2022 IEEE 61st Conference on Decision and Control (CDC)
December 6-9, 2022. Cancún, Mexico

Structural analyses of a parsimonious watermarking policy for data deception attack detection in networked control systems

Arunava Naha[1], André Teixeira[2], Anders Ahlén[1] and Subhrakanti Dey[1]

2022 IEEE 61st Conference on Decision and Control (CDC) | 978-1-6654-6761-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/CDC51059.2022.9993201

Abstract— **In this paper, we perform structural analyses of a parsimonious watermarking policy, which minimizes the average detection delay (ADD) to detect data deception attacks on networked control systems (NCS) for a fixed upper bound on the false alarm rate (FAR). The addition of physical watermarking to the control input of a NCS increases the probability of attack detections with an increase in the control cost. Therefore, we formulate the problem of data deception attack detection for NCS with the facility to add physical watermarking as a stochastic optimal control problem. Then we solve the problem by applying dynamic programming value iterations and find a parsimonious watermarking policy that decides to add watermarking and detects attacks based on the estimated posterior probability of attack. We analyze the optimal policy structure and find that it can be a one, two or three threshold policy depending on a few parameter values. Simulation studies show that the optimal policy for a practical range of parameter values is a two-threshold policy on the posterior probability of attack. Derivation of a threshold-based policy from the structural analysis of the value iteration method reduces the computational complexity during the runtime implementation and offers better structural insights. Furthermore, such an analysis provides a guideline for selecting the parameter values to meet the design requirements.**

I. Introduction

Cyber-Physical Systems (CPS) are considered the next generation of intelligent systems that integrate the cyber and physical layers. The cyber layer consists of software programs. On the other hand, the physical layer consists of the plant or the process being controlled or monitored along with other physical components such as sensors, networks, actuators, and embedded computers. The cyber and physical layers communicate over a network, which is a wireless network in most cases, and provide reliable, accurate, robust, efficient and autonomous operations without human involvement [1]. Due to their enumerated advantages, CPS are getting deployed for various applications, such as advanced healthcare, intelligent transportation systems, and smart grids, to name a few. However, due to the use of commodity software, off-the-shelf networking components, and unattended operations, CPS are vulnerable to adversarial attacks [2]. Therefore, we need to ensure the safety and security of CPS before the large-scale adaptation of such systems for safety-critical applications. The objective of an adversary may be to gain access to user-sensitive data, or the attacker may want to cause some damage to the system by tempering its availability, integrity or reliability. Therefore, an attack on the CPS may cause monetary loss and pose a serious threat to human safety and must be detected as early as possible to reduce the extent of the damage.

The deception attack is one type of adversarial attack on the physical layer of CPS. Under the deception attack, the attacker feeds the CPS with harmful or unwanted information to cause some damage [4], [2] and simultaneously tries to remain stealthy. As reported in the literature, cryptography-based cyber security measures may not be adequate to protect CPS against attacks on the physical layer [2]. For example, in the Stuxnet attack [3], the attackers exploited a known vulnerability of a commercially used operating system and caused damage to a uranium enrichment plant in Iran. In the attack, the attackers issued harmful exogenous inputs to the centrifugal pumps to increase their speed beyond the safety limit. Furthermore, previously recorded observations from the sensors were relayed during the attack to remain stealthy, and such attacks are called replay attacks. The replay attack is one type of data deception attack. In this paper, we have studied the data deception attacks on the physical layer of a networked control system (NCS), a widely used form of CPS, where the attacker replaces the true measurements with fake data.

A. Prior work

A well-studied defence mechanism against data deception attacks is the addition of physical watermarking to the control inputs or the observations of NCS. In general, the existing methods perform various statistical tests on the received observations or the innovation signal from the Kalman estimator to detect the presence of an attack. The concept of physical watermarking is analogous to digital watermarking, which is used to authenticate the rightful owner of the digital content. In [2], a watermarking signal generated from a hidden Markov model (HMM) is added to the optimal control input from a linear quadratic Gaussian (LQG) controller and performed χ^2 test on the innovation signal for attack detections. In another approach, the received observations are used to generate two residue signals, which will have finite values during an attack and otherwise will remain zero [4]. In general, the attack detection mechanisms reported in the literature perform batch processing and, therefore, do not explicitly address the problem of the quickest detection

*This work is supported by The Swedish Research Council under grants 2017-04053 and 2018-04396, and by the Swedish Foundation for Strategic Research.

[1]Arunava Naha, Anders Ahlén and Subhrakanti Dey are with the Department of Electrical Engineering, Uppsala University, 75103 Uppsala, Sweden. arunava.naha@angstrom.uu.se, Anders.Ahlen@angstrom.uu.se, and Subhra.Dey@signal.uu.se

[2]André Teixeira is with the Department of Information Technology, Uppsala University, PO Box 337, SE-75105, Uppsala, Sweden. andre.teixeira@it.uu.se

of attacks. In our prior work, we have studied the problem of the quickest detection of data deception attacks on NCS by applying the cumulative sum (CUSUM) technique using the joint distributions of the watermarking signal and the innovation signal [5], [6]. Furthermore, The reported method is optimal in the sense that it minimises the supremum of the average detection delay (SADD) for a fixed lower limit on the average run length between two false alarm events (ARL).

One inherent problem of the physical watermarking-based defence mechanism is that it increases the control cost [2], [5]. Since the attack on the CPS is a rare event, adding watermarking signal for a very long time before the attack start point increases the control cost considerably. Researchers are studying different techniques to reduce the control cost due to physical watermarking. In one approach, the watermarking signal is added periodically to the control input in such a way to maintain a balance between the detection delay and the control cost [7]. In a different approach, the watermarking signal is added directly to the observation before sending it over the wireless network [8], [9]. Then at the receiving end, the authenticity of the received data is checked by various statistical tests. Finally, the added watermarking signal is filtered out from the received observation before using it in the controller. Since the watermarking signal is filtered out, these methods do not cause any increase in the control cost. However, these methods may fail in the scenario where the attacker hijacks the sensor node and feeds the fake data before the addition of the watermarking.

B. Contributions

In our prior work, we have addressed the problem of increased control cost by deriving a parsimonious watermarking policy that minimises the average detection delay (ADD) for fixed upper limits on the false alarm rates (FAR), and an average number of watermarking (ANW) events before the attack start point [10], [11]. We formulated the problem as a stochastic optimal control problem and solved it by using dynamic programming value iterations. In this current paper, we perform an in-depth study of the structure of the derived parsimonious watermarking policy by the value iterations. By analysing the structure of the Bellman equation and the numerical simulation results, we found that the optimal policy can be a one, two or three threshold policy on the posterior probability of attack, p_k. Our study shows that the number of thresholds in the optimal policy depends on a few parameter values. Furthermore, the optimal policy is a two-threshold policy for the practical range of parameter values. In a two-threshold policy, a watermarking signal is added to the control input at the $(k+1)$-th instant in time if there is evidence of an attack being present in the system, *i.e.*, p_k is greater than or equal to the first threshold Th^s. On the other hand, we decide that an attack is present in the system and terminate the process when we gain much higher confidence, *i.e.*, $p_k \geq Th^d$, where Th^d is the second threshold, and $Th^d > Th^s$. Implementing an optimal policy derived from the value iteration is computationally complex since the Bellman equation needs to be solved at every time instant. However, finding a structure in the optimal policy, such as a threshold-based policy, reduces the computational requirements. In the current paper, we have used the sequential quickest change detection theories studied in [12], [13], [14].

C. Organization

This paper is organized as follows. The NCS model considered in this paper and the data deception attack model are discussed in Section II and Section III, respectively. Section IV formulates the problem as a stochastic optimal control problem. The solution approach and the structure of the optimal policy are studied in Section V. Section VI presents and discusses numerical simulation results to illustrate the proposed parsimonious watermarking scheme and its strength and weakness. Finally, Section VII concludes the paper.

II. System Models during Normal Operations

We have considered a NCS, as shown in Fig. 1, which is one of the most widely used forms of CPS. Figure 1 illustrates the block diagram of a standard NCS during normal operations. Such models have also been used in many works of literature [2]. As illustrated in Fig. 1, the NCS considered for this study consists of a linear time-invariant (LTI) plant, sensors, actuators, LQG controller and Kalman state estimator. The sensors observe the required parameters of the plant, and the observations are sent to the state estimator via a wireless network. The LQG controller uses the estimated states from the Kalman estimator to evaluate the optimal control input, which is then sent to the actuators via a wireless link. Finally, the actuators take necessary actions according to the received control inputs and change the plant parameters.

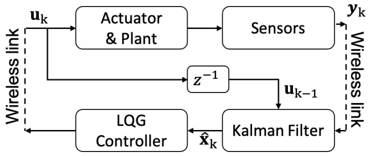

Fig. 1: Schematic diagram of the NCS during normal operation.

The plant is assumed to be a LTI system with the following state-space model,

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k. \qquad (1)$$

Here $\mathbf{x}_k \in \mathbb{R}^n$ and $\mathbf{u}_k \in \mathbb{R}^p$ are the state and input vectors, respectively, at the k-th instant in time. $\mathbf{w}_k \in \mathbb{R}^n \sim \mathcal{N}(0, \mathbf{Q})$ denotes the independent and identically distributed (iid) process noise. The process noise variance $\mathbf{Q} > \mathbf{0}$, which denotes $\mathbf{Q}$ is a positive definite matrix. The observations from the sensors are related to the states as

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{v}_k, \qquad (2)$$

where $\mathbf{y}_k \in \mathbb{R}^m$ is the observation vector at the k-th instant in time. The measurement noise $\mathbf{v}_k \in \mathbb{R}^m \sim \mathcal{N}(0, \mathbf{R})$ is also assumed to be iid and $\mathbf{R} > \mathbf{0}$. In addition to that, process and measurement noises are assumed to be uncorrelated to each other and also to the initial state vector.

The Kalman filter (KF) estimates the state vector in two steps at every time instant. First, it performs the time update using the system model information and the control input (see (3)), and then performs a measurement update using the received measurements (see (4)).

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}\mathbf{u}_{k-1}, \tag{3}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}\gamma_k, \tag{4}$$

where $\mathbf{K}$ is the steady-state Kalman gain and γ_k is the innovation signal at the k-th time instant. $\hat{\mathbf{x}}_{k|k-1} = \mathrm{E}[\mathbf{x}_k|\Psi_{k-1}]$ and $\hat{\mathbf{x}}_{k|k} = \mathrm{E}[\mathbf{x}_k|\Psi_k]$ are the Kalman predicted and filtered states, respectively. $\mathrm{E}[\cdot]$ denotes the expectation operator. Additionally, Ψ_k denotes the set of all input and measurement information up to the k-th time instant. γ_k during the normal system operation is evaluated as

$$\gamma_k = \mathbf{y}_k - \mathbf{C}\hat{\mathbf{x}}_{k|k-1}. \tag{5}$$

We assume that the system has been operational for a very long time, and from $k \geq 0$, the system is at a steady state. Therefore, we can derive the steady-state Kalman gain $\mathbf{K}$ as follows,

$$\mathbf{K} = \mathbf{P}\mathbf{C}^{'}\left(\mathbf{C}\mathbf{P}\mathbf{C}^{'} + \mathbf{R}\right)^{-1}. \tag{6}$$

Here $(\cdot)^{'}$ denotes transpose of a matrix. $\mathbf{P}$ is the steady-state state error covariance matrix, *i.e.*, $\mathbf{P} = \mathrm{E}\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})^{'}\right]$, which is the solution to the following algebraic Riccati equation,

$$\mathbf{P} = \mathbf{A}\mathbf{P}\mathbf{A}^{'} + \mathbf{Q} - \mathbf{A}\mathbf{P}\mathbf{C}^{'}\left(\mathbf{C}\mathbf{P}\mathbf{C}^{'} + \mathbf{R}\right)^{-1}\mathbf{C}\mathbf{P}\mathbf{A}^{'}. \tag{7}$$

The optimal control input from the LQG controller is derived by minimizing the following infinite horizon expected control cost,

$$J = \lim_{N\to\infty} \mathrm{E}\left[\frac{1}{2N+1}\left\{\sum_{k=-N}^{N}\left(\mathbf{x}_k^{'}\mathbf{W}\mathbf{x}_k + \mathbf{u}_k^{'}\mathbf{U}\mathbf{u}_k\right)\right\}\right], \tag{8}$$

where $\mathbf{W} \geq \mathbf{0}$ and $\mathbf{U} \geq \mathbf{0}$ are the two user selected weight matrices. In this context, $\geq \mathbf{0}$ denotes both the matrices are positive semi-definite. Minimization of (8) provides the optimal control input $\mathbf{u}_k^*$ at the k-th instant as a linear function of the estimated states from the KF as follows,

$$\mathbf{u}_k^* = \mathbf{L}\hat{\mathbf{x}}_{k|k}, \text{ where} \tag{9}$$

$$\mathbf{L} = -\left(\mathbf{B}^{'}\mathbf{S}\mathbf{B} + \mathbf{U}\right)^{-1}\mathbf{B}^{'}\mathbf{S}\mathbf{A}. \tag{10}$$

Here $\mathbf{S}$ is the solution to the following algebraic Riccati equation,

$$\mathbf{S} = \mathbf{A}^{'}\mathbf{S}\mathbf{A} + \mathbf{W} - \mathbf{A}^{'}\mathbf{S}\mathbf{B}\left(\mathbf{B}^{'}\mathbf{S}\mathbf{B} + \mathbf{U}\right)^{-1}\mathbf{B}^{'}\mathbf{S}\mathbf{A}. \tag{11}$$

III. Attack Model

The NCS discussed in Section II is vulnerable to adversarial attacks on the sensors, actuators, and networks, and such attacks may not be detected or prevented by the existing cybersecurity measures. In the attack model considered for this paper, the attacker replaces all the sensor measurements with fake data generated by its own system, as shown in Fig. 2. We assume the attacker knows about the system and controller parameters but does not know the instantaneous values of the noise vectors or the watermarking signal. Furthermore, the attacker can gain access to the sensor nodes and replace true observations with fake data.

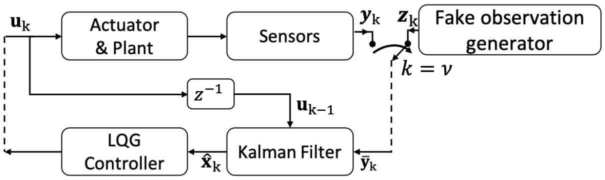

Fig. 2: Schematic diagram of the system during under data deception attack.

We assume the attacker will design the fake data $\mathbf{z}_k$ in such a way that the statistical properties of $\mathbf{z}_k$ will be similar to that of the true observation $\mathbf{y}_k$. Since $\mathbf{y}_k$ is a stationary random process at the steady state, which is statistically dependent on its past values, we assume the attacker will generate $\mathbf{z}_k$ from the following linear stationary stochastic process,

$$\mathbf{z}_k = \mathbf{A}_a\mathbf{z}_{k-1} + \mathbf{w}_{a,k-1}. \tag{12}$$

Here $\mathbf{w}_{a,k} \sim \mathcal{N}(0, \mathbf{Q}_a)$ is the iid noise vector at the k-th time instant, and $\mathbf{Q}_a \in \mathbb{R}^{m\times m} > \mathbf{0}$.

To demonstrate the effectiveness of the attack model considered in this paper, we simulate the system shown in Fig. 2 using the model parameters given in Appendix I. From the plots of true and estimated states in Fig. 3, we observe that the true states become unstable and unbounded shortly after the attack start point. On the other hand, the estimated states based on the received fake observations do not change much, making the detection of such attacks challenging. A more detailed discussion about the attack model (12) is available in [6].

IV. Problem Formulation

This section discusses the formulation of the quickest attack detection problem using physical watermarking parsimoniously. We perform the following hypothesis test at each time instant.

H_0: No attack present.
H_1: Attack present in the system.

To formulate the quickest attack detection problem as a stochastic optimal control problem, we define the following

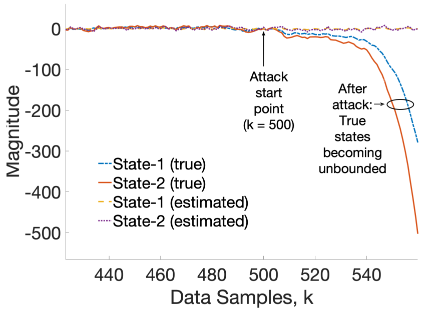

Fig. 3: Schematic diagram of the system during a data deception attack.

two decision variables, s_k and d_k.

$$s_k = \begin{cases} 0, & \text{no watermarking at } (k+1)\text{-th time instant} \\ 1, & \text{add watermarking at } (k+1)\text{-th time instant.} \end{cases} \tag{13}$$

$$d_k = \begin{cases} 0, & \text{Select Hypothesis } H_0. \text{ Process continues.} \\ 1, & \text{Select Hypothesis } H_1. \text{ Process terminates.} \end{cases} \tag{14}$$

Furthermore, we assume that the attack start point, τ, is a random variable (RV), which has geometric distribution with parameter ρ. This assumption is also followed in many works of literature on the sequential change detection problem [13], [15]. In addition to that, as discussed in [15], a particular property of the geometric distribution enables us to derive an analytical expression of ADD by applying non-linear renewal theory.

A. Parsimonious Watermarking Scheme

We consider the watermarking signal to be an iid noise with Gaussian distribution, *i.e.*, $\mathbf{e}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{\Sigma}_e)$. As studied in [2], [6], if the iid watermarking signal is added to the LQG control input at all the time instances, then the increase in the control cost, ΔLQG_a, during the normal system operation will be as follows,

$$\Delta LQG_a = tr\left(\mathbf{H}\mathbf{\Sigma}_e\right), \tag{15}$$

$$\text{where } \mathbf{H} = \mathbf{B}^{'}\mathbf{\Sigma}_L\mathbf{B} + \mathbf{U}, \tag{16}$$

and $\mathbf{\Sigma}_L$ is the solution to the following Lyapunov equation

$$\left(\mathbf{A}+\mathbf{BL}\right)^{'}\mathbf{\Sigma}_L\left(\mathbf{A}+\mathbf{BL}\right) - \mathbf{\Sigma}_L + \mathbf{L}^{'}\mathbf{UL} + \mathbf{W} = 0. \tag{17}$$

However, watermarking is not added to the control input in all instances in the proposed method. The decision of adding watermarking is controlled by the decision variable s_k (13) as illustrated in Fig. 4. On the other hand, the value of s_k is decided based on some evidence, *i.e.*, the posterior probability of attack p_k. In the proposed scheme, we aim to limit the average number of watermarking (ANW) events before the attack start point below a user-defined threshold ANW_{th}. Such parsimonious use of watermarking reduces the control cost. The relationship between ANW and the increase in the control cost ΔLQG for the proposed method is given in (18). The derivation of (18) is provided in [11].

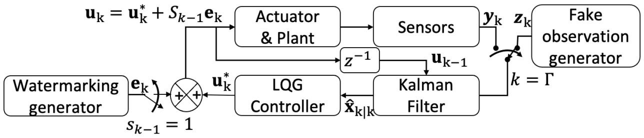

Fig. 4: Parsimonious watermarking scheme.

$$\Delta LQG = \rho ANW \Delta LQG_a. \tag{18}$$

B. Formulation of Bellman Equation

Our objective is to find an optimal policy to satisfy the following constrained optimization problem.

$$\begin{aligned} &\min_{u_d \in \mathcal{U}} \; ADD \\ &\text{s.t. } FAR \leq FAR_{th} \\ &ANW \leq ANW_{th}, \end{aligned} \tag{19}$$

where FAR_{th} and ANW_{th} are the two user-defined thresholds. Here u_d denotes the policy, and $\mathcal{U}$ is the set of all permissible stationary deterministic policies (see Table I). The instantaneous value of the policy is denoted by $u_{d,k}$ and the relationship between the policy value $u_{d,k}$ and the corresponding decision variables s_k and d_k is given in Table I.

TABLE I: $\mathcal{U}$

$u_{d,k}$	d_k	s_k
1	0	0
2	0	1
3	1	0

Note that, if $d_k = 1$, then the process will immediately terminate, and there will be no use of adding watermarking at the $(k+1)$-th time instant. Therefore, the combination $(s_k = 1, d_k = 1)$ has been ignored. We can write ADD, FAR and ANW in terms of the posterior probability of attack p_k, and the decision variables s_k and d_k as follows,

$$ADD = \sum_{k=1}^{\tau} p_k \mathbb{1}_{\{d_k=0\}}, \tag{20}$$

$$FAR = \sum_{k=1}^{\tau} (1-p_k)\mathbb{1}_{\{d_k=1\}}, \text{ and} \tag{21}$$

$$ANW = \sum_{k=1}^{\tau} (1-p_k)\mathbb{1}_{\{s_k=1\}}\mathbb{1}_{\{d_k=0\}}. \tag{22}$$

First, we convert the constrained optimization problem in (19) into an unconstrained cost function J using the Lagrangian multipliers λ_f and λ_e as follows,

$$J^* = \min_{u_d \in \mathcal{U}} \; ADD + \lambda_f FAR + \lambda_e ANW. \tag{23}$$

Then using (20)-(22), we transform (23) into the following Bellman equation,

$$\begin{aligned} J\left(p_k\right) = \min_{u_{d,k}} \Big[& p_k \mathbb{1}_{\{d_k=0\}} + \lambda_f \left(1-p_k\right)\mathbb{1}_{\{d_k=1\}} \\ & + \lambda_e \left(1-p_k\right)\mathbb{1}_{\{s_k=1\}}\mathbb{1}_{\{d_k=0\}} + B_0\left(p_k\right)\mathbb{1}_{\{s_k=0\}}\mathbb{1}_{\{d_k=0\}} \\ & + B_1\left(p_k\right)\mathbb{1}_{\{s_k=1\}}\mathbb{1}_{\{d_k=0\}} \Big], \end{aligned} \tag{24}$$

where $B_0(p_k) = \mathrm{E}[J(\phi_0(p_k))]$, and $B_1(p_k) = \mathrm{E}[J(\phi_1(p_k))]$ are the expected total costs from $(k+1)$-th time instant till the termination of the process when $s_k = 0$ and $s_k = 1$, respectively, and $d_k = 0$. The functions $\phi_0(\cdot)$ and $\phi_1(\cdot)$ denote the one step time update functions of p_k, *i.e.*, $p_k = \phi_0(p_{k-1})$, when $s_{k-2} = 0$, and $p_k = \phi_1(p_{k-1})$, otherwise. Note that when $d_k = 1$, the process terminates immediately, so there will be no additional cost after the k-th time instant. The expressions of $\phi_0(\cdot)$ and $\phi_1(\cdot)$, and a detailed discussion on the formulation of the Bellman equation from the constrained optimization problem can be found in [11]. However, the same has been removed from the current paper to focus mainly on the structural analysis of the optimal policy and also due to the space constraints. The Bellman equation is solved by value iterations, as discussed in the following section.

V. Structural Analysis of the Optimal Policy

In this section, we discuss the solution of the Bellman equation (24) by value iterations [16], and the structure of the optimal policy found from that.

A. Solution of Bellman Equation

We get the following conditions from value iterations to derive the values of the decision variables s_k and d_k, see (25) and (26). Then the subsequent discussion explains how (25) and (26) can be transformed into stationary deterministic threshold-based policy.

$$s_k = \begin{cases} 0, & B_0(p_k) - B_1(p_k) < \lambda_e (1 - p_k), \\ 1, & otherwise. \end{cases} \tag{25}$$

$$d_k = \begin{cases} 0, & p_k + \lambda_e (1 - p_k) \mathbb{1}_{\{s_k=1\}} + B_0(p_k) \mathbb{1}_{\{s_k=0\}} \\ & +B_1(p_k) \mathbb{1}_{\{s_k=1\}} < \lambda_f (1 - p_k), \\ 1, & \text{otherwise.} \end{cases} \tag{26}$$

The accessibility hypothesis discussed in [17] tells us that under this hypothesis, for every stationary deterministic policy $u_d \in \mathcal{U}$, any arbitrary state, say p_k, is accessible from each starting state p_0. According to the theory discussed in [17], under the accessibility hypothesis, the dynamic programming equation (24) is solvable by at least one stationary deterministic policy for each $\lambda_f \geq 0$ and $\lambda_e \geq 0$. Therefore, to satisfy the accessibility hypothesis we discretize the range space of p_k, *i.e.*, $0 \leq p_k \leq 1$. Furthermore, to apply the value iterations, we also discretize the search space of λ_f and λ_e. For each point on the search grid of λ_f and λ_e, we perform value iterations [16] to find the solution of the Bellman equation (24). However, we only keep the best policy that satisfies the given constraints (19).

B. Existence of a stationary deterministic optimal policy

We take the following assumption to ensure that the optimal policy obtained by solving the Bellman equation (24) using value iterations will also satisfy the original constrained optimization problem (19).

Assumption A1: There exist at least one stationary deterministic policy u_d, for which both the constraints as given in (19) will be satisfied.

Now, we need to prove that the optimal policy obtained by value iteration from (24) will be a stationary deterministic policy. By following the similar steps used to prove Lemma 3.1 in [17], we can show that the cost FAR and ANW will be monotone and non-increasing in λ_f and λ_e, respectively. As stated in Lemma 3.3 from [17], the inequality conditions in the original constrained optimization problem (19) will be satisfied for finite values of $\lambda_f \geq 0$ and $\lambda_e \geq 0$ for some deterministic policy due to the monotone and non-increasing properties of FAR and ANW. Finally, under the assumption A1 or the weaker condition of Lemma 3.1, the stationary deterministic optimal policy found by solving the Bellman equation (24) from the unconstrained optimization problem with the Lagrangian multipliers, λ_e and λ_f, will be the solution of the original constrained problem as given in (19) [17].

C. Structural Analysis of Optimal Policy

We perform extensive numerical simulations to analyse the optimal policy structure and the effect of the parameters λ_f, λ_e and ρ on the policy structure. We have used an open-loop unstable multi-input single-output (MISO) system for the simulation study, and Appendix I provides the parameter values used for the simulation. The simulation results are explained using (25)-(26). For the ease of discussion, we will refer to the left and right-hand sides (LHS and RHS) of the inequality in (25) as LHS_s and RHS_s, respectively. Similarly, the left and right-hand sides of the inequality in (26) are denoted as LHS_d and RHS_d, respectively.

From (25)-(26), we can say that when $p_k = 0$ or close to 0, $LHS_s < RHS_s$ and $LHS_d < RHS_d$. Therefore, for $p_k \approx 0$, both the decision variables $s_k = 0$ and $d_k = 0$, *i.e.*, $u_{d,k} = 1$. As p_k increases, $\lambda_e(1 - p_k)$ and $\lambda_f(1 - p_k)$ both will decrease, and eventually $LHS_s > RHS_s$ and $LHS_d > RHS_d$. Now, at what values of p_k such changes will happen depends on λ_f, λ_e and ρ. Moreover, since $J(1) = B_1(1) = B_0(1) = 0$ as the process terminates when $p_k = 1$, $RHS_s = LHS_s$ for atleast two different values of p_k; one obviously at $p_k = 1$. For a relatively high value of λ_e compared to λ_f, the cost for adding watermarking will be relatively high compared to the false alarm event. Therefore, under such situation, it may happen that $LHS_d \geq RHS_d$ before $LHS_s \geq RHS_s$. Since the process terminates when $LHS_d \geq RHS_d$ or $d_k = 1$, we will get a one-threshold optimal policy. In other words, in the one-threshold policy, $s_k = 0$ and $d_k = 0$, *i.e.*, $u_{d,k} = 1$ changes to $s_k = 0$ and $d_k = 1$, *i.e.*, $u_{d,k} = 3$, when $p_k \geq Th^d$. Figures 5 and 6 provide the insights of the one-threshold policy by plotting the LHS and RHS of (25) and (25), for a relatively large λ_e and small λ_f. The corresponding one-threshold optimal policy $u_{d,k}$ is plotted in Fig. 7.

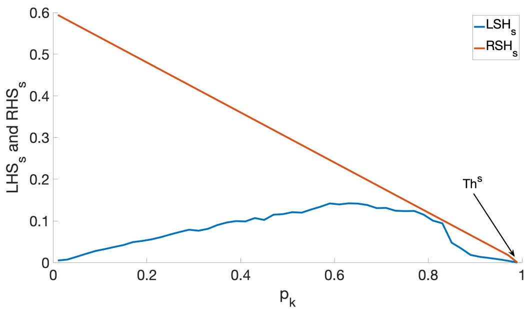

Fig. 5: LHS_s and RHS_s vs. p_k. $\lambda_e = 0.6$, $\lambda_f = 50$, $\rho = 0.001$.

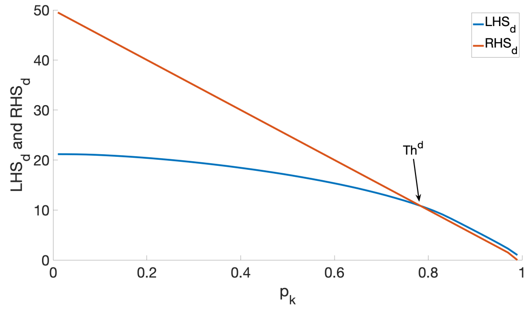

Fig. 6: LHS_d and RHS_d vs. p_k. $\lambda_e = 0.6$, $\lambda_f = 50$, $\rho = 0.001$.

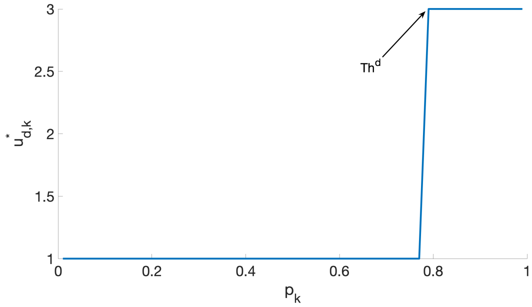

Fig. 7: $u^*_{d,k}$ vs. p_k. $\lambda_e = 0.6$, $\lambda_f = 50$, $\rho = 0.001$.

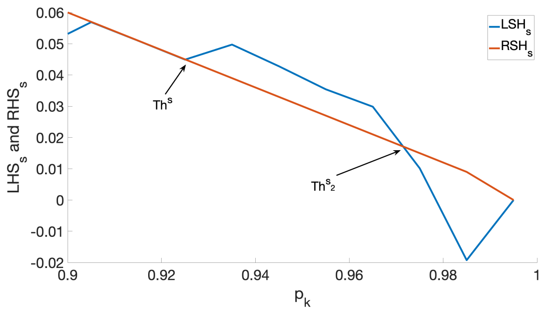

Fig. 8: LHS_s and RHS_s vs. p_k. $\lambda_e = 0.6$, $\lambda_f = 100$, $\rho = 0.5$.

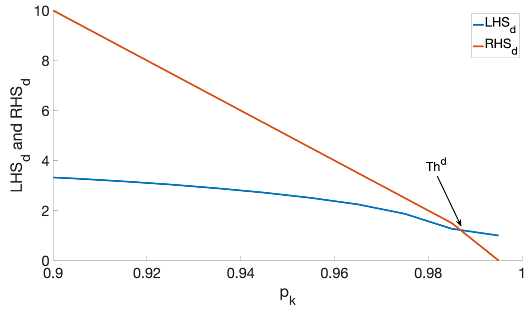

Fig. 9: LHS_d and RHS_d vs. p_k. $\lambda_e = 0.6$, $\lambda_f = 100$, $\rho = 0.5$.

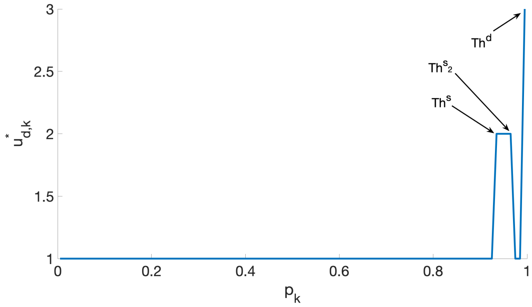

Fig. 10: $u^*_{d,k}$ vs. p_k. $\lambda_e = 0.6$, $\lambda_f = 100$, $\rho = 0.5$.

From simulation studies, we observe that for a higher value of ρ, it may happen that $RHS_s = LSH_s$ for two different values of p_k, say Th^s and Th^{s_2}, where $Th^{s_2} > Th^s$; we ignore the the case for $p_k = 1$ for this discussion. Additionally, $RHS_d \geq LHS_d$ for $p_k \geq Th^d$, and $Th^d > Th^{s_2} > Th^s$. Such a condition will result in a three-threshold optimal policy. Therefore, for the three-threshold policy, $u_{d,k}$ will start from 1 at $p_k = 0$, then become $u_{d,k} = 2$ when $p_k \geq Th^s$. Next, $u_{d,k} = 1$ again when $p_k \geq Th^{s_2}$. Finally, when $p_k \geq Th^d$, $u_{d,k} = 3$. Figures 8 and 9 provide the insights of the three-threshold policy by plotting the LHS and RHS of (25) and (25), for a relatively large ρ. The plots are expanded for the higher values of p_k to illustrate the crossing points better. The corresponding three-threshold optimal policy $u^*_{d,k}$ is plotted in Fig. 10.

Since adversarial attacks are rare events, ρ is generally expected to be low, *i.e.*, close to zero, under practical situations. Moreover, the cost of a false alarm event should be much higher than adding watermarking at a given time point for most systems. Therefore, for most practical cases, we can assume $\lambda_f \gg \lambda_e$ and ρ is small. Simulation studies show that the optimal policy will be a two-threshold policy under such practical parameter values. In other words, $LRS_s > RHS_s$ at $p_k \geq Th^s$ for the first time, and $LRS_s = RHS_s$ at $p_k = Th^{s_2} = 1$ for the second time, also, $Th^{s_2} > Th^s$. Moreover, $LHS_d \geq RHS_d$ at $p_k \geq Th_d$ and $Th^s < Th_d \leq Th^{s_2}$. Figures 11 and 12 provide the insights of the two-threshold policy by plotting the LHS and RHS of (25) and (25), for a relatively small λ_e, large λ_f and ρ close to zero. The corresponding two-threshold optimal policy $u^*_{d,k}$ is summarized in (27) and plotted in Fig. 13. The findings from the structural analysis of the optimal policy of this paper are similar to [13].

$$\mathbf{u}^*_{d,k} = \begin{cases} 1, \textit{ i.e.}, \ (s_k = 0, d_k = 0) & p_k < Th^s, \\ 2, \textit{ i.e.}, \ (s_k = 1, d_k = 0) & p_k \geq Th^s, \\ 3, \textit{ i.e.}, \ (s_k = 0, d_k = 1) & p_k \geq Th^d. \end{cases} \tag{27}$$

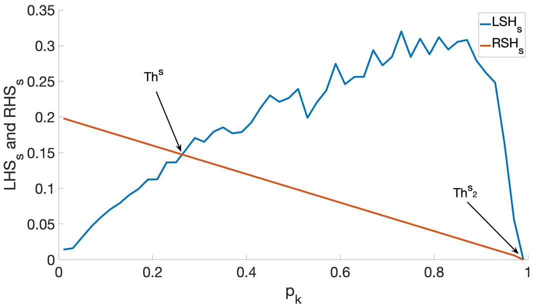

Fig. 11: LHS_s and RHS_s vs. p_k. $\lambda_e = 0.2$, $\lambda_f = 100$, $\rho = 0.001$.

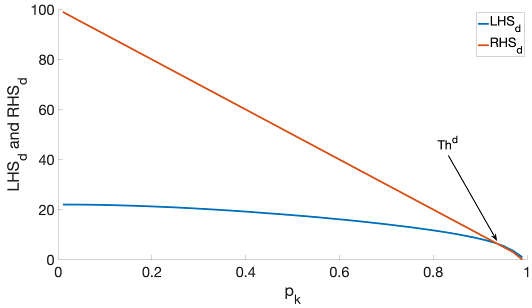

Fig. 12: LHS_d and RHS_d vs. p_k. $\lambda_e = 0.2$, $\lambda_f = 100$, $\rho = 0.001$.

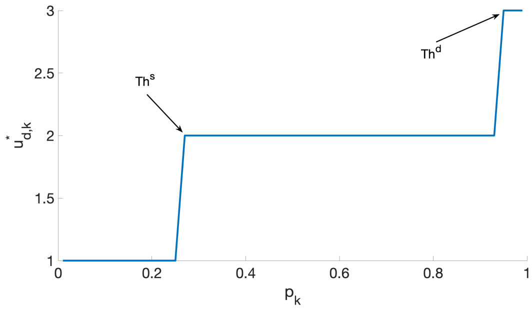

Fig. 13: $u^*_{d,k}$ vs. p_k. $\lambda_e = 0.2$, $\lambda_f = 100$, $\rho = 0.001$.

VI. Numerical Results

In this section, we illustrate the proposed parsimonious watermarking policy by numerical simulations using the MISO system parameters provided in Appendix I. Figure 14 plots the decision variables s_k and d_k, and the posterior probability of attack p_k for a sample trial run under the two-threshold policy shown in Fig. 13. The attack start point is marked in the figure as 'change point'. For the simulation, we have taken watermarking signal variance to be a diagonal matrix with equal signal power, *i.e.*, σ_e^2. We observe that only for a few time instances $s_k = 1$, *i.e.*, watermarking has been added to the control input, before the attack start point. Such parsimonious use of watermarking improves the control cost. On the other hand, after the attack starts, p_k increases rapidly and crosses the threshold Th^s first and then, after a while, the second threshold Th^d. In other words, after the attack starts, watermarking is added to the control inputs most of the time, reducing the ADD compared to the no-watermarking case.

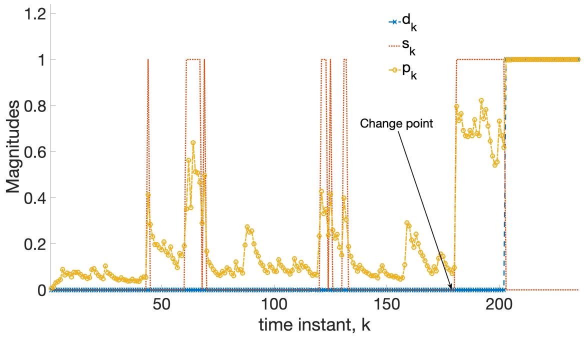

Fig. 14: s_k, d_k and p_k vs. k for a sample trial run. $\lambda_e = 0.2$, $\lambda_f = 100$, $\rho = 0.001$, and $\sigma_e^2 = 1.19$.

Figure 15 compares the ΔLQG for the proposed method and the ΔLQG_a for the always present watermarking case for the same values of Σ_e. Σ_e is assumed to be diagonal, *i.e.*, $\Sigma_e = \text{diag}\left(\sigma_e^2 \ \sigma_e^2\right)$ values. For the proposed method, the thresholds Th^s and Th^d are derived by value iterations for each σ_e^2. We observe a large improvement in the control cost (approx. 99% reduction in ΔLQG) for the proposed method.

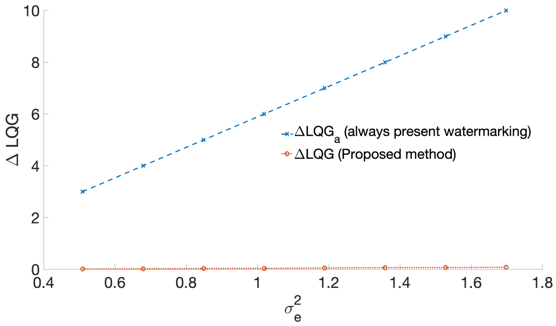

Fig. 15: Comparison between proposed method and the always present watermarking case. ΔLQG vs. σ_e^2 plot. $\lambda_f = 100$, $\lambda_e = 0.2$, and $\rho = 0.001$.

Figure 16 compares the ADD for the proposed method and the ADD for the always present watermarking case at the same watermarking signal power levels. As before, we take $\Sigma_e = \text{diag}\left(\sigma_e^2 \ \sigma_e^2\right)$ for the simulation. Also, the thresholds Th^s and Th^d are derived by value iterations for each σ_e^2 values. We observe an average increase of 35% (approx.) in ADD for the proposed method compared to the always present watermarking case. In other words, ADD increases for the proposed method, but the increase is moderate.

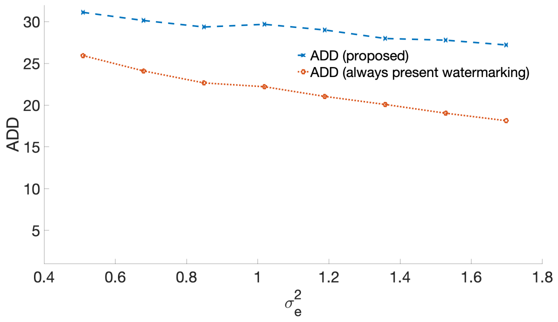

Fig. 16: Comparison between proposed method and the always present watermarking case. ADD vs. σ_e^2 plot. $\lambda_f = 100$, $\lambda_e = 0.2$, and $\rho = 0.001$.

VII. Conclusion

This paper derives a parsimonious watermarking policy to minimize the ADD for fixed upper limits on FAR and ANW. In addition, the limit on ANW reduces the control cost during normal system operations. The optimal policy is derived by formulating the Bellman equation from a constrained optimization problem and solving it by value iterations. The optimal policy found by the value iterations has been studied by numerical simulations and by analyzing the structure of the Bellman equation. We observe that the optimal policy may be a one, two, or three threshold policy, but under a practical range of parameter values, the optimal policy is a two-threshold policy. Furthermore, deriving a threshold base policy from the value iteration solutions reduces the run-time computations when implementing the quickest attack detection mechanism.

Appendix I
System Parameters

The following system parameters are used for simulation study.

$$\mathbf{A} = \begin{bmatrix} 0.75 & 0.2 \\ 0.2 & 1.0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0.9 & 0.5 \\ 0.1 & 1.2 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 1.0 & -1.0 \end{bmatrix}$$
$$\mathbf{Q} = diag\begin{bmatrix} 1 & 1 \end{bmatrix} \qquad \mathbf{R} = 1 \qquad \mathbf{W} = diag\begin{bmatrix} 1 & 2 \end{bmatrix}$$
$$\mathbf{U} = diag\begin{bmatrix} 0.4 & 0.7 \end{bmatrix} \qquad \mathbf{A}_a = 0.5 \qquad \mathbf{Q}_a = 7.5$$

References

[1] R. Alguliyev, Y. Imamverdiyev, and L. Sukhostat, "Cyber-physical systems and their security issues," *Comput. Ind.*, vol. 100, pp. 212–223, 2018.

[2] Y. Mo, S. Weerakkody, and B. Sinopoli, "Physical authentication of control systems: Designing watermarked control inputs to detect counterfeit sensor outputs," *IEEE Control Syst.*, vol. 35, no. 1, pp. 93–109, Jan 2015.

[3] R. Langner, "Stuxnet: Dissecting a cyberwarfare weapon," *IEEE Secur. Priv.*, vol. 9, no. 3, pp. 49–51, 2011.

[4] B. Satchidanandan and P. R. Kumar, "Dynamic Watermarking: Active Defense of Networked Cyber–Physical Systems," *Proc. IEEE*, vol. 105, no. 2, pp. 219–240, Feb 2017.

[5] S. Salimi, S. Dey, and A. Ahlen, "Sequential detection of deception attacks in networked control systems with watermarking," *18th European Control Conference, ECC 2019*, pp. 883–890, 2019.

[6] A. Naha, A. Teixeira, A. Ahlén, and S. Dey, "Quickest detection of deception attacks in networked control systems with physical watermarking," *arXiv preprint arXiv:2101.01466*, 2021.

[7] C. Fang, Y. Qi, P. Cheng, and W. X. Zheng, "Optimal periodic watermarking schedule for replay attack detection in cyber–physical systems," *Automatica*, vol. 112, p. 108698, 2020.

[8] H. S. Sánchez, D. Rotondo, T. Escobet, V. Puig, J. Saludes, and J. Quevedo, "Detection of replay attacks in cyber-physical systems using a frequency-based signature," *J. Franklin Inst.*, vol. 356, no. 5, pp. 2798–2824, 2019.

[9] D. Ye, T. Y. Zhang, and G. Guo, "Stochastic coding detection scheme in cyber-physical systems against replay attack," *Inf. Sci. (Ny).*, vol. 481, pp. 432–444, 2019.

[10] A. Naha, A. Teixeira, A. Ahlen, and S. Dey, "Deception Attack Detection using Reduced Watermarking," *2021 European Control Conference (ECC)*, pp. 74–80, 2021.

[11] A. Naha, A. Teixeira, A. Ahlén, and S. Dey, "Quickest detection of deception attacks on cyber-physical systems with a parsimonious watermarking policy," *arXiv preprint arXiv:2201.09389*, 2022.

[12] A. Tartakovsky, I. Nikiforov, and M. Basseville, *Sequential analysis: Hypothesis testing and changepoint detection*, 2014.

[13] T. Banerjee and V. V. Veeravalli, "Data-Efficient Quickest Change Detection with On-Off Observation Control," *Seq. Anal.*, vol. 31, no. 1, pp. 40–77, 2012.

[14] K. Premkumar and A. Kumar, "Optimal sleep-wake scheduling for quickest intrusion detection using wireless sensor networks," in *IEEE INFOCOM 2008 - The 27th Conference on Computer Communications*, 2008, pp. 1400–1408.

[15] A. G. Tartakovsky and V. V. Veeravalli, "General asymptotic Bayesian theory of quickest change detection," *Theory of Probability and its Applications*, vol. 49, no. 3, pp. 458–497, 2005.

[16] D. P. Bertsekas, *Dynamic programming and optimal control*. Athena scientific Belmont, MA, 1995, vol. 1, no. 2.

[17] F. J. Beutler and K. W. Ross, "Optimal policies for controlled markov chains with a constraint," *Journal of mathematical analysis and applications*, vol. 112, no. 1, pp. 236–252, 1985.