Contents lists available at ScienceDirect

Automatica

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

Brief paper

Rollout approach to sensor scheduling for remote state estimation under integrity attack[☆]

Hanxiao Liu [a,b,c], Yuchao Li [c], Karl Henrik Johansson [c], Jonas Mårtensson [c], Lihua Xie [b,*]

[a] *School of Artificial Intelligence, Shanghai University, Shanghai, China*
[b] *School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore*
[c] *School of Electrical Engineering and Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden*

ARTICLE INFO

Article history:
Received 10 August 2021
Received in revised form 10 February 2022
Accepted 10 May 2022
Available online 15 July 2022

ABSTRACT

We consider the sensor scheduling problem for remote state estimation under integrity attacks. We seek to optimize a trade-off between the energy consumption of communications and the state estimation error covariance when the acknowledgment (ACK) information, sent by the remote estimator to the local sensor, is compromised. The sensor scheduling problem is formulated as an infinite horizon discounted optimal control problem with infinite states. We first analyze the underlying Markov decision process (MDP) and show that the optimal scheduling without ACK attack is of the threshold type. Thus, we can simplify the problem by replacing the original state space with a finite state space. For the simplified MDP, when the ACK is under attack, the problem is modeled as a partially observable Markov decision process (POMDP). We analyze the induced MDP that uses a belief vector as its state for the POMDP. We investigate the properties of the exact optimal solution via contractive models and show that the threshold type of solution for the POMDP cannot be readily obtained. A suboptimal solution is then obtained via a rollout approach, which is a prominent class of reinforcement learning (RL) methods based on approximation in value space. We present two variants of rollout and provide performance bounds of those variants. Finally, numerical examples are used to demonstrate the effectiveness of the proposed rollout methods by comparing them with a finite history window approach that is widely used in RL for POMDP.

1. Introduction

Cyber–physical systems (CPS) refer to physical and engineering systems whose operations are monitored, coordinated, controlled, and integrated by a computing and communication core closely (Baheti & Gill, 2011; Rajkumar, Lee, Sha, & Stankovic, 2010). Examples of CPS can be found in a variety of sectors, and many of them are vital to the normal operation of society. However, there have been many security incidents recently, such as Maroochy water breach in 2000, Stuxnet malware in 2010, and Venezuela blackouts in 2019. Those incidents have motivated the recurring study of the security of CPS.

This work considers the sensor scheduling problem for remote state estimation under cyber attack. There have been many existing works on sensor scheduling under attack on the forward channel, i.e. from a sensor to a remote estimator. From the perspective of attackers, an optimal denial-of-service (DoS) attack scheduling problem with energy constraint was studied in Zhang, Cheng, Shi, and Chen (2015). In Qin, Li, Shi, and Yu (2017), the authors considered how to maximize the system performance loss with a DoS attack on remote state estimation over packet-dropping networks. Other related works can be found in Zhang, Qi, Wu, Fu, and He (2016) and Qin et al. (2020).

There are also many works on attacks over the feedback channel. Most works are based on the ACK protocol, where the fusion center sends an acknowledgment whenever it receives a packet, proposed in Mo, Sinopoli, Shi, and Garone (2012). The authors studied sensor scheduling for both no-ACK and ACK protocols and analyzed properties of the optimal scheduling. The ACK-based sensor scheduling was proved to outperform the one without ACK, i.e., offline scheduling, under the same energy constraint studied in Han, Cheng, Chen, and Shi (2013). Based on the ACK

[☆] This work is supported by the A*STAR Industrial Internet of Things Research Program, Singapore, under the RIE2020 IAF-PP Grant A1788a0023, the Knut and Alice Wallenberg Foundation, Sweden, the Swedish Foundation for Strategic Research, and the Swedish Research Council. This work is mainly completed at Nanyang Technological University. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Yilin Mo under the direction of Editor Christos G. Cassandras.

[*] Corresponding author.
E-mail addresses: hanxiaoliu@shu.edu.cn (H. Liu), yuchao@kth.se (Y. Li), kallej@kth.se (K.H. Johansson), jonas1@kth.se (J. Mårtensson), elhxie@ntu.edu.sg (L. Xie).

https://doi.org/10.1016/j.automatica.2022.110473
0005-1098/© 2022 Elsevier Ltd. All rights reserved.

feedback scheme, many works consider the effect of ACK-based attacks. Guo, Wang, and Shi (2017) studied the DoS attack on the feedback channel in the ACK-based sensor scheduling, and proved that the optimal policy has a threshold structure. From the perspective of defenders, an ACK-based deception scheme for sensors was first proposed in Ding, Ren, and Shi (2016). Furthermore, the authors presented an equivalent belief-based stochastic game to obtain the optimal stationary strategy for each agent in Ding, Ren, Quevedo, Dey, and Shi (2020). The above works focus on the attack scheduling problem or active deception-based schemes. To the best of our knowledge, few works focus on obtaining an optimal scheduling when the ACK received by the sensor is attacked. Note that the uncertainty induced by the attack poses a major challenge in this problem.

For most existing works without ACK-based attacks, the sensor scheduling problem is formulated as an optimal control problem with system dynamics model given by MDP (Leong, Ramaswamy, Quevedo, Karl, & Shi, 2020; Wu, Ren, Jia, Johansson, & Shi, 2019). When an ACK-based attack is present, it is most conveniently formulated by POMDP whose structural results could be obtained via some stochastic ordering (Krishnamurthy, 2016). In this paper, we investigate the possibility to derive a structural result via Monotone likelihood Ratio (MLR) ordering. It is proved that a threshold type of solution for POMDP cannot be readily available. A suboptimal solution approach via rollout is then proposed. "Rollout" was first proposed in Tesauro and Galperin (1997). It can be considered as single policy iteration (Bertsekas, 2019) and provides an online approach for solving stochastic scheduling, combinatorial optimization, and sequential repairing problems (Bertsekas & Castanon, 1999; Bertsekas, Tsitsiklis, & Wu, 1997; Bhattacharya, Badyal, Wheeler, Gil, & Bertsekas, 2020).

In this work, we focus on the sensor scheduling problem for remote state estimation under the presence of an ACK-based attack. We aim to obtain scheduling rules that minimize the expectation of an infinite horizon discounted accumulated cost. The contributions of our work are: (1) We prove that the optimal policy of the underlying MDP is of threshold type. Therefore, we can simplify the MDP by truncating the state space. (2) We present some properties of the exact optimal solution through contractive models for the MDP with belief vectors as states, and also prove that the structural result cannot be established through MLR ordering. (3) We present a suboptimal scheme via approximation in value space and implement it through rollout with fixed and geometrically distributed truncated steps. The corresponding performance guarantee is provided. A theoretical investigation closely related to contributions (2) and (3) is given in Patek (2007). However, due to the simplification achieved in (1), we are able to use contractive models, which are simpler and better suited for our problem.

The rest of the paper is organized as follows. Section 2 introduces the system model, smart sensor, remote state estimation as well as ACK feedback scheme. The sensor scheduling problem under an ACK-based attack is formulated as an infinite horizon discounted problem. The underlying MDP is studied and the optimal policy is shown to be of threshold type in Section 3. In the presence of attack, a decision making process of the sensor is modeled as a POMDP, which is analyzed in Section 4. In Section 5, some numerical examples are provided to demonstrate the effectiveness of the proposed strategy. Conclusions are provided in Section 6.

Notations: The notation $h^n(x)$ stands for the function composition $h(h^{n-1}(x))$, where $n \in \mathbb{N}$ and $h^0(x) = x$. The operator $\rho(\cdot) : \mathbb{R}^{n\times n} \to \mathbb{R}_+$ denotes the spectral radius, i.e., $\rho(A) = \max\{|\lambda_1|, |\lambda_2|, \ldots, |\lambda_n|\}$, where $\lambda_1, \ldots, \lambda_n$ are the eigenvalues of a matrix $A \in \mathbb{R}^{n\times n}$. The identity matrix of size n is denoted by I_n. The transpose of a matrix A is represented by A^{T}. The notation $\mathbb{Z}$ denotes the set of integers. The cardinality of a finite set $\mathcal{S}$ is represented by $|\mathcal{S}|$. $\mathbb{S}_+^n$ ($\mathbb{S}_{++}^n$) is the set of $n \times n$ positive semi-definite (definite) matrices. When $X \in \mathbb{S}_+^n$ ($\mathbb{S}_{++}^n$), we simply write $X \succeq 0$ ($X \succ 0$).

2. Problem formulation

2.1. System model

Let us consider a linear time-invariant (LTI) system described by the following equations:

$$x_{k+1} = Ax_k + w_k, \tag{1}$$

$$y_k = Cx_k + \varphi_k, \tag{2}$$

where $x_k \in \mathbb{R}^n$ and $y_k \in \mathbb{R}^m$ are the vectors of state variables and sensor measurements at time k, respectively. $w_k \in \mathbb{R}^n$ denotes the process noise and $\varphi_k \in \mathbb{R}^m$ is the measurement noise. $w_k \in \mathcal{N}(0, Q)$ and $\varphi_k \in \mathcal{N}(0, R)$, where $Q \succeq 0$ and $R \succ 0$, respectively. It is assumed that $w_0, w_1, \ldots$ and $\varphi_0, \varphi_1, \ldots$ are mutually independent.

Assumption 2.1. The system matrix A is of full rank, i.e., rank(A) $= n$.

Assumption 2.2. The pair (A, C) is observable and $(A, \sqrt{Q})$ is controllable.

Remark 2.1. Assumption 2.1 holds for all discrete systems discretized from their continuous counterparts.

2.2. Smart sensor

We consider the so-called "smart sensor" as described in Lewis et al. (2004), which first locally estimates the state x_k based on all the measurements it has collected up to time k and then transmits its local estimate to the remote estimator.

The sensor's local minimum mean squared error (MMSE) estimate of the state x_k and the corresponding error covariance are defined as $\hat{x}_k^s = \mathbb{E}[x_k \mid y_1, y_2, \ldots, y_k]$ and $\hat{P}_k^s = \mathbb{E}\left[(x_k - \hat{x}_k^s)(x_k - \hat{x}_k^s)^{\mathrm{T}} \mid y_1, y_2, \ldots, y_k\right]$, respectively. They can be calculated by a standard Kalman filter. Under Assumption 2.2, the estimation error covariance of the Kalman filter converges to a unique value. Without loss of generality, we assume that the Kalman filter at the sensor side has entered the steady state. Therefore, we simplify our subsequent discussion by setting $\hat{P}_k^s = \bar{P}$, $k \geq 1$, where $\bar{P}$ is the steady-state error covariance. For notational ease, we define the Lyapunov operator $h: \mathbb{S}_+^n \to \mathbb{S}_+^n$ as $h(X) \triangleq AXA^{\mathrm{T}} + Q$. Then $\bar{P}$ is given by the unique positive definite solution of the equation $X = h(X) - h(X)C^{\mathrm{T}}[Ch(X)C^{\mathrm{T}} + R]^{-1}Ch(X)$.

After obtaining $\hat{x}_k^s$, the sensor will transmit it as a data packet to the remote estimator. Random data drops may occur because of the existence of fading and interference. We assume that the sensor has two choices: one is to send the local state estimate with high power, which will consume energy e_h ($e_h > 0$); the other is to send the estimate with low power, which will consume energy e_l ($0 < e_l < e_h$). We assume that for the first choice, the packet will always arrive at the remote estimator, while for the other one, the arrival rate is $\upsilon \in (0, 1)$. In Fig. 1, we use two lines to denote these two choices. The upper line refers to the choice e_h, and the lower line represents the choice e_l.

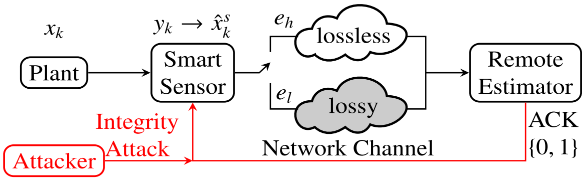


Fig. 1. The system diagram.

2.3. Remote state estimation

The transmission of $\hat{x}_k^s$ between the sensor and the remote estimator can be characterized by a binary random variable γ_k, $k \in \mathbb{N}$:

$$\gamma_k = \begin{cases} 1, & \text{if } \hat{x}_k^s \text{ arrives at time } k, \\ 0, & \text{otherwise (regarded as dropout).} \end{cases}$$

Denote as $\hat{x}_k$ and $\hat{P}_k$ the remote estimator's own MMSE state estimate and the corresponding error covariance based on all the sensor data packets received up to time step k. The remote state estimate $\hat{x}_k$ follows the recursion:

$$\hat{x}_k = \begin{cases} \hat{x}_k^s, & \text{if } \gamma_k = 1, \\ A\hat{x}_{k-1}, & \text{if } \gamma_k = 0. \end{cases} \tag{3}$$

The corresponding state estimation error covariance $\hat{P}_k$ satisfies:

$$\hat{P}_k = \begin{cases} \bar{P}, & \text{if } \gamma_k = 1, \\ h(\hat{P}_{k-1}), & \text{if } \gamma_k = 0. \end{cases} \tag{4}$$

The objective of sensor scheduling for the smart sensor is to optimize a trade-off between the energy consumption of communications and the trace of the estimation error covariance of the remote estimator. We formally formulate the cost that the sensor aims to minimize as $J_{\text{obj}} = \lim_{N\to\infty} \mathbb{E}\left\{\sum_{k=0}^{N} \alpha^k g_k\right\}$, where $\alpha \in (0,1)$ is the discount factor and g_k denotes the stage cost given by $g_k = \beta \cdot (\text{energy consumption}) + (1-\beta) \cdot (\text{trace of error covariance})$, while $\beta \in (0,1)$ is the weighting coefficient. The term "energy consumption" is related to e_l and e_h, and "error covariance" is related to Eq. (4). Note that both stage cost and control selection depend on the holding time, which is the time since the last successful reception of the data. Therefore, it is important for the sensor to know the holding time in order to make right control selection. For this reason, the following acknowledgment scheme is employed.

2.4. ACK feedback scheme

To improve the estimation performance under an energy constraint, an online power scheduling approach based on the ACK from the remote estimator was proposed in Li, Quevedo, Lau, and Shi (2013). We consider the same setting and use a scheme akin to Li et al. (2013). The remote estimator generates a 1-bit ACK signal to indicate whether the data packet is delivered successfully or not, which is illustrated in Fig. 1. We set $\gamma_k = 1$ when the information $\hat{x}_k^s$ has been delivered to the remote state estimator, and $\gamma_k = 0$ otherwise. In this way, the sensor can obtain the real-time information from the remote estimator. However, ACK scheme faces the risk of being attacked. The next section shows a possible ACK-based attack model, which results in degradation of the performance of sensor scheduling.

2.5. Attack model

While the ACK feedback scheme is simple and easy to implement, the simple structure makes it a likely target of an attacker who can carry out integrity attacks or DoS attacks (Li, Quevedo, Dey, & Shi, 2015). In this section, we propose a possible attack strategy for the attacker and investigate the corresponding mitigation of this kind of attack in the rest of the paper.

When the ACK channel is perfect, the smart sensor will receive a real-time ACK. If there is an integrity attack launched by "Attacker" shown in Fig. 1, the ACK may be modified, i.e., $\gamma_k = 0$ may be modified to 1 and $\gamma_k = 1$ to 0, according to certain probability defined by the attacker. In order to differentiate the ACK signal under the normal and attack scenarios, we use z_k to denote the ACK received by the smart sensor under a possible attack for later discussion. Note that z_k may not be equal to γ_k. We denote as κ_0 and κ_1 the probability that $\gamma_k = 0$ and $\gamma_k = 1$ are flipped by the attacker, respectively, i.e., $\kappa_0 = p(z_k = 1 \mid \gamma_k = 0)$, $\kappa_1 = p(z_k = 0 \mid \gamma_k = 1)$, where $p(\cdot)$ denotes the probability. It is assumed that the attack probabilities do not vary during the attack process. Our goal is to design a power scheduling approach, knowing that the ACK information is under the above flip attack.

Remark 2.2. If the smart sensor is only concerned with the error covariance of remote state estimation, namely, $\beta = 0$, the optimal attack strategy for an attacker is given as $(\kappa_0, \kappa_1) = (1, 0)$. On the other hand, if the energy consumption of smart sensor is the only concern ($\beta = 1$), corresponding attack probabilities should be given as $(\kappa_0, \kappa_1) = (0, 1)$. Since the smart sensor aims to optimize a trade-off between the remote state estimation accuracy and transmission energy cost, intuitively, the optimal attack probability should be between 0 and 1. This can be also observed in the example in Section 5.1.

Remark 2.3. Note that the flipping probability of the attack may be obtained by the smart sensor which transmits its state estimate with high power e_h over a period of time and computes the statistics of the ACK signals. In view of this, we assume that the system has the knowledge of attack probabilities and focus on the mitigation of this kind of attack.

Remark 2.4. The related study on the fake acknowledgment attack can be found in Li et al. (2015). The optimal DoS attack on the feedback channel against the ACK-based sensor power scheduling is studied in Guo et al. (2017). In Li, Quevedo, Dey, and Shi (2016), a game-theoretic approach to acknowledgment attack is proposed and the Nash equilibrium is studied. Note that the above works consider the DoS attack and emphasize the performance analysis of attacks on the feedback channel. Our work focuses on mitigating the effect of the attack on ACK.

Remark 2.5. It is worth noting that our proposed attack model is different from the lossy acknowledgment channel model. The key difference is that the remote estimator sends a feedback at each time and the attacker modifies the feedback signal with certain probabilities.

2.6. Preliminary analysis

In this section, we define the state, control, and observation of our problem. We apply the POMDP framework to model the process given that ACK is under a possible attack. Note that due to the recursion of the dynamics in Eq. (4), the covariance P_k can only take value in the infinitely countable set $\{\bar{P}, h(\bar{P}), h^2(\bar{P}), \ldots\}$. Denote as $s_k \in \mathbb{Z}$ the holding time since the most recent successful reception of the data from the sensor by the remote estimator:

$$s_k \triangleq k - 1 - \max_{1\le t\le k-1} \{t : \gamma_t = 1\}. \tag{5}$$

Therefore, $s_k = 0$ means that the message sent at time $k-1$ has been successfully received. We denote by $\mathcal{S}$ the state space of the MDP. The state space has countable elements $0, 1, \ldots$, standing for the holding time s_k, and we will use s_k to represent the unspecified state in $\mathcal{S}$. Denote as u_k the control option and as $\mathcal{U}$ the control space of the problem. The control options are to send the state estimate with high or low energy, denoted as 1 and 0, respectively, i.e., $\mathcal{U} = \{0, 1\}$, and we will use u_k to represent the unspecified control at time k. In particular, we use s and u to refer to a value in $\mathcal{S}$ and $\mathcal{U}$. We denote by $p_{ss'}(u)$ the transition probability from s to s' under control u and denote by $g(s, u, s')$ the stage cost when such transition occurs. It is clear that the probabilities are given as

$$p_{ss'}(1) = \begin{cases} 1, & s' = 0, \\ 0, & \text{o.w.}, \end{cases} \quad p_{ss'}(0) = \begin{cases} \upsilon, & s' = 0, \\ 1-\upsilon, & s' = s+1, \\ 0, & \text{o.w.}. \end{cases} \tag{6}$$

For the power scheduling problem of our interest, the following functions are usually used as cost per stage

$$\begin{aligned} g(s, 1, s') &= \beta e_h + (1-\beta)\operatorname{tr}(\bar{P}), \\ g(s, 0, s') &= \beta e_l + (1-\beta)\operatorname{tr}(h^{s'}(\bar{P})). \end{aligned} \tag{7}$$

Due to the presence of attack, the true state s_k cannot be computed according to Eq. (5) since γ_k is potentially compromised. Therefore, the observation received at current time is conditioned on the current state s, denoted by $q_z(s)$, and its conditional probability is given as

$$q_1(s) = \begin{cases} 1-\kappa_1, & s = 0, \\ \kappa_0, & \text{o.w.}, \end{cases} \quad q_0(s) = \begin{cases} \kappa_1, & s = 0, \\ 1-\kappa_0, & \text{o.w.}. \end{cases} \tag{8}$$

2.7. Problem of interest

Given that we know the successful arriving rate υ when transmitted with low energy, the presence of the attacker, the observation probabilities $q_z(s)$, and the observations z_k, we are interested in obtaining a scheduling rule that minimizes the following expectation of the infinite horizon accumulated cost of $g(s_k, u_k, s_{k+1})$ given by Eq. (7) with a given discount factor of $\alpha \in (0, 1)$: $J(s) = \lim_{N\to\infty} \mathbb{E}\left\{\sum_{k=0}^{N} \alpha^k g(s_k, u_k, s_{k+1}) \mid s_0 = s\right\}$.

The challenge of the problem comes from two folds. First, the problem is a POMDP, which is difficult in its own right. Second, the underlying MDP has a state space that is composed of infinite states. In what follows, we will first investigate the properties of the MDP. It is shown that the optimal policy, if the states are known, is of the threshold type. Therefore, it is then sufficient to consider the finitely many state case, thus obtaining a MDP with a truncated state space. Then we will show for the POMDP induced by the truncated state space, contractive models apply and some performance guarantees can be established.

3. Simplification of the underlying MDP

Table 1 summarizes some parameters related to the analysis of the underlying MDP.

3.1. Analysis on the properties of MDP

Recall that the Lyapunov operator is defined as $h(X) \triangleq AXA^{\mathrm{T}} + Q$. For the error covariance iterated through the Lyapunov operator, we have the following lemma.[1]

Lemma 3.1. *Under Assumptions 2.1 and 2.2, the inequality*

$$\operatorname{tr}\left(h^k(\bar{P})\right) < \operatorname{tr}\left(h^{k+1}(\bar{P})\right), \quad \forall k \in \mathbb{N},$$

holds, where $\operatorname{tr}\left(h^0(\bar{P})\right) = \operatorname{tr}\left(\bar{P}\right)$.

From Lemma 3.1, we know that the sequence $\{\operatorname{tr}(h^k(\bar{P}))\}_{k=0}^{\infty}$ is monotonically increasing as k grows.

Lemma 3.2. *If* $\rho(A) \geq 1$, $\lim_{k\to\infty} \operatorname{tr}(h^k(\bar{P})) = +\infty$. *Otherwise, the sequence* $\{\operatorname{tr}(h^k(\bar{P}))\}_{k=0}^{\infty}$ *is bounded.*

The above lemma can be derived by Gelfand's formula (Lax, 2014). Recall the definition of the cost g given by Eq. (7), we have $g \geq 0$ viz., the MDP we are investigating here is nonnegative. This is formalized as the following lemma.

Lemma 3.3. *The cost function g satisfies*

$g(s, u, s') \geq 0, \forall (s, u, s') \in \mathcal{S} \times \mathcal{U} \times \mathcal{S}$.

In this paper, we will focus on the case where the stage cost is unbounded, which is formalized in the following assumption.

Assumption 3.1. The system matrix A satisfies $\rho(A) \geq 1$.

Remark 3.1. When $\rho(A) < 1$, in view of Lemma 3.2, the stage cost is bounded for all states and controls, in which case a large portion of the following analysis still applies. However, it would be possible, depending on the tuning parameter β, to have low energy transmission always as the optimal control, which would not be practically interesting to investigate.

For the nonnegative MDP with state space $\mathcal{S}$ and control space $\mathcal{U}$, we denote as $\bar{\mu} : \mathcal{S} \to \mathcal{U}$ a mapping from state space to control space, and $\bar{\pi} = \{\bar{\mu}_0, \bar{\mu}_1, \ldots\}$ as a sequence of $\bar{\mu}_k$. Thus, the cost functions under $\bar{\pi}$ and $\{\bar{\mu}, \bar{\mu}, \ldots\}$ are defined as $J_{\bar{\pi}}(s) = \lim_{N\to\infty} \mathbb{E}\left\{\sum_{k=0}^{N} \alpha^k g(s_k, \bar{\mu}_k(s_k), s_{k+1}) \mid s_0 = s\right\}$, $J_{\bar{\mu}}(s) = \lim_{N\to\infty} \mathbb{E}\left\{\sum_{k=0}^{N} \alpha^k g(s_k, \bar{\mu}(s_k), s_{k+1}) \mid s_0 = s\right\}$, in view of g being nonnegative. For such problems, we are interested in obtaining the optimal cost function J^* and optimal stationary policy $\bar{\mu}^*$ given as

$$J^*(s) = \inf_{\bar{\pi}} J_{\bar{\pi}}(s), \; J_{\bar{\mu}^*}(s) = J^*(s), \; \forall s \in \mathcal{S}. \tag{9}$$

For the problem of interest, the optimal cost function can be achieved since the cost function is nonnegative (Lemma 3.3) and the control space is compact under an unbounded cost. This is formalized as the following lemma.

Lemma 3.4 (Chapter 6, Bertsekas, 1976)**.** *(a) The optimal cost function* J^* *of (9) satisfies Bellman's equation. Namely, for all* $s \in \mathcal{S}$, $J^*(s) = \min_{u\in\mathcal{U}} \sum_{s'\in\mathcal{S}} p_{ss'}(u)\left\{g(s, u, s') + \alpha J^*(s')\right\}$. *(b) Let* $\bar{\pi} = \{\bar{\mu}, \bar{\mu}, \ldots\}$ *be an admissible stationary policy. We have for all* $s \in \mathcal{S}$, $J_{\bar{\mu}}(s) = \sum_{s'\in\mathcal{S}} p_{ss'}\left(\bar{\mu}(s)\right)\left\{g\left(s, \bar{\mu}(s), s'\right) + \alpha J_{\bar{\mu}}(s')\right\}$. *(c) There is an optimal stationary policy* $\bar{\mu}^*$ *that satisfies (9). (d) Starting with* $J_0(s) \equiv 0$, *the value iteration (VI) sequence* $\{J_k\}_{k=0}^{\infty}$ *generated by* $J_{k+1}(s) = \min_{u\in\mathcal{U}} \sum_{s'\in\mathcal{S}} p_{ss'}(u)\left\{g(s, u, s') + \alpha J_k(s')\right\}$ *for all* $s \in \mathcal{S}$, *converges pointwise to* J^*.

Theorem 3.1. *Let Assumptions 2.2 and 3.1 hold. The optimal policy* $\bar{\mu}^*$ *defined by Eq. (9) is of the threshold type, viz., there exists a constant* $\epsilon^* \in \mathcal{S}$ *such that*

$$\bar{\mu}^*(s) = \begin{cases} 0, & \text{if } s < \epsilon^*, \\ 1, & \text{if } s \geq \epsilon^*. \end{cases} \tag{10}$$

Moreover, the optimal cost is strictly increasing for all $s \leq \epsilon^*$, *viz.,* $J^*(s-1) < J^*(s)$ *when* $s \leq \epsilon^*$, *and remains constant for* $s \geq \epsilon^*$.

[1] We corrected an error in Lemma 2.3 of Shi, Johansson, and Qiu (2011) by imposing lemma under Assumptions 2.1 and 2.2.

Table 1
Descriptions of parameters.

Parameter	Description
$\alpha \in (0, 1)$	The discount factor for the accumulated cost.
$\beta \in (0, 1)$	The weighting coefficient related to energy consumption and error covariance.
$\upsilon \in (0, 1)$	The arrival rate if the sensor sends $\hat{x}_k^s$ with low power e_l.
$\ell \in \mathbb{Z}$, $\ell > \epsilon^*$	The largest value for the truncated space as defined in Section 3.2.
$\kappa_0, \kappa_1 \in (0, 1)$	The probabilities that $\gamma_k = 0$ and $\gamma_k = 1$ are flipped by the attacker.
$\gamma_k \in \{0, 1\}$	The ACK signal that indicates if $\hat{x}_k^s$ arrives at time k.
$z_k \in \{0, 1\}$	The ACK received by the smart sensor under a possible attack.
$\bar{\mu}^*(s) \in \{0, 1\}$	The optimal policy at state s.

Proof. For the simplicity of notations, we define $g_h \triangleq g(s, 1, 0) = \beta e_h + (1-\beta)\operatorname{tr}(\bar{P})$. By Lemma 3.4(b), (c), and (d), we can prove that $J^*(s) = \lim_{k\to\infty} J_k(s) \le \lim_{k\to\infty} J_k(s') = J^*(s')$. Namely, $J^*(s)$ is monotonically increasing. In addition, the term $g(s, 0, s+1)$ grows unbounded with s due to $\rho(A) \ge 1$. Thus, there exists some $\bar{s}$ such that $\bar{\mu}^*(\bar{s}) = 1$ and $J^*(\bar{s}) = g_h + \alpha J^*(0)$ and $J^*(s') \ge J^*(\bar{s}) = g_h + \alpha J^*(0), \quad \forall s' > \bar{s}$. On the other hand, as discussed above, for all s, we have $J^*(s) \le g_h + \alpha J^*(0)$. Thus, $J^*(s') = g_h + \alpha J^*(0)$. Define $\epsilon^* \triangleq \arg\min\{\bar{s} \mid \bar{\mu}^*(\bar{s}) = 1\}$. The above proof has shown that for all $s \ge \epsilon^*$, the optimal cost $J^*(s)$ is constant and equals $g_h + \alpha J^*(0)$. In view of definition of ϵ^*, we see that $J^*(\epsilon^* - 1) < J^*(\epsilon^*)$ [since otherwise, we would have $\bar{\mu}^*(\epsilon^* - 1) = 1$, contradicting the definition of ϵ^*]. For $s \le \epsilon^* - 2$, if $J^*(s) = J^*(s+1)$, we have $(1-\upsilon)[g(s, 0, s+1) + \alpha J^*(s+1)] = (1-\upsilon)[g(s+1, 0, s+2) + \alpha J^*(s+2)]$. However, this contradicts with $g(s, 0, s+1) < g(s+1, 0, s+2)$ and $J^*(s+1) \le J^*(s+2)$. Thus, the assumption is false and the proof is complete.

3.2. Computational approach

Due to the fact that the optimal policy is of the threshold nature, we can thus consider a truncated state space $\mathcal{S}_t = \{0, 1, \ldots, \ell\}$ with $\ell > \epsilon^*$. However, since ϵ^* is not known, neither is ℓ. For the truncated problem, with a slight abuse of notion, we will still use $p_{ss'}(u)$ to denote the transition probability, with the modification that $p_{\ell\ell}(0) = \upsilon$ and $p_{\ell 0}(0) = 1 - \upsilon$, while the stage costs remain the same as for the original problem. In particular, $g(\ell, u, \ell) = \beta e_l + (1-\beta)\operatorname{tr}\left(h^{\ell+1}(\bar{P})\right)$. When the truncated problem fulfills the condition that $\ell > \epsilon^*$, its optimal policy is the same as the one of original problem with infinite state space. The proof is omitted due to the limit of space.

Theorem 3.2. *For the truncated problem with state space $\mathcal{S}_t = \{0, 1, \ldots, \ell\}$, the optimal cost function and optimal policy satisfy that: (a) If $\ell \ge \epsilon^*$, the optimal control of the truncated problem is also given by Eq. (10), while the optimal cost function is $J^*_{|\mathcal{S}_t}$, the restriction of optimal cost of original MDP to $\mathcal{S}_t$. (b) If $\ell < \epsilon^*$, the optimal control of the truncated problem is 0 for all $s \in \mathcal{S}_t$ while the optimal cost is upper bounded by $J^*_{|\mathcal{S}_t}$.*

In view of Theorem 3.2 and Proposition 12 in Bertsekas (1976), given some ℓ, we can use VI to get the corresponding optimal policy for the truncated problem. If it appears to be the threshold type, it corresponds to Theorem 3.2(a) and the optimal cost and policy obtained for the truncated problem are exactly the same as the underlying MDP with the original state space. If it is not the threshold type, it corresponds to Theorem 3.2(b) and we need to increase ℓ to get the threshold ϵ^*. From now on, we refer to the underlying MDP as the one with the truncated state space provided that $\ell > \epsilon^*$, and with a slight abuse of notion, we will still use $p_{ss'}(u)$ to denote the transition probability, with the modification that $p_{\ell\ell}(0) = \upsilon$ and $p_{\ell 0}(0) = 1 - \upsilon$, while the transition costs for all $s \in \mathcal{S}_t$ remain unchanged as mentioned before Theorem 3.2. Also, we will still use J^* to denote the optimal cost function.

4. Scheduling under integrity attack

When the ACK information is under attack, we have a POMDP problem where the underlying state $\mathcal{S}$ has an infinite dimension, with a known probability of observation. Based on the study of the previous section, the original state space can be replaced by a truncated version $\mathcal{S}_t$ with no impact on control selection. Thus, the POMDP of our concern is the one with state space $\mathcal{S}_t$, control $\mathcal{U}$, observation $\mathcal{Z}$, transition probability given by Eq. (6), stage cost given by Eq. (7), observation probability given by Eq. (8), with the exception that $p_{\ell\ell}(0) = \upsilon$ and $p_{\ell 0}(0) = 1 - \upsilon$, while the transition costs for all $s \in \mathcal{S}_t$ remain unchanged.

4.1. Properties of the exact optimal solution

For the POMDP introduced above, we analyze the induced MDP that uses a belief vector as a state. It is well-known that those two problems are equivalent, albeit the problem with belief state is infinite-dimensional (Astrom, 1965). For this study, we apply the contractive model detailed in Bertsekas (2018). To this end, we consider as states the functions $b : \mathcal{S}_t \to \mathbb{R}$ such that $\sum_{s\in\mathcal{S}_t} b(s) = 1$, and $b(s) \ge 0$, $\forall s$. It is easy to see that b is a vector and $b \in \mathbb{R}^{\ell+1}$. We denote as $\mathcal{B}$ the set of all such belief states. Note that $\mathcal{B} \subset \mathbb{R}^{\ell+1}$. Denote as $V : \mathcal{B} \to \mathbb{R}$ a function defined on $\mathcal{B}$, and as $\mathcal{V}$ the set of functions that contains all V where $\|V\|_\infty < \infty$. Given a belief state b and a control u, the distribution of z can be computed as $\hat{p}(z \mid b, u) = \sum_{s=0}^{\ell} b(s) \sum_{s'=0}^{\ell} p_{ss'}(u) q_z(s')$. The dynamics of b is governed by a Bayesian estimator denoted as $\Phi : \mathcal{B} \times \mathcal{U} \times \mathcal{Z} \to \mathcal{B}$. As an illustration, if $\kappa_1 = \kappa_2 = \kappa$, then the dynamics of the estimator is simplified as

$$\Phi(b, 0, 1) = \begin{bmatrix} \dfrac{\upsilon(1-\kappa)}{\upsilon(1-\kappa) + (1-\upsilon)\kappa} \\ \dfrac{(1-\upsilon)\kappa b(0)}{\upsilon(1-\kappa) + (1-\upsilon)\kappa} \\ \vdots \\ \dfrac{(1-\upsilon)\kappa(b(\ell) + b(\ell-1))}{\upsilon(1-\kappa) + (1-\upsilon)\kappa} \end{bmatrix},$$

$$\Phi(b, 0, 0) = \begin{bmatrix} \dfrac{\upsilon\kappa}{\upsilon\kappa + (1-\upsilon)(1-\kappa)} \\ \dfrac{(1-\upsilon)(1-\kappa)b(0)}{\upsilon\kappa + (1-\upsilon)(1-\kappa)} \\ \vdots \\ \dfrac{(1-\upsilon)(1-\kappa)(b(\ell) + b(\ell-1))}{\upsilon\kappa + (1-\upsilon)(1-\kappa)} \end{bmatrix},$$

$$\Phi(b, 1, z) = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^{\mathrm{T}}, \ \forall z \in \mathcal{Z}.$$

For the induced infinite-dimensional MDP, the stage cost $\hat{g} : \mathcal{B} \times \mathcal{U} \to \mathbb{R}$ is given as $\hat{g}(b, u) = \sum_{s=0}^{\ell} b(s) \sum_{s'=0}^{\ell} p_{ss'}(u) g(s, u, s')$. Based on the above definitions, we introduce an abstract operator $H : \mathcal{B} \times \mathcal{U} \times \mathcal{V} \to \mathbb{R}$ that defines the induced MDP. It is given as

$$H(b, u, V) = \hat{g}(b, u) + \alpha \sum_{z\in\mathcal{Z}} \hat{p}(z \mid b, u) V\left(\Phi(b, u, z)\right). \quad (11)$$

We denote as μ a function $\mu : \mathcal{B} \to \mathcal{U}$ and all such functions form the set $\mathcal{M}$. In addition, we denote as π a sequence of admissible policies $\{\mu_k\}_{k=0}^{\infty}$ and Π as the set of all π. Then we introduce two Bellman operators $T_\mu, T : \mathcal{V} \to \mathcal{V}$ given by

$$(T_\mu V)(b) = H(b, \mu(b), V), \ (TV)(b) = \inf_{\mu \in \mathcal{M}} (T_\mu V)(b). \tag{12}$$

Theorem 4.1 (*Fixed Point Properties*)**.** *Let Assumptions 2.2 and 3.1 hold. For every T_μ and T, we have the following hold:*

(a) There exists unique V_μ, $V^* \in \mathcal{V}$ such that

$$V_\mu = T_\mu V_\mu, \ V^* = TV^*. \tag{13}$$

(b) V^* given in Eq. (13) is the optimal cost function, viz., $V^* = \inf_{\pi \in \Pi} V_\pi$, where V_π is given as $V_\pi = \lim_{N\to\infty} T_{\mu_0}\Big(T_{\mu_1}\big(\cdots (T_{\mu_N} V_0)\cdots\big)\Big)$, with $V_0 \equiv 0$. In particular, $V^* = \inf_{\mu\in\mathcal{M}} V_\mu$.

(c) For a policy $\mu \in \mathcal{M}$, $V_\mu = V^*$ if and only if $T_\mu V^* = TV^*$.

Proof. Given that the space $\mathcal{V}$ is complete, we only need to verify that the operators defined in Eq. (12) has the following properties: (1) For all V, $V' \in \mathcal{V}$ such that $V \le V'$, it holds that $T_\mu V \le T_\mu V'$, $\mu \in \mathcal{M}$, $TV \le TV'$. (2) For all V, $V' \in \mathcal{V}$, $\|T_\mu V - T_\mu V'\|_\infty \le \alpha \|V - V'\|_\infty$, $\forall \mu \in \mathcal{M}$. Then the results stated in (a), (b) and (c) follow from Prop. 2.1.1 and Prop. 2.1.2 in Bertsekas (2018).

(1) For all V, $V' \in \mathcal{V}$,

$$\begin{aligned}
&(T_\mu V)(b) = H(b, \mu(b), V)\\
=&\hat{g}(b, \mu(b)) + \alpha \sum_{z\in\mathcal{Z}} \hat{p}(z \mid b, \mu(b)) V\big(\Phi(b, \mu(b), z)\big)\\
\le&\hat{g}(b, \mu(b)) + \alpha \sum_{z\in\mathcal{Z}} \hat{p}(z \mid b, \mu(b)) V'\big(\Phi(b, \mu(b), z)\big)\\
=&H(b, \mu(b), V') = (T_\mu V')(b),\\
&(TV)(b) = \inf_{\mu\in\mathcal{M}} (T_\mu V)(b) \le \inf_{\mu\in\mathcal{M}} (T_\mu V')(b) = (TV')(b).
\end{aligned}$$

(2) For all $\mu \in \mathcal{M}$, we have

$$\begin{aligned}
&(T_\mu V)(b) - (T_\mu V')(b)\\
=&\alpha \sum_{z\in\mathcal{Z}} \hat{p}(z \mid b, u)\Big[V\big(\Phi(b, u, z)\big) - V'\big(\Phi(b, u, z)\big)\Big]\\
\le&\alpha \|V - V'\|_\infty,
\end{aligned}$$

which holds for all $b \in \mathcal{B}$. Reverse the order of V and V', which implies $|V(b) - V'(b)| \le \alpha\|V - V'\|_\infty$, $\forall b \in \mathcal{B}$. Take supremum over $\mathcal{B}$ and we get the desired result.

As is mentioned in Theorem 3.1, the optimal cost function is monotonically increasing as the holding time increases. However, for the induced MDP where now the state b is an element in the simplex of $\mathbb{R}^{\ell+1}$, additional conditions are required to obtain structural results like the one in Theorem 3.1. One simple and useful approach is through Monotone likelihood ratio (MLR) ordering (Krishnamurthy, 2016). For two belief states b and b', we call b dominates b' in the MLR sense if $b(s)b'(s') \le b'(s)b(s')$, $s < s'$, $s, s' \in \mathcal{S}_t$. Under suitable conditions, computing MLR ordering between two belief states b and b' is sufficient to conclude whether $V^*(b) \ge V^*(b')$. In particular, if b dominating b' in the MLR sense implies $V^*(b) \le V^*(b')$, then we say the POMDP is MLR decreasing in b. For the result to hold true, one key property required for the underlying MDP is that the related matrix is totally positive of order 2 (TP2). A stochastic matrix [including rectangle one, Krishnamurthy (2016, Definition 10.2.1)] is called TP2 if all its second-order minors are nonnegative. The relevant stochastic matrices of our concern under control u are the transition matrix $P(u)$ and observation matrix $M(u)$, given as

$$P(u) = \begin{bmatrix} p_{00}(u) & p_{01}(u) & \cdots & p_{0\ell}(u)\\ p_{10}(u) & p_{11}(u) & \cdots & p_{1\ell}(u)\\ \vdots & \vdots & \ddots & \vdots\\ p_{\ell 0}(u) & 0 & \cdots & p_{\ell\ell}(u) \end{bmatrix},$$

$$M(u) = \begin{bmatrix} q_0(0) & q_0(1) & \cdots & q_0(\ell)\\ q_1(0) & q_1(1) & \cdots & q_1(\ell) \end{bmatrix}^{\mathrm{T}}.$$

The following lemma outlines the sufficient conditions under which the POMDP is MLR decreasing in b.

Lemma 4.1 (*Theorem 11.2.1, Krishnamurthy, 2016*)**.** *For the POMDP with state space $\mathcal{S}_t$, control $\mathcal{U}$, observation $\mathcal{Z}$, and stage cost $g : \mathcal{S}_t \times \mathcal{U} \times \mathcal{S}_t \to \mathbb{R}$, if the following three conditions hold: (1) For each control $u \in \mathcal{U}$, the expected stage cost defined as $\sum_{s'\in\mathcal{S}_t} p_{ss'}(u) g(s, u, s')$ is decreasing in s; (2) The transition matrix $P(u)$ is TP2 for all u; (3) The observation matrix $M(u)$ is TP2 for all u, then the POMDP is MLR decreasing in b.*

Remark 4.1. The stage cost in Krishnamurthy (2016) is defined as a function of current state and current action $\mathcal{S}_t \times \mathcal{U}$, while here our stage cost $g(s, u, s')$ also depends on the next state. Thus, the stage cost in Krishnamurthy (2016) is the expected stage cost $\sum_{s'\in\mathcal{S}_t} p_{ss'}(u) g(s, u, s')$ in our paper.

In the following theorem, we will show that the problem of interest here does not fulfill the TP2 property, thus MLR decreasing in b cannot be established through the above conditions. For the following discussion, we introduce a new class of functions. We call $\sigma : \mathcal{S}_t \to \mathcal{S}_t$ a permutation if it is a bijection. For a given bijection σ, we obtain a new POMDP accordingly, with state space $\mathcal{S}_t$, control space $\mathcal{U}$, observation space $\mathcal{Z}$, transition probability given as $p^\sigma_{ss'}(u) = p_{\sigma^{-1}(s)\sigma^{-1}(s')}(u)$, transition cost given as $g^\sigma(s, u, s') = g\big(\sigma^{-1}(s), u, \sigma^{-1}(s')\big)$, observation probability $q^\sigma_z(s) = q_z\big(\sigma^{-1}(s)\big)$, transition matrix $P^\sigma(u)$ and observation matrix $M^\sigma(u)$ defined accordingly. Now we are ready to proceed to state the following result, which essentially means that for the problem of interest, the MLR decreasing relation cannot be established through Lemma 4.1.

Theorem 4.2. *Let Assumptions 2.2 and 3.1 hold. There exists no permutation σ such that the corresponding POMDP fulfills the conditions (1), (2), and (3) given in Lemma 4.1.*

Proof. In view of Lemma 3.1, stage cost $g(s, u, s')$ of the original POMDP is increasing in s. Thus, condition (1) in Lemma 4.1 is violated. To have condition (1) hold, the only valid permutation σ is given as $\sigma(s) = \ell - s$. Under σ, the stage cost $g^\sigma(s, u, s')$ is decreasing in s. However, we can see the transition matrix $P^\sigma(0)$ is not TP2. Indeed, the last second order minor of $P^\sigma(0)$ is given as,

$$\begin{aligned}
&\begin{vmatrix} p^\sigma_{(\ell-1)(\ell-1)}(0) & p^\sigma_{(\ell-1)(\ell)}(0)\\ p^\sigma_{(\ell)(\ell-1)}(0) & p^\sigma_{(\ell)(\ell)}(0) \end{vmatrix} = \begin{vmatrix} p_{11}(0) & p_{10}(0)\\ p_{01}(0) & p_{00}(0) \end{vmatrix}\\
=&\begin{vmatrix} 0 & \upsilon\\ 1-\upsilon & \upsilon \end{vmatrix} = \upsilon(\upsilon - 1) < 0,
\end{aligned}$$

which means $P^\sigma(0)$ is not TP2. Thus, there exists no permutation σ such that its corresponding POMDP fulfills the conditions (1), (2), and (3) in Lemma 4.1.

Theorem 4.2 shows that the structural result for the induced MDP is not readily available. Therefore, we seek a suboptimal solution based on approximation in value space, implemented through the use of rollout (Bertsekas, 2019).

4.2. Approximate solution through rollout

Given that the exact solution cannot be established through the MLR ordering for the belief states, we seek an approximate solution. Here, we use the rollout approach, which is a simulation-based approach. To this end, we fix certain base policy $\mu_b \in \mathcal{M}$. Given current belief state as b, we aim to obtain a control option as the following minimizer

$$\tilde{\mu}(b) \in \arg\min_{u \in \mathcal{U}} \hat{g}(b, u) + \alpha \sum_{z \in \mathcal{Z}} \hat{p}(z \mid b, u) \tilde{V}\big(\Phi(b, u, z)\big), \tag{14}$$

where we examine two options for $\tilde{V}$, which we denote as $\tilde{V}^r$ and $\tilde{V}^{(\lambda)}$. The first option $\tilde{V}^r$ is to evaluate base policy μ_b for some fixed $r \in \mathbb{N}$ steps, with final cost $\bar{V}(b) = \sum_{s \in \mathcal{S}_t} b(s) J^*(s)$, thus

$$\tilde{V}^r(b) = (T^r_{\mu_b} \bar{V})(b). \tag{15}$$

The second option $\tilde{V}^{(\lambda)}$ is to evaluate the base policy μ_b with geometrically distributed steps with final cost $\bar{V}$, thus $\tilde{V}$ is defined as

$$\tilde{V}^{(\lambda)}(b) = (T^{(\lambda)}_{\mu_b} \bar{V})(b),$$

where $\lambda \in (0, 1)$ is some design parameter, and $T^{(\lambda)}_{\mu_b} : \mathcal{V} \to \mathcal{V}$ is defined as

$$(T^{(\lambda)}_{\mu_b} V)(b) = (1 - \lambda) \sum_{\ell=1}^{\infty} \lambda^{\ell-1} \big(T^{\ell}_{\mu_b} V\big)(b). \tag{16}$$

Note that Eq. (16) is a well-defined, infinite-dimensional operator with the same fixed point as T_{μ_b}, and for all $V, V' \in \mathcal{V}$, it holds that $\|T^{(\lambda)}_{\mu_b} V - T^{(\lambda)}_{\mu_b} V'\|_\infty \le \alpha_\lambda \|V - V'\|_\infty$, where $\alpha_\lambda = \frac{\alpha(1-\lambda)}{1-\lambda\alpha}$. Refer to Li, Johansson, and Mårtensson (2019) for details of the above results.

When rollout method defined in Eq. (14) is implemented exactly, with $\tilde{V} = \tilde{V}^r$, the performance of $\tilde{\mu}$ with respect to the optimal cost V^* can be characterized by the following lemma, which is given as Proposition 2.2.1 (Bertsekas, 2018).

Lemma 4.2 (*Proposition 2.2.1, Bertsekas, 2018*). *Denote as $V_{\tilde{\mu}}$ the cost function of the policy $\tilde{\mu}$ that is given by Eq. (14), and $\tilde{V} = \tilde{V}^r$. Then the suboptimality of $V_{\tilde{\mu}}$ with respect to V^* is given by*

$$\|V_{\tilde{\mu}} - V^*\|_\infty \le \frac{2\alpha}{1-\alpha} \|T^r_{\mu_b} \bar{V} - V^*\|_\infty. \tag{17}$$

Compared with above result, when $\tilde{V} = \tilde{V}^{(\lambda)}$, we have the following performance bound.

Theorem 4.3 (*Performance Bound*). *Denote as $V_{\tilde{\mu}}$ the cost function of the policy $\tilde{\mu}$ that is given by Eq. (14), and $\tilde{V} = \tilde{V}^{(\lambda)}$. Then the suboptimality of $V_{\tilde{\mu}}$ with respect to V^* is given by*

$$\|V_{\tilde{\mu}} - V^*\|_\infty \le \frac{2\alpha}{1-\alpha} \|T^{(\lambda)}_{\mu_b} \bar{V} - V^*\|_\infty. \tag{18}$$

Proof. We view $T^{(\lambda)}_{\mu_b} V_0$ as the final cost and the above suboptimality is a direct application of Proposition 2.2.1, (Bertsekas, 2018).

Remark 4.2. The scalars in Eqs. (17) and (18) are usually unknown, so the resulting analysis will have a mostly qualitative character. However, Lemma 4.2 and Theorem 4.3 provide some insight on the performance of the rollout approach. By decreasing the discount factor α and increasing the number of lookahead step, a better performance bound could be expected.

4.3. Rollout implementation via Monte–Carlo sampling

Here we exemplify a rollout implementation via Monte-Carlo simulation method detailed in Bertsekas (2019). The algorithm is summarized in Algorithm 1, where $\tilde{V} = \tilde{V}^r$. Given a current belief state b, we run N_s Monte Carlo simulations to decide the control input, where the function "SIMULATOR" is used to take the Monte Carlo simulations with parameter action u, belief state b and truncated steps r. Then, a rollout control is obtained and can be applied in the system. It is worth noting that if we use the second variant where $\tilde{V} = \tilde{V}^{(\lambda)}$, the truncated step r is not anymore fixed, but rather a random variable drawn from geometric distribution with parameter λ.

Algorithm 1 Rollout policy for fixed truncated steps

Input: The discount factor $\alpha \in (0, 1)$, the sample number N_s, the arrival rate under lower energy υ, current belief state b, the truncated steps r, and the base policy $\mu_b \in \mathcal{M}$
Output: The rollout control.

```
function SIMULATOR($u, b, r$)
    $v \leftarrow 0$
    Apply $u$ and collect observation $z$
    $v \leftarrow \hat{g}(b, u)$, $b \leftarrow \Phi(b, u, z)$
    for $l = 1 \to r$ do
        $u \leftarrow \mu_b(b)$
        Apply $u$ and collect observation $z$
        $v \leftarrow v + \alpha^l \hat{g}(b, u)$, $b \leftarrow \Phi(b, u, z)$
    end for
    $v \leftarrow v + \alpha^{r+1} \bar{V}(b)$
    return $v$
end function
$v_0 \leftarrow 0$, $v_1 \leftarrow 0$
for $k = 1 \to N_s$ do
    $v_0 \leftarrow v_0 +$ SIMULATOR($0, b, r$)
    $v_1 \leftarrow v_1 +$ SIMULATOR($1, b, r$)
end for
$\tilde{u} \in \text{argmin}_{u' \in \mathcal{U}} \frac{1}{N_s} v_{u'}$
```

4.4. Finite history window approach as a baseline

In this subsection, we introduce a finite history window approach, which is widely employed in RL (Murphy, 2000) for comparison. Different from the above model-based and on-line method, it is a model-free and off-line method. For this problem, the control space is the same as the problem in Section 2. We use o_k as an observation at time k, which includes a stored observation with length $m \ge 1$ and control input with length $n \ge 0$ given by $o_k \triangleq [z_{k-m}, \ldots, z_{k-1}, u_{k-n}, \ldots, u_{k-1}]^{\mathrm{T}} \in \mathbb{R}^{m+n}$. Here, the control input also needs to be recorded as it also effects the decision-making of the sensor. Denote by $\mathcal{O}$ the set of all possible observations $o \in \mathbb{R}^{m+n}$, and $|\mathcal{O}| = |\mathcal{Z}|^m |\mathcal{U}|^n$. It is straightforward to see that the number of states grows exponentially with the length of the window. Correspondingly, the cost per stage can be obtained from the simulator according to Eq. (7).

With above formulation, we obtain a standard RL problem with $\mathcal{O}$ as state space and $\mathcal{U}$ as control space. It can be solved by many different RL methods and we use Q-learning (Watkins, 1989) as an example. The pseudocode of the algorithm is omitted here due to the limit of space.

Remark 4.3. Rollout is an on-line and simulation-based approach to obtain a suboptimal control. Different from other off-line approaches like Q-learning, it does not need to spend much time to train and does not need to maintain a lookup table. At the

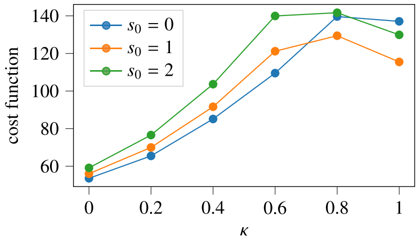


Fig. 2. Cost functions of attacks with different probabilities and different initial states.

same time, rollout can avoid the curse of dimensionality via online computation. Finally, the performance of Q-learning for our focused problem will be affected by the choice of history window size, while the performance of rollout will not be limited by the base policy.

5. Simulation

5.1. A simple illustration of attack effect

An intuitive approach for the sensor scheduling would be to make the decision according to the most recent observation without remembering anything from the past. However, due to the existence of integrity attack, the sensor cannot get the true state and it may not be optimal that the sensor decides whether to send the data with high or low power directly according to the current state. In order to give some insight into our proposed attack model, we provide a simple simulation to illustrate the effect of this kind of attack.

The system parameters are as follows:

$$A = \begin{bmatrix} 1.2 & 0.3 \\ 0.3 & 0.8 \end{bmatrix}, \qquad C = \begin{bmatrix} 1 & 1.7 \\ 0.3 & 1 \end{bmatrix}, \qquad Q = R = I_2.$$

It is straightforward to obtain that the steady-state Kalman filtering error covariance $\bar{P} = \begin{bmatrix} 1.7249 & -0.7250 \\ -0.7250 & 0.5144 \end{bmatrix}$.

The energy consumptions of different levels are tuning parameters. We have tested different combinations and here we present one choice. They are set as $10 \operatorname{tr} \bar{P}$ and $2 \operatorname{tr} \bar{P}$. The successful arrival rate υ for lower energy is set as 0.4. The discount factor α is set to be 0.9. For illustration purpose, we set $\kappa_0 = \kappa_1 = \kappa$ for the rest of this section. However, the method applies to the general case where $\kappa_0 \neq \kappa_1$.

When the above intuitive approach is employed for sensor scheduling, Fig. 2 shows that the cost function values with different attack probabilities and initial states. From this figure, one can see that under the above parameter settings, the optimal attack is of flip probability between 0 and 1.

5.2. Threshold policy of underlying MDP

In this subsection, we provide a numerical example to verify the threshold policy with different weight parameters β's and arrival rate υ's. Other related parameters are the same as the ones in the above subsection. Here, we use the value iteration to compute the optimal policy. The optimal policies under different weight parameter β's and arrival rate υ's are shown in Table 2.

From this table, one can obtain that with the weight parameter β increasing, the optimal threshold increases, which is expected due to higher weight on the energy consumption. Besides, we can see that as the arrival rate υ increases, the optimal threshold increases, which is also expected since the sensor tends to send the packet with low energy due to the increase of the arrival rate.

Table 2
The threshold ϵ^* under different β's and υ's.

Parameters	β with $\upsilon = 0.4$				υ with $\beta = 0.6$			
	0.2	0.4	0.6	0.8	0.2	0.4	0.6	0.8
ϵ^*	2	3	4	5	3	4	5	6

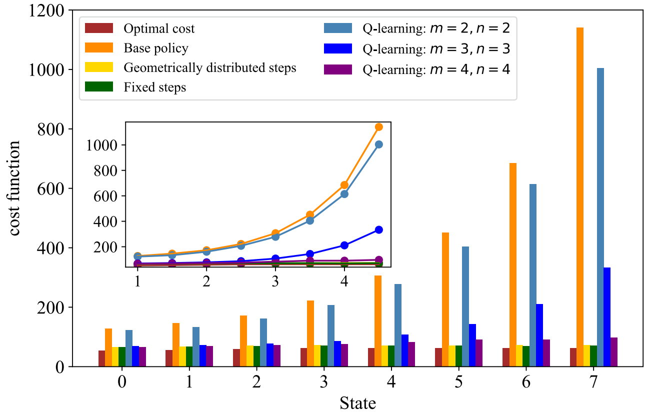


Fig. 3. Performance under different approaches with $\kappa = 0.5$ and $\beta = 0.6$.

5.3. Comparison with rollout policy and finite history window approach

In this section, we consider the sensor scheduling under integrity attack. The attack probability κ is set as 0.5 and the weight parameter β is set as 0.6. Other related parameters are the same as the ones in the above subsection. From Table 2, it is straightforward to get that the optimal threshold is $\epsilon^* = 4$ when $\beta = 0.6$. Here, we truncate the state space and the state ℓ is set as 7. Here, the base policy that we use is

$$\mu_b(b) = \begin{cases} 0, & \sum_{i=0}^{3} b(i) < \sum_{i=4}^{7} b(i), \\ 1, & \text{o.w..} \end{cases}$$

In Fig. 3, the brown bar denotes the true values of the optimal costs for each state; the dark orange, gold and dark green bars denote the cost functions of the base policy, rollout policies with geometrically distributed and fixed truncated steps; the steel blue, blue, and purple bars represent the cost functions using the policies obtained through Q-learning with different window sizes. It is obvious that the performance of the rollout policy is much better than the base policy, and the cost functions of the rollout policies with geometrically distributed and fixed truncated steps are close to the ones of the true optimal costs. It is also shown that with the window size increases, the corresponding cost decreases. This is expected as a better Q-value can be obtained with increased m and n. By computing the stationary distribution under optimal policy for the underlying MDP, we can obtain that $\phi = \{0.2, 0.2, 0.2, 0.2, 0.2\}$ under the designed parameters. Then we compute the two norms of the difference between the optimal cost function and other cost functions of other suboptimal policies from state 0 to state 4 and they are 333.9849 (Base policy), 24.4510 (Geometrically distributed steps), 23.2594 (Fixed steps), 298.3401 (Q-learning: $m = 2, n = 2$), 59.0104 (Q-learning: $m = 3, n = 3$), and 33.2229 (Q-learning: $m = 4, n = 4$). We can see that the performance of the rollout policy is also better than the ones obtained via Q-learning with all different window sizes despite that no offline training is needed for rollout.

6. Conclusion

This paper studied the scheduling of sensor transmission problem for remote state estimation under integrity attacks. It was proved that the underlying MDP has a threshold type optimal policy. Thus we simplified the problem by truncating the state space. When an integrity attack is present, the problem was formulated as a POMDP. The existence of optimal policy for the MDP with belief state induced by the POMDP was studied and it was proved that the monotonicity of value function cannot be established via MLR ordering. The main result of this work is a suboptimal, on-line and model-based approach based on the approximation in value space and implemented through rollout with fixed and geometrically distributed truncated steps, and corresponding performance guarantees were provided. Furthermore, numerical examples were provided to demonstrate the effectiveness of the proposed approaches when compared with a finite history window approach. For the future work, how to design mitigation mechanisms against more intelligent attacker is of great interest. We are also interested in studying the computational cost for real-time operation, including but not limited to finding an appropriate metric to characterize the computational cost.

References

Astrom, K. J. (1965). Optimal control of Markov processes with incomplete state information. *Journal of Mathematical Analysis and Applications*, *10*(1), 174–205.

Baheti, R., & Gill, H. (2011). Cyber-physical systems. *The Impact of Control Technology*, *12*(1), 161–166.

Bertsekas, D. P. (1976). *Dynamic programming and stochastic control*. USA: Academic Press Inc., ISBN: 0120932504.

Bertsekas, D. P. (2018). *Abstract dynamic programming* (2nd ed.). Athena Scientific.

Bertsekas, D. P. (2019). *Reinforcement learning and optimal control*. Athena Scientific Belmont, MA.

Bertsekas, D. P., & Castanon, D. A. (1999). Rollout algorithms for stochastic scheduling problems. *Journal of Heuristics*, *5*(1), 89–108.

Bertsekas, D. P., Tsitsiklis, J. N., & Wu, C. (1997). Rollout algorithms for combinatorial optimization. *Journal of Heuristics*, *3*(3), 245–262.

Bhattacharya, S., Badyal, S., Wheeler, T., Gil, S., & Bertsekas, D. (2020). Reinforcement learning for POMDP: Partitioned rollout and policy iteration with application to autonomous sequential repair problems. *IEEE Robotics and Automation Letters*, *5*(3), 3967–3974.

Ding, K., Ren, X., Quevedo, D. E., Dey, S., & Shi, L. (2020). Defensive deception against reactive jamming attacks in remote state estimation. *Automatica*, *113*, Article 108680.

Ding, K., Ren, X., & Shi, L. (2016). Deception-based sensor scheduling for remote estimation under dos attacks. *IFAC-PapersOnLine*, *49*(22), 169–174.

Guo, Z., Wang, J., & Shi, L. (2017). Optimal denial-of-service attack on feedback channel against acknowledgment-based sensor power schedule for remote estimation. In *2017 IEEE 56th annual conference on decision and control (CDC)* (pp. 5997–6002). IEEE.

Han, D., Cheng, P., Chen, J., & Shi, L. (2013). An online sensor power schedule for remote state estimation with communication energy constraint. *IEEE Transactions on Automatic Control*, *59*(7), 1942–1947.

Krishnamurthy, V. (2016). *Partially observed markov decision processes: from filtering to controlled sensing*. Cambridge University Press, http://dx.doi.org/10.1017/CBO9781316471104.

Lax, P. D. (2014). *Functional analysis*. John Wiley & Sons.

Leong, A. S., Ramaswamy, A., Quevedo, D. E., Karl, H., & Shi, L. (2020). Deep reinforcement learning for wireless sensor scheduling in cyber–physical systems. *Automatica*, *113*, Article 108759.

Lewis, F. L., et al. (2004). Wireless sensor networks. *Smart Environments: Technologies, Protocols, and Applications*, *11*, 46.

Li, Y., Johansson, K. H., & Mårtensson, J. (2019). Lambda-policy iteration with randomization for contractive models with infinite policies: Well posedness and convergence (extended version). arXiv preprint arXiv:1912.08504.

Li, Y., Quevedo, D. E., Dey, S., & Shi, L. (2015). Fake-acknowledgment attack on ACK-based sensor power schedule for remote state estimation. In *2015 54th IEEE conference on decision and control (CDC)* (pp. 5795–5800). IEEE.

Li, Y., Quevedo, D. E., Dey, S., & Shi, L. (2016). A game-theoretic approach to fake-acknowledgment attack on cyber-physical systems. *IEEE Transactions on Signal and Information Processing over Networks*, *3*(1), 1–11.

Li, Y., Quevedo, D. E., Lau, V., & Shi, L. (2013). Online sensor transmission power schedule for remote state estimation. In *52nd IEEE conference on decision and control* (pp. 4000–4005). IEEE.

Mo, Y., Sinopoli, B., Shi, L., & Garone, E. (2012). Infinite-horizon sensor scheduling for estimation over lossy networks. In *2012 IEEE 51st IEEE conference on decision and control (CDC)* (pp. 3317–3322). IEEE.

Murphy, K. P. (2000). A survey of POMDP solution techniques. *Environment*, *2*, X3.

Patek, S. D. (2007). Partially observed stochastic shortest path problems with approximate solution by neurodynamic programming. *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, *37*(5), 710–720.

Qin, J., Li, M., Shi, L., & Yu, X. (2017). Optimal denial-of-service attack scheduling with energy constraint over packet-dropping networks. *IEEE Transactions on Automatic Control*, *63*(6), 1648–1663.

Qin, J., Li, M., Wang, J., Shi, L., Kang, Y., & Zheng, W. X. (2020). Optimal denial-of-service attack energy management against state estimation over an SINR-based network. *Automatica*, *119*, Article 109090.

Rajkumar, R., Lee, I., Sha, L., & Stankovic, J. (2010). Cyber-physical systems: the next computing revolution. In *Design automation conference* (pp. 731–736). IEEE.

Shi, L., Johansson, K. H., & Qiu, L. (2011). Time and event-based sensor scheduling for networks with limited communication resources. *IFAC Proceedings Volumes*, *44*(1), 13263–13268.

Tesauro, G., & Galperin, G. R. (1997). On-line policy improvement using Monte-Carlo search. In *Advances in neural information processing systems* (pp. 1068–1074).

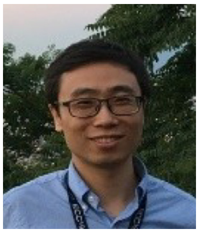
Watkins, C. J. C. H. (1989). *Learning from delayed rewards* (Ph.D. thesis), King's College, Cambridge.

Wu, S., Ren, X., Jia, Q.-S., Johansson, K. H., & Shi, L. (2019). Learning optimal scheduling policy for remote state estimation under uncertain channel condition. *IEEE Transactions on Control of Network Systems*.

Zhang, H., Cheng, P., Shi, L., & Chen, J. (2015). Optimal denial-of-service attack scheduling with energy constraint. *IEEE Transactions on Automatic Control*, *60*(11), 3023–3028.

Zhang, H., Qi, Y., Wu, J., Fu, L., & He, L. (2016). Dos attack energy management against remote state estimation. *IEEE Transactions on Control of Network Systems*, *5*(1), 383–394.

Hanxiao Liu received the Bachelor of Engineering degree from the School of Control Science and Engineering, Shandong University, Jinan, China, and the joint Ph.D. degree in electrical and electronic engineering from Nanyang Technological University, Singapore and electrical engineering from KTH Royal Institute of Technology, Sweden, in 2017, and 2021, respectively. She is currently a lecturer at the School of Artificial Intelligence, Shanghai University, Shanghai, China. Her research interests include optimal control, reinforcement learning, and security and privacy of cyber–physical system.



Yuchao Li is a Ph.D. student with the Division of Decision and Control Systems, KTH Royal Institute of Technology since August 2017. He received B.Eng. in Mechanical Engineering from the Honors School, Harbin Institute of Technology in 2015, and M.Sc. in Mechatronics from the Department of Machine Design, KTH in 2016. His research interests are optimal control, reinforcement learning and their applications in transportation systems.

Karl Henrik Johansson is Professor with the School of Electrical Engineering and Computer Science at KTH Royal Institute of Technology in Sweden and Director of Digital Futures. He received M.Sc. degree in Electrical Engineering and Ph.D. in Automatic Control from Lund University. He has held visiting positions at UC Berkeley, Caltech, NTU, HKUST Institute of Advanced Studies, and NTNU. His research interests are in networked control systems and cyber–physical systems with applications in transportation, energy, and automation networks. He is President of the European Control Association and member of the IFAC Council, and has served on the IEEE Control Systems Society Board of Governors and the Swedish Scientific Council for Natural Sciences and Engineering Sciences. He has received several best paper awards and other distinctions from IEEE, IFAC, and ACM. He has been awarded Swedish Research Council Distinguished Professor, Wallenberg Scholar with the Knut and Alice Wallenberg Foundation, Future Research Leader Award from the Swedish Foundation for Strategic Research, the triennial IFAC Young Author Prize, and IEEE Control Systems Society Distinguished Lecturer. He is Fellow of the IEEE and the Royal Swedish Academy of Engineering Sciences.

Jonas Mårtensson received the M.Sc. degree in vehicle engineering and the Ph.D. degree in automatic control from the KTH Royal Institute of Technology, Stockholm, Sweden, in 2002 and 2007, respectively. In 2016, he was appointed as a Docent. He currently is Professor with the Division of Decision and Control Systems, KTH Royal Institute of Technology. He is also engaged as the Director of the Integrated Transport Research Laboratory and the Thematic Leader for the area transport in the information age with the KTH Transport Platform. His research interests are cooperative and autonomous transport systems, heavy-duty vehicle platooning, collaborative adaptive cruise control, look-ahead platooning, route optimization and coordination for platooning, path planning and predictive control of autonomous heavy vehicles, and related topics.

Lihua Xie received the Ph.D. degree in electrical engineering from the University of Newcastle, Australia, in 1992. Since 1992, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, where he is currently a professor and Director, Delta-NTU Corporate Laboratory for Cyber–physical Systems and Centre for Advanced Robotics Technology Innovation. He served as the Head of Division of Control and Instrumentation from July 2011 to June 2014. Dr Xie's research interests include robust control and estimation, networked control systems, multi-agent networks, localization and unmanned systems. He is an Editor-in-Chief for Unmanned Systems and has served as an editor of IET Book Series in Control and an Associate Editor of a number of journals including IEEE Transactions on Automatic Control, Automatica, IEEE Transactions on Control Systems Technology, IEEE Transactions on Control of Network Systems, and IEEE Transactions on Circuits and Systems-II. He was an IEEE Distinguished Lecturer and elected member of Board of Governors, IEEE Control System Society (Jan 2016–Dec 2018). Dr Xie is Fellow of IEEE, IFAC and Academy of Engineering Singapore.