] \) according to the Nash Q-learning algorithm presented in [20]. The transmission and attack policies for \( \bar{\tau} \in [0, N - 2] \) are derived from (36)

\[
\begin{align*}
\pi_N^\tau(\bar{\tau}) &= \arg \min_{\pi_N} \max_{\pi_N} \sum_{\nu_k, a_k} Q(\bar{\tau}, \nu_k, a_k) p(\nu_k|\pi_N) p(a_k|\pi_N), \\
\pi_N^\tau(\bar{\tau}) &= \arg \max_{\pi_N} \min_{\pi_N} \sum_{\nu_k, a_k} Q(\bar{\tau}, \nu_k, a_k) p(\nu_k|\pi_N) p(a_k|\pi_N).
\end{align*}
\]

Here \( \pi_N^\tau(\bar{\tau}) \) and \( \pi_N^\tau(\bar{\tau}) \) specify \( P(\nu_k | \bar{\tau}_{k-1} = \bar{\tau}) \) and \( P(a_k | \bar{\tau}_{k-1} = \bar{\tau}) \) for each \( \bar{\tau} \in [0, N - 2] \) and each \( \nu_k, a_k \in \{0, 1\} \). For \( \bar{\tau} = N - 1, \) we set

\[
\begin{align*}
P(\nu_k = 1 | \bar{\tau}_{k-1} = N - 1) &= 1, \\
P(a_k = 1 | \bar{\tau}_{k-1} = N - 1) &= 1.
\end{align*}
\]

Next, we show the \( \epsilon \)-optimality of the truncated policy.

**Theorem 8** Denote the truncated policy in (36) as \( (\pi_N^\tau, \pi_N^\tau) \). Then, when \( \eta \) is close to 1 and \( N \) is large, the pair \( (\pi_N^\tau, \pi_N^\tau) \) provides an \( \epsilon \)-Nash equilibrium of the game \( G \), i.e.

\[
\begin{align*}
J(\pi_N^\tau, \pi_N^\tau) &\leq \min_{\pi_N} J(\pi_N, \pi_N^\tau) + \epsilon_{\eta, N}, \\
J(\pi_N^\tau, \pi_N^\tau) &\geq \max_{\pi_N} J(\pi_N^\tau, \pi_N) - \epsilon_{\eta, N},
\end{align*}
\]

where \( \lim_{\eta \to 1, N \to \infty} \epsilon_{\eta, N} = 0 \) and \( J \) is defined in (33).

**PROOF.** See Appendix B.

Theorem 8 indicates that the truncated policy derived from the discounted cost can make the value of the game arbitrarily close to the Nash-equilibrium by increasing the truncation horizon \( N \) and controlling the discounted factor \( \eta \) close to 1.

5 Simulation

Consider an infinite time horizon co-design problem with the system matrices set as

\[
A = \begin{bmatrix} 1.2 & 1 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 2 \end{bmatrix}, \quad C = \begin{bmatrix} 1 \\ 1 \end{bmatrix}^T.
\]

The noise covariances are taken as \( Q = I_{n_x} \) and \( R = I_{n_u} \). The weight matrices are set as \( U = I_{n_x} \) and \( W = I_{n_u} \). When there is no attack, the successful transmission rate is \( \lambda = 0.95 \) and the transmission cost is set as \( c_s = 2000 \). Accordingly, the optimal control law is computed from (32) as \( L = \begin{bmatrix} -0.39 & -0.63 \end{bmatrix} \).
We study the interaction between the attacker and the controller. Fix the attacking cost as \( c_a = 2000 \). At instances that an attack is launched, the successful transmission rate drops to \( \lambda_s \). We vary the parameter \( \lambda_s \) from 0.3 to 0.9 to simulate attackers with different level of interfere ability. A small \( \lambda_s \) indicates that the attacker is powerful to cause significant packet dropouts. For each \( \lambda_s \), the game is solved correspondingly via the Nash-Q learning in [20] with the truncation horizon \( N = 50 \) and the discount factor \( \eta = 1 - (0.1)^8 \). The resulted transmission policies present a threshold-type structure, i.e.

\[
\nu_k = \begin{cases} 
0 & \bar{\tau} < \text{thres}_c \\
1 & \bar{\tau} \geq \text{thres}_c 
\end{cases} \quad \text{and} \quad a_k = \begin{cases} 
0 & \bar{\tau} < \text{thres}_a \\
1 & \bar{\tau} \geq \text{thres}_a 
\end{cases}
\]

(40)

We run a Monte Carlo simulation with the derived transmission and attack policies with the simulation length \( T \) taken as \( 10^6 \). As shown in Fig. 2(a), the averaged cost \( \bar{J} \) defined in (33) monotonically decreases with the increase of \( \lambda_s \), indicating that the weaker the attacker is, the more easier the controller achieves a smaller cost. From Fig. 2 (b), we may note that the attack threshold \( \text{thres}_a \) is always greater than or equal to the transmission threshold \( \text{thres}_c \) to save the attacking cost. Meanwhile, we calculate the averaged transmission probability \( p_c \) and averaged attack probability according to

\[
p_c = \frac{\sum_{k=0}^{T} \nu_k}{T+1} \quad \text{and} \quad p_a = \frac{\sum_{k=0}^{T} a_k}{T+1}.
\]

Their trajectories are plotted in Fig. 2 (b) and (c). When \( \lambda_s \) gets larger than 0.45, the attacker chooses to stay silent in most of the time since it can hardly cause a significant degrade of the controller’s tracking performance while paying a high cost on attacking. From the controller’s perspective, the transmission probability \( p_c \) is decreasing when \( \lambda_s \geq 0.38 \) to save transmission cost when the attacker becomes inactive.

This numerical experiment shows that both the controller and the attacker achieve the Nash-equilibrium by learning from the interactions between each other without the knowledge of either \( \lambda \) or \( \lambda_s \). This is the main advantage of the game theoretic approaches. To verify this, we compare with the conventional fix-threshold approach. Assume that the adversary is smart and can adapt its own policy according to the controller’s via the standard Q-learning. In Fig. 3, we compare the averaged cost \( \bar{J} \) corresponding to \( \text{thres}_c = 1, 3, 6 \) with the cost \( \bar{J} \) derived from the Nash Q-learning. We may note that the policy derived with the Nash-Q learning always outperform the fixed threshold policy.

6 Conclusions

In the present work, we have studied the design of control and scheduling in the presence of stochastic DoS attacks. We have designed a zero-sum partially observable stochastic game to jointly optimize the control and scheduling strategies. We have established that the optimal transmission scheduling command is independent of the state estimate and that the design of control and scheduling can be conducted separately without sacrificing optimality. Based on this property, the partially observable game is simplified as a complete information game. Moreover, we have proved that by applying a Nash Q-learning to the stochastic game with appropriately truncated states, an \( \epsilon \)-equilibrium can be derived. Future works may consider employing the sensor as the scheduler, in which case more information can be exploited for scheduling and enhanced security can be expected.
A  Proof of Lemma 2

PROOF. Item (a) is the same as Lemma 4.1 in [25]. Item (b) can be proved based on [7]. For that purpose, define

\[ r_{k-1} \triangleq K_k C A \tilde{x}_{k-1} + K_k C w_{k-1} + K_k v_k. \]

Assume that the latest successful transmission occurs at time \( k_0 \) where \( k_0 \leq k \). According to Eq. (27) in [7], we have

\[ \tilde{x}_k^s - \tilde{x}_k = \sum_{i=k_0}^{k-1} A^{k-1-i} r_i. \]

Note that \( r_i \) is independent of \( \tilde{x}_k^s \) for any \( k_0 \leq i \leq k-1 \), and \( \tilde{x}_k \) and \( v_i \) both depend on \( \tilde{x}_k^s \). It can be concluded that \( \tilde{x}_k^s - \tilde{x}_k \) is independent of \( \tilde{x}_k \) and therefore Item (b) is proved.

Moreover, from Item (a) and (b), we have \( \mathbb{E}(\tilde{x}_k \tilde{x}_k^T \mid \tilde{x}_k) = 0 \), i.e., \( \tilde{x}_k \) is independent of \( \tilde{x}_k \). Hence, we have \( \mathbb{E}(\tilde{x}_k \tilde{x}_k^T \mid \tilde{x}_k) = \mathbb{E}(\tilde{x}_k \tilde{x}_k^T) = P_k^s \). Then, Item (c) can be proved as follows:

\[
\begin{align*}
\mathbb{E}((\tilde{x}_k^s - \tilde{x}_k)(\tilde{x}_k^s - \tilde{x}_k)^T \mid \tilde{x}_k) &= \mathbb{E}((\tilde{x}_k^s - x_k + x_k - \tilde{x}_k)(\tilde{x}_k^s - x_k + x_k - \tilde{x}_k)^T \mid \tilde{x}_k) \\
&= \mathbb{E}(\tilde{x}_k \tilde{x}_k^T + (x_k - \tilde{x}_k)(x_k - \tilde{x}_k)^T - 2 \tilde{x}_k(x_k - \tilde{x}_k)^T \mid \tilde{x}_k) \\
&= \mathbb{E}((x_k - \tilde{x}_k)(x_k - \tilde{x}_k)^T - \tilde{x}_k \tilde{x}_k^T \mid \tilde{x}_k) = f(P_k-1) - P_k^s.
\end{align*}
\]

B  Proof of Theorem 8

PROOF.

Denote \( J_{N,N}^{\pi,\pi} \) as the value of (B.1)

\[
\lim_{K \to \infty} (1 - \eta) \sum_{k=0}^{K-1} \eta^k \mathbb{E} \left[ c_s V_k - c_a a_k + \operatorname{tr} \left( M_\infty f^r(\tilde{P}) \right) \right] \tag{B.1}
\]

when \( (\pi^*, \pi^*) \) is applied. Since \( \pi^* \) and \( \pi^* \) give the stationary Nash equilibrium, we have

\[
\max_{\pi_N} J_{N,N}^{\pi,\pi} \leq J_{N,N}^{\pi,\pi} \leq \min_{\pi_N} J_{N,N}^{\pi,\pi}. \tag{B.2}
\]

Next, we prove (38) by showing that there exists a scalar \( \epsilon_{\eta,N} \) such that \( \lim_{\eta \to 1, N \to \infty} \epsilon_{\eta,N} = 0 \) and

\[
\left| J_{N,N}^{\pi,\pi} - \min_{\pi} \tilde{J}(\pi, \pi^*) \right| < \frac{1}{2} \epsilon_{\eta,N}, \tag{B.3}
\]

\[
\left| J_{N,N}^{\pi,\pi} - \tilde{J}(\pi, \pi^*) \right| < \frac{1}{2} \epsilon_{\eta,N}. \tag{B.4}
\]

First, fix the attacker’s policy to be \( \pi^*_N \). Define

\[
\pi^{c_0} = \arg \min_{\pi} \tilde{J}(\pi, \pi^*), \quad \pi^{c_1} = \arg \min_{\pi} J_{\eta,\infty}^{\pi,\pi^*}.
\]

According to the boundedness of \( \tilde{J} \) and Abel’s theorem [10], for any \( \eta < 1 \), there exists a scalar \( \bar{\epsilon}_\eta > 0 \), such that

\[
|\tilde{J}(\pi^{c_0}, \pi^*_N) - J_{\eta,\infty}^{\pi,\pi^*}| \leq \bar{\epsilon}_\eta, \tag{B.5}
\]

and \( \lim_{\eta \to 1} \bar{\epsilon}_\eta = 0 \). Eq. (B.5) shows that the optimality gap induced by the discount factor diminishes when \( \eta \) goes to 1. Following this, we further analyze the optimality gap induced by state truncation.

From the transmission policy \( \pi^*_N \), we can derive the occupation measures, denoted as \( \bar{\omega}(\bar{\tau}, (\nu, a)) \), for the corresponding truncated MDP. Here \( \bar{\tau} \in [0, N-1] \) and \( \bar{\omega} \) can be interpreted as the total expected discount time spent in the state-action pairs \((\bar{\tau}, (\nu, a)) \) (Section 3.1 in [1]). Inspired by the Bellman equation, the occupation measures satisfy that

\[
\sum_{\bar{\tau} \in [0, N-1]} \sum_{\nu, a \in \{0,1\}} \bar{\omega}(\bar{\tau}, (\nu, a)) = 1, \quad \bar{\omega}(\bar{\tau}, (\nu, a)) \geq 0
\]

\[
\sum_{\bar{\tau} \in [0, N-1]} \sum_{\nu, a \in \{0,1\}} \bar{\omega}(\bar{\tau}, (\nu, a))(1(\bar{\tau'} = \bar{\tau}) - \eta P_{\bar{\tau}(\nu, a)\bar{\tau}'})
\]

\[
= (1 - \eta)1(\bar{\tau'} = 0), \tag{B.6}
\]

Here \( 1(\cdot) \) denotes an indicator function. \( P_{\bar{\tau}(\nu, a)\bar{\tau}'} \) is a short form for the transition probability \( \mathbb{P}(s_{k+1} = \bar{\tau}' \mid s_k = \bar{\tau}, \nu_k = \nu, a_k = a) \). According to (B.6) and the transition probability given in (13), we have

\[
\sum_{\nu, a \in \{0,1\}} \bar{\omega}(\bar{\tau} + 1, (\nu, a)) = \eta \sum_{a \in \{0,1\}} \bar{\omega}(\bar{\tau}, (\nu = 0, a))
\]

\[
+ \eta \lambda \bar{\omega}(\bar{\tau}, (\nu, a) = (1,1)) + \eta \lambda \bar{\omega}(\bar{\tau}, (\nu, a) = (1,0)). \tag{B.7}
\]

Since \( 0 < \lambda_a < \lambda < 1 \), \( \bar{\omega} \) exponentially decreases, i.e.

\[
\sum_{\nu, a \in \{0,1\}} \bar{\omega}(\bar{\tau} + 1, (\nu, a) \leq \eta \sum_{\nu, a \in \{0,1\}} \bar{\omega}(\bar{\tau}, (\nu, a))
\]

In addition, the transmission policy can be derived from \( \bar{\omega} \) as

\[
\mathbb{P}(\nu_k, a_k \mid \bar{\tau}) = \frac{\bar{\omega}(\bar{\tau}, (\nu, a))}{\sum_{\nu_k, a_k \in \{0,1\}} \bar{\omega}(\bar{\tau}, (\nu, a))}. \tag{B.8}
\]

The value of the cost functions can be expressed in terms
of $\bar{\omega}$ as
\[ J_{\pi_N^{\infty}}^{\eta, N}(\pi_N^{\infty}, \pi_N^{\infty}) = \sum_{N=1}^{\infty} \sum_{\tau \geq 0} \omega(\bar{\tau}, (\nu, a)) f^\tau(\bar{P}). \quad \text{(B.9)} \]

Therefore, minimizing $J$ over $\pi_N$ is equivalent to minimizing it over $\omega$.

Next, we consider the non-truncated MDP corresponding to the transmission policy $\pi_N^{\infty}$. Denote the associated occupation measure as $\omega(\tau, (\nu, a))$, where $\tau \in [0, \infty)$. Then, $\omega$ satisfies (B.6) with $N = \infty$. Moreover, similar to (B.8), from (37) we have
\[ \omega(\tau, (\nu, a)) = \sum_{\nu, a \in \{0, 1\}} \omega(\tau, (\nu, a)) = P((\nu_k, a_k) | \tau), \quad \text{if } \tau < N - 1, \]
\[ \omega(\tau, (\nu, a) = (1, 1)) = 1 \quad \text{if } \tau \geq N - 1. \quad \text{(B.10)} \]

From (B.10) and (B.7), if $\tau \geq N - 1$, then we have
\[ \omega(\tau, (\nu, a)) = 0 \quad \text{if } (\nu, a) \neq (1, 1), \]
\[ \sum_{\nu, a \in \{0, 1\}} \omega(\tau + 1, (\nu, a)) \leq \eta \lambda_a \sum_{\nu, a \in \{0, 1\}} \omega(\tau, (\nu, a)). \quad \text{(B.11)} \]

Similar to (B.9), $J_{\pi_N^{\infty}}^{\eta, \infty}(\pi_N^{\infty}, \pi_N^{\infty})$ can be expressed in terms of $\omega$ as
\[ J_{\pi_N^{\infty}}^{\eta, \infty}(\pi_N^{\infty}, \pi_N^{\infty}) = \sum_{\tau \geq 0} \sum_{\nu, a \in \{0, 1\}} \omega(\tau, (\nu, a)) f^\tau(\bar{P}). \quad \text{(B.12)} \]

From (B.6) and (B.10), we have
\[ \omega(\bar{\tau}, (\nu, a)) = \omega(\bar{\tau}, (\nu, a)), \quad \bar{\tau} \in [0, N - 1), \]
\[ \sum_{i=N-1}^{\infty} \omega(i, (\nu, a)) = \omega(\bar{\tau}, (\nu, a)), \quad \bar{\tau} = N - 1. \quad \text{(B.13)} \]

The condition in (31) ensures $\sum_{i=0}^{\infty} \omega(i, (\nu, a)) f^i(\bar{P})$ converges exponentially to 0 with $i$. This guarantees the existence of $\epsilon_N$ such that $\lim_{N \to \infty} \epsilon_N = 0$ and
\[ \left| \sum_{i=N-1}^{\infty} \sum_{\nu, a} \omega(i, (\nu, a)) f^i(\bar{P}) - \omega(\bar{N} - 1, (\nu, a)) f^{N-1}(\bar{P}) \right| \leq \sum_{i=N-1}^{\infty} \sum_{\nu, a} \omega(i, (\nu, a)) \left( f^i(\bar{P}) - f^{N-1}(\bar{P}) \right) \]
\[ \leq \sum_{i=N-1}^{\infty} \sum_{\nu, a} \omega(i, (\nu, a)) f^i(\bar{P}) \leq \epsilon_N. \quad \text{(B.14)} \]

From (B.9) and (B.12)- (B.14), we can prove that
\[ |J_{\pi_N^{\infty}}^{\eta, N}(\pi_N^{\infty}, \pi_N^{\infty})| - |J_{\pi_N^{\infty}}^{\eta, \infty}(\pi_N^{\infty}, \pi_N^{\infty})| \leq \epsilon_N. \quad \text{(B.15)} \]

According to the optimality of $\pi_N^{\infty}$, we have
\[ J_{\pi_N^{\infty}}^{\eta, \infty}(\pi_N^{\infty}, \pi_N^{\infty}) \leq J_{\pi_N^{\infty}}^{\eta, \infty}(\pi_N^{\infty}, \pi_N^{\infty}), \quad \text{(B.16)} \]

(B.15) and (B.16) together give that
\[ J_{\pi_N^{\infty}}^{\eta, N}(\pi_N^{\infty}, \pi_N^{\infty}) \leq J_{\pi_N^{\infty}}^{\eta, N}(\pi_N^{\infty}, \pi_N^{\infty}) + \epsilon_N. \quad \text{(B.17)} \]

In a similar way, using the boundedness of $J_{\pi_N^{\infty}}^{\eta, \infty}(\pi_N^{\infty}, \pi_N^{\infty})$ and the optimality of $\pi_N^{\infty}$ to the truncated optimization problem, we can prove the existence of $\epsilon_N$ such that $\lim_{N \to \infty} \epsilon_N = 0$ and
\[ J_{\pi_N^{\infty}}^{\eta, N}(\pi_N^{\infty}, \pi_N^{\infty}) \leq J_{\pi_N^{\infty}}^{\eta, \infty}(\pi_N^{\infty}, \pi_N^{\infty}) + \epsilon_N. \quad \text{(B.18)} \]

Eq. (B.5), (B.17), and (B.18) prove (B.3) since any $\epsilon_N, \epsilon_N$, satisfying that $2\epsilon_N + 2\epsilon_N \leq \epsilon_N$, $\epsilon_N$, and $\lim_{N \to \infty} \epsilon_N = 0$ can make (B.3) hold. (B.3) and (B.4) give (38). Eq. (39) can be proved similarly.

References

[1] Eitan Altman. Constrained Markov decision processes, volume 7. CRC Press, 1999.
[2] Brian DO Anderson and John B Moore. Optimal filtering. Courier Corporation, 2012.
[3] Duarte Antunes, WPMH Heemels, and Paulo Tabuada. Dynamic programming formulation of periodic event-triggered control: Performance guarantees and co-design. In 2019 IEEE 51st IEEE conference on decision and control (CDC), pages 7212–7217. IEEE, 2012.
[4] Karl J Aström. Introduction to stochastic control theory. Courier Corporation, 2012.
[5] Michelle S Chong, Henrik Sandberg, and André MH Teixeira. A tutorial introduction to security and privacy for cyber-physical systems. In 2019 18th European Control Conference (ECC), pages 968–978. IEEE, 2019.
[6] C. Cooper and N. Hahi. An optimal stochastic control problem with observation cost. IEEE Transactions on Automatic Control, 16(2):185–189, 1971.
[7] Burak Demirel, Alex S Leong, Vijay Gupta, and Daniel E Quevedo. Tradeoffs in stochastic event-triggered control. IEEE Transactions on Automatic Control, 64(6):2567–2574, 2018.
[8] Tahmoores Farjam, Themistoklis Charalambous, and Henk Wymeersch. Timer-based distributed channel access for control over unknown unreliable time-varying communication channels. In 2019 18th European Control Conference (ECC), pages 2975–2982. IEEE, 2019.
[9] Vijay Gupta, Demetri Spanos, Babak Hassibi, and Richard M Murray. On lqg control across a stochastic packet-dropping link. In Proceedings of the 2005, American Control Conference, 2005., pages 360–365. IEEE, 2005.
[10] Onésimo Hernández-Lerma and Jean B Lasserre. Discrete-time Markov control processes: basic optimality criteria, volume 30. Springer Science & Business Media, 2012.

[11] Junling Hu and Michael P Wellman. Nash Q-learning for general-sum stochastic games. Journal of machine learning research, 4(Nov):1039–1069, 2003.

[12] Shizuo Kakutani. A generalization of Brouwer's fixed point theorem. Duke mathematical journal, 8(3):457–459, 1941.

[13] Kyoung-Dae Kim and Panganamala R Kumar. Cyber–physical systems: A perspective at the centennial. Proceedings of the IEEE, 100(Special Centennial Issue):1287–1308, 2012.

[14] Steffi Knorn and Subhrakanti Dey. Optimal energy allocation for linear control with packet loss under energy harvesting constraints. Automatica, 77:259–267, 2017.

[15] Markus Kögel, Daniel Quevedo, and Rolf Findeisen. Combined control and communication scheduling for constrained system using robust output feedback MPC. In 2019 18th European Control Conference (ECC), pages 1778–1783. IEEE, 2019.

[16] Alex S Leong, Subhrakanti Dey, and Daniel E Quevedo. Transmission scheduling for remote state estimation and control with an energy harvesting sensor. Automatica, 91:54–60, 2018.

[17] Alex S Leong, Daniel E Quevedo, Daniel Dolz, and Subhrakanti Dey. Transmission scheduling for remote state estimation over packet dropping links in the presence of an eavesdropper. IEEE Transactions on Automatic Control, 64(9):3732–3739, 2018.

[18] Alex S Leong, Daniel E Quevedo, Takashi Tanaka, Subhrakanti Dey, and Anders Ahlén. Event-based transmission scheduling and LQG control over a packet dropping link. IFAC-PapersOnLine, 50(1):8945–8950, 2017.

[19] Wenshuo Li, Zidong Wang, Qinyuan Liu, and Lei Guo. An information aware event-triggered scheme for particle filter based remote state estimation. Automatica, 103:151–158, 2019.

[20] Yuzhe Li, Daniel E Quevedo, Subhrakanti Dey, and Ling Shi. SINR-based DoS attack on remote state estimation: A game-theoretic approach. IEEE Transactions on Control of Network Systems, 4(3):632–642, 2017.

[21] Mohammad Hossein Manshaei, Quanyan Zhu, Tansu Alpcan, Tamer Başar, and Jean-Pierre Hubaux. Game theory meets network security and privacy. ACM Computing Surveys (CSUR), 45(3):1–39, 2013.

[22] Fei Miao, Quanyan Zhu, Miroslav Pajic, and George J Pappas. A hybrid stochastic game for secure control of cyber-physical systems. Automatica, 93:55–63, 2018.

[23] Yilin Mo and Bruno Sinopoli. Secure control against replay attacks. In 2009 47th annual Allerton conference on communication, control, and computing (Allerton), pages 911–918. IEEE, 2009.

[24] Edwin GW Peters, Daniel E Quevedo, and Minyue Fu. Controller and scheduler codesign for feedback control over IEEE 802.15. 4 networks. IEEE Transactions on Control Systems Technology, 24(6), 2016.

[25] Luca Schenato, Bruno Sinopoli, Massimo Franceschetti, Kameshwar Poolla, and S Shankar Sastry. Foundations of control and estimation over lossy networks. Proceedings of the IEEE, 95(1):163–187, 2007.

[26] Greg Welch, Gary Bishop, et al. An introduction to the Kalman filter. 1995.

[27] Stefan Wildhagen and Frank Allgöwer. Scheduling and control over networks using MPC with time-varying terminal ingredients. In 2020 American Control Conference (ACC), pages 1913–1918. IEEE, 2020.

[28] Junfeng Wu, Xiaokang Ren, Duo Han, Dawei Shi, and Ling Shi. Finite-horizon gaussianity-preserving event-based sensor scheduling in Kalman filter applications. Automatica, 72:100–107, 2016.

[29] Heng Zhang, Peng Cheng, Ling Shi, and Jiming Chen. Optimal DoS attack scheduling in wireless networked control system. IEEE Transactions on Control Systems Technology, 24(3):843–852, 2015.

[30] Junhui Zhang, Jitao Sun, and Chengcui Zhang. Stochastic game in linear quadratic Gaussian control for wireless networked control systems under DoS attacks. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2020.

[31] Quanyan Zhu and Tamer Basar. Game-theoretic methods for robustness, security, and resilience of cyberphysical control systems: games-in-games principle for optimal cross-layer resilient control systems. IEEE Control Systems Magazine, 35(1):46–65, 2015.