Symmetries and privacy in control over the cloud: uncertainty sets and side knowledge*

Alimzhan Sultangazin[1] and Paulo Tabuada[1*†]

May 21, 2019

Abstract

Control algorithms, like model predictive control, can be computationally expensive and may benefit from being executed over the cloud. This is especially the case for nodes at the edge of a network since they tend to have reduced computational capabilities. However, control over the cloud requires the transmission of sensitive data (e.g., system dynamics, measurements) which undermines privacy of these nodes. When choosing a method to protect the privacy of these data, efficiency must be considered to the same extent as privacy guarantees to ensure adequate control performance. In this paper, we review a transformation-based method for protecting privacy, previously introduced by the authors, and quantify the level of privacy it provides. Moreover, we also consider the case of adversaries with side knowledge and quantify how much privacy is lost as a function of the side knowledge of the adversary.

1 Introduction

1.1 Motivation

With an increasing connectivity of devices there has been a marked growth in the use of cloud-based services, in which a powerful server provides memory and computational capabilities to clients. These ideas have also recently garnered attention in control and manifested in control over the cloud, wherein a controller is placed on the cloud and both measurements and control inputs are sent over a communication channel.

Control over the cloud has several advantages, which include the opportunity to outsource expensive computational tasks to the cloud, easier installation and maintenance [1], and the availability of information from all of the cloud's clients when making control decisions [2]. Several works [2, 3, 4] have shown practical feasibility of model predictive control (MPC) over the cloud.

Notwithstanding the benefits of control over the cloud, the exposure of existing systems to the cloud may lead to security vulnerabilities in a vast variety of applications [5, 6, 7, 8, 9], including control of process plants, traffic infrastructure, and smart meter systems. One of the most basic attacks exploiting these vulnerabilities is eavesdropping. In the context of control over the cloud, eavesdropping involves an adversary listening in to the communication channel to leak valuable information about the model, the controller, and trajectory [10].

**The work of the authors was partially supported by the NSF grants 1740047, 1705135 and by the Army Research Laboratory under Cooperative Agreement W911NF-17-2-0196. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

†[1]Alimzhan Sultangazin and Paulo Tabuada are with Department of Electrical and Computer Engineering, University of California - Los Angeles, USA {asultangazin, tabuada}@ucla.edu

Traditionally, eavesdropping attacks are prevented by means of encryption - the client and the cloud establish a shared key with which they encrypt transmitted messages and decrypt the messages they receive. This technique, however, fails to protect the system if the privacy breach occurs within the cloud. Hence, there is a need for control-over-the cloud methods that do not require the incoming data to be decrypted by the cloud. While approaching this problem, one must surely keep in mind two other important concerns: efficiency and safety. We do not want to achieve privacy at the cost of degradation of control performance either due to delays in the feedback loop or inaccurate control inputs.

1.2 Related work

So far the issue of privacy in control over the cloud has been approached under the frameworks of homomorphic encryption, differential privacy, and algebraic transformations.

Homomorphic encryption techniques allow the cloud to perform the necessary computations on encrypted data without the need to decrypt it [11]. Fully homomorphic encryption (FHE) was considered in [12, 13]. Unfortunately, FHE is inefficient in terms of execution time [11], making it impractical for online optimization. This has led to increased interest in partially homomorphic encryption (PHE), see [1,10,14,15,16,17,18]. While improved, execution time remains a valid concern in these methods [10, 16].

The notion of differential privacy was also recently applied to privacy in control (see [19, 20]). The main idea of these methods is to perturb the response to a data query with appropriate noise [21]. However, to achieve more privacy, the user must sacrifice accuracy (i.e., add more noise), which, in the context of control, degrades the control performance.

Algebraic transformation methods, to which this work relates, have been initially applied in optimization to produce a different, but equivalent optimization problem. Despite the cloud not knowing the original optimization problem, it can provide to the client the optimal solution to an equivalent optimization problem from which the client recovers the optimal solution to the original problem. These methods found applications in control due to their efficiency and guaranteed optimality of the solution [22]. For example, in [23] the authors propose a hybrid transformation-based method to preserve privacy of an MPC controller in networked control systems. In [24], transformation-based methods are used to provide privacy in a specific problem of AC optimal power flow.

1.3 Contributions

While ensuring privacy in control systems poses difficulties due to the dynamic nature of the problem, some features of dynamical systems can be leveraged to solve the problem of privacy. We propose to use isomorphisms and symmetries of the dynamics as a source of transformations so as to keep not only the optimization problems, but also the resulting plant dynamics, equivalent. The benefits of this approach include increased computational efficiency, possibility of computation on encoded data and simplicity of design.

The proposed method was initially introduced in [25]. As opposed to [23], it applies to a more general class of quadratic programs and provides encoding for the state and the output. In comparison to [24], the proposed scheme is also more general since it is applied to a wider class of problems and considers the scenarios when the output is not equal to the state. In [26], the scheme was extended to networked control systems with several agents and a single cloud. In this paper, we address two problems that were not discussed in [25, 26]:

1. we propose a measure of privacy and show how to compute it for the different scenarios discussed in the paper;
2. we quantify how much privacy is lost when an adversary has access to side knowledge.

2 Problem Definition

2.1 Plant dynamics and control objective

We consider discrete-time affine plants, denoted by Σ, and described by:

$$\Sigma: \quad \begin{aligned} \bar{x}_{k+1} &= \bar{A}\bar{x}_k + \bar{B}\bar{u}_k + \bar{c} \\ \bar{y}_k &= \bar{C}\bar{x}_k + \bar{d}, \end{aligned} \tag{2.1}$$

where $\bar{A} \in \mathbb{R}^{n\times n}$, $\bar{B} \in \mathbb{R}^{n\times m}$, $\bar{C} \in \mathbb{R}^{p\times n}$, $\bar{c} \in \mathbb{R}^n$, and $\bar{d} \in \mathbb{R}^p$ describe the dynamics of the system, and $\bar{x} \in \mathbb{R}^n$, $\bar{u} \in \mathbb{R}^m$ and $\bar{y} \in \mathbb{R}^p$ denote the state, input and output of the system, respectively. We assume that system Σ is controllable and observable. We also assume, without loss of generality, that $\ker \bar{B} = 0$ and $\operatorname{Im} \bar{C} = \mathbb{R}^p$, since we can always eliminate linearly independent columns (resp. rows) from $\bar{B}$ (resp. $\bar{C}$).

To simplify notation, we lift every affine map $Ax + c$ to a linear map through the following construction:

$$Ax + c \mapsto \begin{bmatrix} A & c \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ 1 \end{bmatrix}. \tag{2.2}$$

Applying (2.2) to (2.1), we obtain:

$$\begin{aligned} x_{k+1} \triangleq \begin{bmatrix} \bar{x}_{k+1} \\ 1 \end{bmatrix} &= \begin{bmatrix} \bar{A} & \bar{c} \\ 0_{1\times n} & 1 \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ 1 \end{bmatrix} + \begin{bmatrix} \bar{B} \\ 0 \end{bmatrix} u_k \\ &\triangleq Ax_k + Bu_k \\ y_k \triangleq \begin{bmatrix} \bar{y}_k \\ 1 \end{bmatrix} &= \begin{bmatrix} \bar{C} & \bar{d} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \bar{x}_k \\ 1 \end{bmatrix} \triangleq Cx_k. \end{aligned} \tag{2.3}$$

In the remainder of the paper we suppress the inner structure for simplicity. Nevertheless, the reader should remember that we are dealing with affine maps. This is also true for the affine maps we will use to define isomorphisms. We refer to system (2.3) as the triple $\Sigma = (A, B, C)$.

We call a triple $\{x_k, u_k, y_k\}_{k\in\mathbb{N}}$ a trajectory of Σ if it satisfies (2.1) for all $k \in \mathbb{N}$.

Additionally, we define a cost function $J : \mathbb{R}^n \times (\mathbb{R}^m)^{N+1} \to \mathbb{R}$ for $N \in \mathbb{N} \cup \{+\infty\}$ that allows to compare trajectories, thereby formulating different control objectives. We consider quadratic cost functions given by:

$$J(x, u) = \sum_{i=0}^{N} \Delta\eta_i^T M \Delta\eta_i, \tag{2.4}$$

where $\Delta\eta_i = \begin{bmatrix} x_i - x_i^* & u_i - u_i^* \end{bmatrix}^T$, $x = \{x_0, ..., x_N\}$ and $u = \{u_0, ..., u_N\}$. Sequences $x^* = \{x_0^*, ..., x_N^*\}$ and $u^* = \{u_0^*, ..., u_N^*\}$ denote the desired setpoints. We define $M \in \mathbb{R}^{(n+m+1)\times(n+m+1)}$ to be a positive-definite matrix.

In addition to a cost, we also consider control objectives that require certain constraints to be satisfied at all times. These constraints are defined as:

$$D\eta_i \leq 0, \quad i = 0, ..., N, \tag{2.5}$$

where $\eta_i = \begin{bmatrix} x_i & u_i \end{bmatrix}^T$ and $D \in \mathbb{R}^{h\times(n+m+1)}$. Note that, despite appearing to be linear constraints, the constraints above are in fact affine, in view of the construction (2.2).

We also assume that there are $n+1$ linearly independent constraints on the state x_k. This is a valid assumption since real-world systems usually have an operational envelope that is bounded (e.g., maximum velocity or maximum pressure).

2.2 Attack model and privacy objectives

The cloud is treated as a curious but honest adversary: the cloud adheres to the computations prescribed by the agreed-upon protocol, but may seek to extract and leak confidential information by keeping record of all computations and communicated messages.

The interaction between the plant and the cloud is performed in two steps. During handshaking, the plant provides the cloud with a suitably modified version of the plant model, cost, and constraints. In exchange, the cloud agrees to compute the input minimizing the provided cost, subject to the constraints and plant dynamics. During the plant execution, the plant repeatedly sends a suitably modified version of its measurements to the cloud. The cloud computes a new input based on the received measurements and sends it to the plant, where it is suitably modified before being applied to the plant.

In the previous paragraph we purposely used the vague expression "suitably modified". Making this expression concrete requires that we first define the knowledge available to the plant. We consider the following two scenarios.

1. the cloud has no knowledge about the plant;
2. the cloud has no knowledge about the plant, except for knowing what its sensors and actuators are;
3. the cloud has complete knowledge about the plant dynamics, including its sensors and actuators.

These scenarios dictate which modifications can be applied. For more details on the scenarios, refer to [25].

3 Main definitions and summary of previous results

In this section, we summarize the theoretical results from [25] since they provide the context for the main result of this work. The key notions that were discussed in [25] are those of isomorphism and symmetry of control systems.

Let us denote by $\mathcal{S}_{n,m,p}$ the set of all controllable and observable linear control systems with state, input and output dimensions n, m, and p, respectively.

Definition 3.1. An isomorphism of control systems in $\mathcal{S}_{n,m,p}$ is a quadruple $\psi = (P, F, G, S)$ consisting of a change of state coordinates $P : \mathbb{R}^n \to \mathbb{R}^n$, a state feedback $F : \mathbb{R}^n \to \mathbb{R}^m$, a change of coordinates in the input space $G : \mathbb{R}^m \to \mathbb{R}^m$ and a change of coordinates in the output space $S : \mathbb{R}^p \to \mathbb{R}^p$. Transformations P and S are affine invertible maps, F is an affine map, and G is a linear invertible map.

Recall that, to simplify notation, we lift affine maps to linear maps using the transformation (2.2).

Let us also denote the set of isomorphisms of $\mathcal{S}_{n,m,p}$ described in Definition 3.1 as $\mathcal{G}_{n,m,p}$. The set $\mathcal{G}_{n,m,p}$ forms a group under function composition as the group operation[1]. This allows us to define a group action of $\mathcal{G}_{n,m,p}$ on the set of linear control systems $\mathcal{S}_{n,m,p}$.

Definition 3.2. Each element $\psi \in \mathcal{G}_{n,m,p}$ acts on $\Sigma \in \mathcal{S}_{n,m,p}$ to produce $\psi_*\Sigma$ given by:

$$\begin{aligned} \psi_*\Sigma &= (P, F, G, S)_*(A, B, C) \\ &= (P(A - BG^{-1}F)P^{-1}, PBG^{-1}, SCP^{-1}) \\ &\triangleq (\tilde{A}, \tilde{B}, \tilde{C}). \end{aligned} \tag{3.1}$$

[1] A composition of two isomorphisms is given by $\psi_2 \circ \psi_1 = (P_2P_1, G_2F_1 + F_2P_1, G_2G_1, S_2S_1)$, the identity is $\psi_e = (I, 0, I, I)$ and the inverse is given by $\psi^{-1} = (P^{-1}, -G^{-1}FP, G^{-1}, S^{-1})$.

This map is called an isomorphism action.

An isomorphism maps the state x_k, input u_k and output y_k of system Σ to the state $\tilde{x}_k$, input $\tilde{u}_k$, and output $\tilde{y}_k$ of system $\psi_*\Sigma$ as follows:

$$\tilde{x}_k = Px_k \qquad \tilde{u}_k = Fx_k + Gu_k \qquad \tilde{y}_k = Sy_k. \tag{3.2}$$

Similarly, an isomorphism induces transformation on the control objectives, i.e., the cost and constraints. The effect of ψ on η_k can be represented by:

$$\tilde{\eta}_k = \begin{bmatrix} \tilde{x}_k \\ \tilde{u}_k \end{bmatrix} = \begin{bmatrix} P & 0 \\ F & G \end{bmatrix} \begin{bmatrix} x_k \\ u_k \end{bmatrix} \triangleq L\eta_k. \tag{3.3}$$

Therefore, the cost function J can be expressed as a function of new states $\tilde{x}$ and inputs $\tilde{u}$ as follows:

$$\tilde{J}(\tilde{x}, \tilde{u}) = \psi_* J\,(x, u) = \sum_{k=0}^{N} \Delta\tilde{\eta}_i^T \tilde{M} \Delta\tilde{\eta}_i, \tag{3.4}$$

where $\tilde{M} = L^{-T} M L^{-1}$. Applying the isomorphism action to the constraints in (2.5) yields:

$$\tilde{D}\tilde{\eta}_i \leq 0, \tag{3.5}$$

where $\tilde{D} = \psi_* D = DL^{-1}$.

Let us now define $\bar{\mathcal{S}}_{n,m,p}$ to be a set of all quadruples $\left(\Sigma, J, D, \{x_k, y_k, u_k\}_{k\in\mathbb{N}}\right)$ such that $\{x_k, y_k, u_k\}_{k\in\mathbb{N}}$ is a trajectory of Σ minimizing cost function J under constraints D. Similarly to $\mathcal{S}_{n,m,p}$, we can define a group action of $\mathcal{G}_{n,m,p}$ on $\bar{\mathcal{S}}_{n,m,p}$ in view of the previous discussion.

Therefore, we can use the action of $\mathcal{G}_{n,m,p}$ to define the equivalence relation on $\bar{\mathcal{S}}_{n,m,p}$.

Definition 3.3. Let $\Omega = (\Sigma, J, D, \{x_k, u_k, y_k\}_{k\in\mathbb{N}})$ and $\tilde{\Omega} = (\tilde{\Sigma}, \tilde{J}, \tilde{D}, \{\tilde{x}_k, \tilde{u}_k, \tilde{y}_k\}_{k\in\mathbb{N}})$ be elements of $\bar{\mathcal{S}}_{n,m,p}$. The equivalence relation $\sim_{\mathcal{G}}$ on $\bar{\mathcal{S}}_{n,m,p}$ denoted by:

$$\Omega \sim_{\mathcal{G}} \hat{\Omega}, \tag{3.6}$$

is defined by the existence of $\psi \in \mathcal{G}_{n,m,p}$ such that:

$$\tilde{\Omega} = \psi_*\Omega, \tag{3.7}$$

i.e., $\tilde{\Sigma} = \psi_*\Sigma$, $\tilde{J} = \psi_* J$, $\tilde{D} = \psi_* D$, and $\{\tilde{x}_k, \tilde{u}_k, \tilde{y}_k\}_{k\in\mathbb{N}}$ is given in terms of $\{x_k, u_k, y_k\}_{k\in\mathbb{N}}$ as in (3.2).

The equivalence relation $\sim_{\mathcal{G}}$, in turn, defines equivalence classes in $\bar{\mathcal{S}}_{n,m,p}$. The equivalence class of $\Omega \in \bar{\mathcal{S}}_{n,m,p}$ via $\sim_{\mathcal{G}}$ is the set:

$$\begin{aligned} [\Omega] &\triangleq \{\Omega' \in \bar{\mathcal{S}}_{n,m,p} | \exists \psi \in \mathcal{G}_{n,m,p} \text{ such that } \Omega' = \psi_*\Omega\} \\ &= \{\psi_*\Omega | \psi \in \mathcal{G}_{n,m,p}\}. \end{aligned} \tag{3.8}$$

For a given system Σ, there is also a special subgroup in $\mathcal{G}_{n,m,p}$ called the subgroup of symmetries.

Definition 3.4. Let Σ be a linear control system. An isomorphism ψ of Σ is said to be a symmetry of Σ if $\psi_*\Sigma = \Sigma$. The subgroup of symmetries of Σ is denoted here by $\mathcal{K}_{n,m,p}(\Sigma)$.

Algorithm 1 (Plant $\Longleftrightarrow$ Cloud)

Input: Plant (P): Σ, J, D, y_k, $\tilde{u}_k$;
Cloud (C): $\tilde{y}_k$, $\tilde{\Sigma}$, $\tilde{J}$, $\tilde{D}$
Output: P: $\tilde{\Sigma}$, $\tilde{J}$, $\tilde{D}$, $\tilde{y}_k$;
C: $\tilde{u}_k$

Phase 1: Handshaking

1: P: Select an isomorphism ψ;
2: P: Encode $\tilde{\Sigma} = \psi_* \Sigma$, $\tilde{J} = \psi_* J$ and $\tilde{D} = \psi_* D$;
3: P: Output $\tilde{\Sigma}$, $\tilde{J}$, and $\tilde{D}$ to the cloud;

Phase 2: Execution

4: P: Encode measurements as $\tilde{y}_k = S y_k$ and send to the cloud;
5: C: Use $\tilde{y}_k$ to estimate the state $\tilde{x}_k$ and compute the input $\tilde{u}_k$ minimizing $\tilde{J}$ subject to the constraints $\tilde{D}$ and the dynamics $\tilde{\Sigma}$;
6: C: Send $\tilde{u}_k$ to the plant;
7: P: Use ψ to decode $\tilde{u}_k$ and produce u_k using (3.2).

In [25], we have proposed to use Algorithm 1 to preserve privacy of information communicated to the cloud. We have shown that, by using this scheme, information communicated to the cloud remains consistent (i.e., after the modification, the resulting trajectory remains a valid trajectory of the modified dynamics). Moreover, we have proven that the client is able to perfectly reconstruct the desired input (i.e., the optimal solution of the original optimization problem) from the cloud's input .

The main reason for using isomorphisms is that the cloud is unable to distinguish between isomorphic systems and, thus, the information is kept private from the cloud. We now formalize the notion of indistinguishability.

Definition 3.5. A protocol renders two quadruples Ω and $\hat{\Omega}$ indistinguishable by the cloud if the exchanged messages, when using the protocol between the cloud and plant Ω, and the exchanged messages, when using the protocol between the cloud and $\hat{\Omega}$, can be made the same.

Theorem 3.6 (Theorem III.6 in [25])**.** *Algorithm 1 renders isomorphic systems* $\Omega = (\Sigma, J, D, \{x_k, u_k, y_k\}_{k\in\mathbb{N}})$ *and* $\tilde{\Omega} = (\tilde{\Sigma}, \tilde{J}, \tilde{D}, \{\tilde{x}_k, \tilde{u}_k, \tilde{y}_k\}_{k\in\mathbb{N}})$ *indistinguishable by the cloud.*

The result described in Theorem 3.6 states that the cloud cannot differentiate between any two plants, costs, constraints or trajectories contained in the same equivalence class of the $\sim_{\mathcal{G}}$-equivalence relation, thereby protecting the privacy of the system. In the next section, we quantify the amount of privacy provided by Theorem 3.6.

4 Quantifying privacy

Privacy is created by preventing the cloud from knowing which quadruple Ω in its equivalence class $[\Omega]$ it is interacting with. Clearly, the larger the equivalence class, the more privacy is ensured. Since each equivalence class has infinitely many elements, cardinality cannot be used as a measure of privacy. In this section, we show that each equivalence class is a smooth manifold and we quantify privacy using the dimension of this manifold.

In order to do that, we must first consider the stabilizer subgroup of $\mathcal{G}_{n,m,p}$ for any $\Omega \in \bar{S}_{n,m,p}$, which we denote by $\mathcal{K}_{n,m,p}(\Omega)$. Recall that the stabilizer subgroup is defined by:

$$\mathcal{K}_{n,m,p}(\Omega) = \{\psi \in \mathcal{G}_{n,m,p} | \psi_* \Omega = \Omega\}. \tag{4.1}$$

We note that $\mathcal{K}_{n,m,p}(\Omega) \subset \mathcal{K}_{n,m,p}(\Sigma)$ since the stabilizer subgroup must preserve dynamics. To gain insight about the stabilizer subgroup $\mathcal{K}_{n,m,p}(\Omega)$, let us consider the subgroup of symmetries $\mathcal{K}_{n,m,p}(\Sigma)$ in more detail.

In [27], Respondek gives a characterization of symmetries of linear systems (A, B). This result can be interpreted to state that the symmetry ψ is uniquely determined by its P. We use this result to characterize $\mathcal{K}_{n,m,p}(A, B)$.

Proposition 4.1 ([27,28]). *Let (A,B) be a linear system. Then, dim* $\mathcal{K}_{n,m,p}(A, B) = m(n+1) - \sum_{i=2}^{m} r_{i-1} r_i$, *where:*

$$\begin{aligned} &r_1 = \textit{rank } B, \\ &r_i = \textit{rank } S_{i-1}(A, B) - \textit{rank } S_{i-2}(A, B), \;\; i = 2, ..., m, \\ &S_j(A, B) = \begin{bmatrix} B & AB & ... & A^j B \end{bmatrix}, \;\; j = 1, ..., m-1. \end{aligned}$$

This result can be used to estimate the dimension of $\mathcal{K}_{n,m,p}(\Sigma)$. To go from keeping (A, B) invariant to keeping (A, B, C) invariant, we need to find S that helps keep C invariant. In other words, assuming that we have found (P, F, G) that preserves (A, B), we need to additionally find S such that $C = SCP^{-1}$. Since we assume C to have linearly independent rows, this equation has at most one solution. This gives an upper bound on the number of dimensions of the subgroup of symmetries dim $\mathcal{K}_{n,m,p}(\Sigma) \leq m(n+1) - \sum_{i=2}^{k_1} r_{i-1} r_i$. In future work, we plan to further investigate the symmetry subgroup of Σ in order to find what exactly dim $\mathcal{K}_{n,m,p}(\Sigma)$ is equal to.

To find a transformation P that keeps Ω invariant, let us use the assumption that we have $n+1$ linearly independent constraints on the state x_k expressed by the constraint matrix D. Therefore, any $\psi \in \mathcal{K}_{n,m,p}(\Omega)$ must satisfy:

$$\begin{aligned} DL^{-1} = D &\iff DL = D \\ &\iff \begin{bmatrix} D_{11} & 0 \\ D_{21} & D_{22} \end{bmatrix} \begin{bmatrix} P & 0 \\ F & G \end{bmatrix} = \begin{bmatrix} D_{11} & 0 \\ D_{21} & D_{22} \end{bmatrix} \\ &\implies D_{11} P = D_{11}. \end{aligned}$$

Given that $D_{11} \in \mathbb{R}^{h_1 \times (n+1)}$ is injective, the last equality is satisfied if and only if $P = I$. Since P uniquely defines F, G and S, we also have that the only isomorphism that keeps (A, B, C, D_{11}) invariant is $\psi = \psi_e = (I, 0, I, I)$. Therefore, the only element of $\mathcal{K}_{n,m,p}(\Omega)$ is $\phi_e = (I, 0, I, I)$.

Let us now define a map:

$$\begin{aligned} f_\Omega : \mathcal{G}_{n,m,p} &\to \bar{\mathcal{S}}_{n,m,p} \\ \psi &\mapsto \psi_* \Omega. \end{aligned} \tag{4.2}$$

Due to the fact that $\mathcal{K}_{n,m,p}(\Omega)$ is trivial, we can easily show that f_Ω is injective.

The result of the discussion above can be formalized in the following statement.

Lemma 4.2. *Let* $\Omega = (\Sigma, J, D, \{x_k, u_k, y_k\}_{k \in \mathbb{N}})$ *be an arbitrary system in* $\bar{\mathcal{S}}_{n,m,p}$ *with* D *such that* D_{11}, *the constraint matrix on the state* x_k, *has rank* $n+1$. *Then,* $f_\Omega : \mathcal{G}_{n,m,p} \to \bar{\mathcal{S}}_{n,m,p}$, *mapping* ψ *to* $\psi_* \Omega$, *is injective.*

To facilitate further results, let us show that $\bar{\mathcal{S}}_{n,m,p}$ is a smooth manifold and $\mathcal{G}_{n,m,p}$ is a Lie group.

Lemma 4.3. *Let* $\bar{\mathcal{S}}_{n,m,p}$ *denote the set of controllable and observable systems along with costs and constraints. Then,* $\bar{\mathcal{S}}_{n,m,p}$ *is a smooth manifold.*

Proof. We can see that $\bar{\mathcal{S}}_{n,m,p}$ is, in fact, a Cartesian product of $\mathcal{S}_{n,m,p}$, the set of cost functions $\mathcal{M}^{++}(m+n+1, \mathbb{R})$, expressed by positive-definite matrices, set of constraints on states $\mathcal{M}_{n+1}(h_1 \times (n+1), \mathbb{R})$, expressed by the set of full-rank matrices, where $h_1 \geq n+1$, and set of joint constraints on states and inputs $\mathcal{M}_d(h_2 \times (m+n+1), \mathbb{R})$, expressed by the set of full-rank matrices, where $d = \min(h_2, m+n+1)$. It is known that the

product space is a smooth manifold if its constituents are smooth manifolds [29, p. 21]. It remains to show that these constinuents are indeed smooth manifolds.

Let us construct a map:

$$\begin{aligned} f_S :&\mathbb{R}^{n\times(n+1)} \times \mathbb{R}^{n\times m} \times \mathbb{R}^{p\times(n+1)} \to \mathbb{R}^2 \\ &(A, B, C) \mapsto (\det \mathcal{C}, \det \mathcal{O}), \end{aligned} \tag{4.3}$$

where $\mathcal{C}$ and $\mathcal{O}$ are the controllability and observability matrices of the dynamics (A, B, C). It can be seen that $\mathcal{S}_{n,m,p} = f_S^{-1}(\mathbb{R} \setminus (0, 0))$. Function f_S is continuous since each of its elements is defined as a polynomial function of elements of (A, B, C). Since for continuous functions the preimage of every open set is an open set, we have that $\mathcal{S}_{n,m,p}$ is an open subset of the domain of f_S. Since the domain of f_S is a smooth manifold, $\mathcal{S}_{n,m,p}$ is a smooth manifold of dimension $n(n + 1) + nm + p(n + 1)$.

The set of positive-definite matrices $\mathcal{M}^{++}(m + n + 1, \mathbb{R})$ is shown to be a smooth embedded submanifold of $\mathbb{R}^{(m+n+1)\times(m+n+1)}$ of dimension $(m + n + 1)(m + n + 2)/2$ in [30].

Both the set of full-rank matrices $\mathcal{M}_{n+1}(h_1 \times (n + 1), \mathbb{R})$ and $\mathcal{M}_d(h_2 \times (m + n + 1), \mathbb{R})$ are smooth manifolds of dimension $h_1(n + 1)$ and $h_2(m + n + 1)$, respectively [29, p. 19]. □

Lemma 4.4. *Let $\mathcal{G}_{n,m,p}$ be the isomorphism group. Then, $\mathcal{G}_{n,m,p}$ is a Lie group of dimension $n(n + 1) + m(n + 1) + m^2 + p(p + 1)$ acting smoothly, freely, and properly on $\bar{\mathcal{S}}_{n,m,p}$.*

Proof. It was previously established that $\mathcal{G}_{n,m,p}$ is a group. It is a Lie group because it is a Cartesian product of smooth manifolds (i.e., general linear groups and vector spaces of various dimensions) and its multiplication and inversion maps are smooth. Moreover, since the dimension of a product of smooth manifolds is equal to the sum of the factors' dimensions, the dimension of $\mathcal{G}_{n,m,p}$ is $n(n + 1) + m(n + 1) + m^2 + p(p + 1)$ [29, p. 21]. The group $\mathcal{G}_{n,m,p}$ acts smoothly on $\bar{\mathcal{S}}_{n,m,p}$ since its action involves matrix multiplication and matrix inversion: the former results in every element being a polynomial function of the elements of the product, while the latter is smooth by Cramer's rule [29].

The action of $\mathcal{G}_{n,m,p}$ is free since the only element that fixes any element $\Omega \in \bar{\mathcal{S}}_{n,m,p}$ is the identity $\phi_e = (I, 0, I, I)$ (see Lemma 4.2).

To show that $\mathcal{G}_{n,m,p}$ acts properly, we need to show that the map:

$$\begin{aligned} g : \mathcal{G}_{n,m,p} \times \bar{\mathcal{S}}_{n,m,p} &\to \bar{\mathcal{S}}_{n,m,p} \times \bar{\mathcal{S}}_{n,m,p} \\ (\psi, \Omega) &\mapsto (\psi_* \Omega, \Omega) \end{aligned} \tag{4.4}$$

is a proper map.

From [29, p. 611], g is a proper map if its codomain is Hausdorff and it has a continuous left inverse. The codomain of g is a product of smooth manifolds and, therefore, is Hausdorff. We must show that there exists a continuous left inverse of g, denoted by t, that, given Ω and $\tilde{\Omega}$, produces an isomorphism ψ such that $\tilde{\Omega} = \psi_* \Omega$, i.e.:

$$\begin{aligned} t : \bar{\mathcal{S}}_{n,m,p} \times \bar{\mathcal{S}}_{n,m,p} &\to \mathcal{G}_{n,m,p} \times \bar{\mathcal{S}}_{n,m,p} \\ (\tilde{\Omega}, \Omega) &\mapsto (\psi, \Omega). \end{aligned} \tag{4.5}$$

Given Ω and $\tilde{\Omega}$, the encoding isomorphism ψ can be calculated as:

$$\begin{aligned} P &= (\tilde{D}_{11}^T \tilde{D}_{11})^{-1} \tilde{D}_{11}^T D_{11} \\ F &= (B^T P^{-T} P^{-1} B)^{-1} B^T P^{-T} (A - P^{-1} \tilde{A} P) \\ G &= ((B^T B)^{-1} B^T P^{-1} \tilde{B})^{-1} \\ S &= \tilde{C} P C^T (C C^T)^{-1}, \end{aligned} \tag{4.6}$$

because $\tilde{D}_{11}$, B, and $P^{-1}\tilde{B}$ are injective and C is surjective. The solutions for P, F, G, and S are given by continuous functions of (A, B, C, D_{11}) and $(\tilde{A}, \tilde{B}, \tilde{C}, \tilde{D}_{11})$ and, therefore, t is continuous. □

Consider the first scenario from Section 2, in which the cloud does not know anything about the system. In this scenario, the plant encodes Ω using an isomorphism $\psi = (P, F, G, S)$ that can be regarded as a private key used to encode and decode the information exchanged with the cloud. This isomorphism ψ can be picked from $\mathcal{G}_{n,m,p}$, the group of all isomorphisms. The results of aforementioned lemmas allow us to prove the following result.

Proposition 4.5. *Let $\bar{\mathcal{S}}_{n,m,p}$ be a set of controllable and observable systems along with costs and constraints and let the group of isomorphisms $\mathcal{G}_{n,m,p}$ act smoothly, freely and properly on this set. Then, for any $\Omega \in \bar{\mathcal{S}}_{n,m,p}$, the equivalence class defined by the action of $\mathcal{G}_{n,m,p}$ on $\bar{\mathcal{S}}_{n,m,p}$, denoted by $[\Omega]_{\mathcal{G}}$, is smooth manifold of dimension dim $\mathcal{G}_{n,m,p} = n(n+1) + m(n+1) + m^2 + p(p+1)$*

Proof. From Lemmas 4.3 and 4.4, we know that $\bar{\mathcal{S}}_{n,m,p}$ is a smooth manifold, and $\mathcal{G}_{n,m,p}$ is a Lie group acting smoothly, freely and properly on $\bar{\mathcal{S}}_{n,m,p}$. Hence, by quotient manifold theorem [29, p. 544], we have that orbit space $\bar{\mathcal{S}}_{n,m,p}/\mathcal{G}_{n,m,p}$ is a smooth manifold of dimension equal to dim $\bar{\mathcal{S}}_{n,m,p}$ − dim $\mathcal{G}_{n,m,p}$ and the quotient map $\pi : \bar{\mathcal{S}}_{n,m,p} \to \bar{\mathcal{S}}_{n,m,p}/\mathcal{G}_{n,m,p}$ is a smooth submersion. Using submersion level set theorem [29, p. 105], we can further show that, since both $\bar{\mathcal{S}}_{n,m,p}$ and $\bar{\mathcal{S}}_{n,m,p}/\mathcal{G}_{n,m,p}$ are smooth manifolds and π is a smooth submersion, the orbit $[\Omega] = \pi^{-1}(\omega)$, where $\omega \in \bar{\mathcal{S}}_{n,m,p}/\mathcal{G}_{n,m,p}$ is a representative element of this orbit, is a properly embedded submanifold whose dimension is equal to dim $\mathcal{G}_{n,m,p}$. □

This proposition is used to quantify privacy of other scenarios presented in Section 2.

Consider the scenario where the cloud does not know the dynamics but knows which sensors and actuators will be used. We can no longer use an arbitrary isomorphism since it could lead to inputs and outputs that are inconsistent with existing sensors and actuators. This inconsistency would signal the cloud that the plant is being dishonest about its measurements and provide the cloud with an opportunity to exploit this fact to gather additional knowledge. Therefore, we need to restrict the group of isomorphisms used for encoding. These isomorphisms are given by any composition of $\psi_1 = (P, 0, I, I)$ for any $P \in GL(n, \mathbb{R})$ and $\psi_2 \in \mathcal{K}_{n,m,p}(\Sigma)$. It can be shown that this set of isomorphisms forms a subgroup that we denote by $\mathcal{H}_{n,m,p}(\Sigma) \subset \mathcal{G}_{n,m,p}$.

Corollary 4.6. *Consider the group of isomorphisms $\mathcal{H}_{n,m,p}(\Sigma)$ to be acting on $\bar{\mathcal{S}}_{n,m,p}$. Then, the equivalence classes defined by the action of $\mathcal{H}_{n,m,p}(\Sigma)$ on $\bar{\mathcal{S}}_{n,m,p}$ are smooth manifolds of dimension dim $\mathcal{H}_{n,m,p}(\Sigma)$, where*

$$n(n+1) \leq \textit{dim } \mathcal{H}_{n,m,p}(\Sigma) \leq n(n+1) + m(n+1) - \sum_{i=2}^{k_1} r_{i-1} r_i$$

Proof. It can be shown that $\mathcal{H}_{n,m,p}(\Sigma)$ is a Lie subgroup of $\mathcal{G}_{n,m,p}$. This subgroup $\mathcal{H}_{n,m,p}(\Sigma)$ can be thought of as a product manifold of $\mathcal{K}_{n,m,p}(\Sigma)$ and a space of invertible affine maps. Since the dimension of a product manifold is a sum of its factors' dimensions, we have the bounds on dim $\mathcal{H}_{n,m,p}(\Sigma)$ from the corollary's statement (see discussion after Proposition 4.1 for dimension of $\mathcal{K}_{n,m,p}(\Sigma)$). The result follows by applying Proposition 4.5 for $\mathcal{H}_{n,m,p}(\Sigma)$. □

Finally, in the scenario where the cloud possesses the complete knowledge of dynamics, only the isomorphisms from the symmetry subgroup $\psi \in \mathcal{K}_{n,m,p}(\Sigma)$ can be used.

Corollary 4.7. *Consider the group of isomorphisms $\mathcal{K}_{n,m,p}(\Sigma)$ to be acting on $\bar{\mathcal{S}}_{n,m,p}$. Then, the equivalence classes defined by the action of $\mathcal{K}_{n,m,p}(\Sigma)$ on $\bar{\mathcal{S}}_{n,m,p}$ are smooth manifolds of dimension*

$$\textit{dim } \mathcal{K}_{n,m,p}(\Sigma) \leq m(n+1) - \sum_{i=2}^{k_1} r_{i-1} r_i.$$

Proof. From Proposition 4.1, it can be shown that $\mathcal{K}_{n,m,p}(\Sigma)$ is a Lie subgroup of $\mathcal{G}_{n,m,p}$ of dimension less or equal to $m(n+1) - \sum_{i=2}^{k_1} r_{i-1}r_i$. The result again follows by applying Proposition 4.5 for $\mathcal{K}_{n,m,p}(\Sigma)$. □

In future work, we plan to give a better quantification of the dimension of $\mathcal{K}_{n,m,p}(\Sigma)$ (preferably, with an equality), which will directly affect statements of Corollaries 4.6 and 4.7 by changing the estimates for the dimension of the equivalence classes.

5 Side knowledge

There may be some instances in which the cloud has partial information about either a system or the encoding isomorphism. The cloud may have learned those through some external channels or through some prior knowledge about the system.

Recall that by Lemma 4.4, $\mathcal{G}_{n,m,p}$ is a Lie group of dimension $n(n+1) + m(n+1) + m^2 + p(p+1)$. Suppose the cloud has partial knowledge about the encoding isomorphism. We shall represent the partial knowledge available to the cloud as a projection from $\mathcal{G}_{n,m,p}$ onto a k-dimensional vector space. Let us define $\rho : \mathcal{G}_{n,m,p} \to \mathbb{R}^k$ to be a surjective map of constant rank k, providing side knowledge about the encoding isomorphism. Then, we can say that the cloud knows some vector $l \in \mathbb{R}^k$, where:

$$l = \rho(P, F, G, S). \tag{5.1}$$

Note that this map is not known to us, and the results that follow do not require the knowledge of this map.

Side knowledge does not change the result of Theorem 3.6, however the privacy guaranteed by the scheme changes. It is obvious that the size of the uncertainty set constructed by isomorphisms that satisfy (5.1) is smaller than it is with no restrictions. Moreover, the uncertainty set is no longer neither an orbit nor an equivalence class because the preimage of ρ does not necessarily have a group structure.

Let us show that the object defined by (5.1) on $\mathcal{G}_{n,m,p}$ is still a manifold.

Lemma 5.1. *Let $\mathcal{G}_{n,m,p}$ be the group of all isomorphisms, $\rho : \mathcal{G}_{n,m,p} \to \mathbb{R}^k$ be a surjective map of constant rank k and assume the cloud knows that $l = \rho(P, F, G, S)$. Then, $\rho^{-1}(l)$, representing the possible encoding isomorphisms used by the client, is a smooth manifold of dimension $n(n+1) + m(n+1) + m^2 + p(p+1) - k$.*

Proof. By the global rank theorem [29, p. 83], since ρ is a surjective map of constant rank k, it is a smooth submersion. From submersion level set theorem [29, p. 105], since both $\mathcal{G}_{n,m,p}$ and $\mathbb{R}^k$ are smooth manifolds and ρ is a smooth submersion, we have that $\rho^{-1}(l)$ is a properly embedded submanifold of dimension $\dim \mathcal{G}_{n,m,p} - \dim \mathbb{R}^k = n(n+1) + m(n+1) + m^2 + p(p+1) - k$. □

Let us now consider the map f_Ω defined earlier in (4.2). It was shown in Lemma 4.2 that f_Ω is injective. The image of $f_\Omega(\rho^{-1}(l))$ constitutes the uncertainty set, between the elements of which the cloud is not be able to distinguish. Therefore, the main result of this section is to find the dimension of $f_\Omega(\rho^{-1}(l))$.

Proposition 5.2. *Let $\rho^{-1}(l) \subset \mathcal{G}_{n,m,p}$ be a set of isomorphisms that comply with side knowledge and let $f_\Omega : \mathcal{G}_{n,m,p} \to \bar{\mathcal{S}}_{n,m,p}$, mapping ψ to $\psi_*\Omega$, be the orbit map for $\Omega \in \bar{\mathcal{S}}_{n,m,p}$. Then, $\mathcal{U} = f_\Omega(\rho^{-1}(l))$ is a smooth manifold of dimension $n(n+1) + m(n+1) + m^2 + p(p+1) - k$.*

Proof. By the property of the orbit map [29, p. 166], for each Ω, the orbit map f_Ω is smooth and has constant rank. Since f_Ω is also injective, we have, by Global Rank Theorem, that it is a smooth immersion [29, p. 83]. Moreover, by following an argument similar to the one used in the proof of Lemma 4.4, one can show that f_Ω is a proper map. Again, since we know that $\bar{\mathcal{S}}_{n,m,p}$ is Hausdorff, the only thing left to show is that f_Ω has a continuous inverse. As it was previously shown in Lemma 4.4, the encoding isomorphism ψ is given

by a continuous function of Ω and $\tilde{\Omega}$, which shows that f_Ω has a continuous inverse. Using Proposition 4.22 from [29] and the fact that f_Ω is a proper map, we can show that f_Ω is a smooth embedding. By Proposition 5.2 from [29], $\mathcal{U} = f_\Omega(\rho^{-1}(l))$ is an embedded submanifold of $\bar{\mathcal{S}}_{n,m,p}$ diffeomorphic to $\rho^{-1}(l)$ and, hence, has the same dimension (refer to Lemma 5.1). □

This result shows that the proposed scheme degrades gracefully with side knowledge i.e., side knowledge allows the cloud to reduce the dimension of the uncertainty set only by the amount of side knowledge and not more. Moreover, this result can be generalized for other scenarios considered in Section 4 using similar proofs.

6 Conclusion

In this paper, we have extended the results of the transformation-based privacy algorithm we introduced in [25]. The proposed algorithm has benefits over existing solutions due to its computational efficiency at the client, conceptual simplicity and connection to the properties of dynamical systems. We have, for the first time, provided a criterion for measuring the amount of privacy provided by the proposed algorithm. Moreover, we have considered the implications of the adversary having side channels, other than its direct communication with the client, from which it is able to learn information about the system. In the future, we want to give a better quantification of the privacy in scenarios, where the cloud has some knowledge about plant dynamics, by studying the dimension of $\mathcal{K}_{n,m,p}(\Sigma)$. Furthermore, we wish to determine how this algorithm performs in practice - in particular, we would like to see how privacy is affected if we only have a finite number of keys (i.e., $\mathcal{G}_{n,m,p}$ is no longer infinite) because this accurately models what happens in real computer-based systems.

References

[1] Y. Lin, F. Farokhi, I. Shames, and D. Nezic, "Secure control of nonlinear systems using semi-homomorphic encryption," in *the 57th IEEE Conference on Decision and Control*, 2018, pp. 5002–5007.

[2] B. Hoh, T. Iwuchukwu, Q. Jacobson, D. Work, A. M. Bayen, R. Herring, J. C. Herrera, M. Gruteser, M. Annavaram, and J. Ban, "Enhancing privacy and accuracy in probe vehicle-based traffic monitoring via virtual trip lines," *IEEE Transactions on Mobile Computing*, vol. 11, no. 5, pp. 849–864, May 2012.

[3] A. Vick, J. Guhl, and J. Kruger, "Model predictive control as a service - concept and architecture for use in cloud-based robot control," in *the 2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, Aug 2016, pp. 607–612.

[4] T. Hegazy and M. Hefeeda, "Industrial automation as a cloud service," *IEEE Transactions on Parallel and Distributed Systems*, vol. 26, no. 10, pp. 2750–2763, Oct 2015.

[5] A. Burg, A. Chattopadhyay, and K. Y. Lam, "Wireless communication and security issues for cyber-physical systems and the internet-of-things," *Proceedings of the IEEE*, vol. 106, no. 1, pp. 38–60, Jan 2018.

[6] S. Checkoway, D. McCoy, B. Kantor, D. Anderson, H. Shacham, S. Savage, K. Koscher, A. Czeskis, F. Roesner, and T. Kohno, "Comprehensive experimental analyses of automotive attack surfaces," in *Proceedings of the 20th USENIX Conference on Security*, ser. SEC'11, 2011, pp. 6–6.

[7] D. Gollmann, P. Gurikov, A. Isakov, M. Krotofil, J. Larsen, and A. Winnicki, "Cyber-physical systems security: Experimental analysis of a vinyl acetate monomer plant," in *the 1st ACM Workshop on Cyber-Physical System Security*, 2015, pp. 1–12.

[8] F. M. Cleveland, "Cyber security issues for advanced metering infrasttructure (ami)," in *2008 IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, July 2008, pp. 1–5.

[9] B. Ghena, W. Beyer, A. Hillaker, J. Pevarnek, and J. A. Halderman, "Green lights forever: Analyzing the security of traffic infrastructure," in *Proceedings of the 8th USENIX Conference on Offensive Technologies*, 2014, pp. 7–7.

[10] F. Farokhi, I. Shames, and N. Batterham, "Secure and private control using semi-homomorphic encryption," *Control Engineering Practice*, vol. 67, pp. 13–20, Oct 2017.

[11] F. Armknecht, C. Boyd, C. Carr, K. Gjosteen, A. Jaeschke, C. A. Reuter, and M. Strand, "A guide to fully homomorphic encryption," *IACR Cryptology ePrint Archive*, vol. 2015, p. 1192, 2015.

[12] K. Kogiso and T. Fujita, "Cyber-security enhancement of networked control systems using homomorphic encryption," in *2015 54th IEEE Conference on Decision and Control (CDC)*, 2015, pp. 6836–6843.

[13] T. Fujita, K. Kogiso, K. Sawada, and S. Shin, "Security enhancements of networked control systems using rsa public-key cryptosystem," in *2015 10th Asian Control Conference (ASCC)*, May 2015, pp. 1–6.

[14] F. Farokhi, I. Shames, and N. Batterham, "Secure and private cloud-based control using semi-homomorphic encryption," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 163 – 168, 2016, 6th IFAC Workshop on Distributed Estimation and Control in Networked Systems.

[15] Y. Shoukry, K. Gatsis, A. Alanwar, G. J. Pappas, S. A. Seshia, M. Srivastava, and P. Tabuada, "Privacy-aware quadratic optimization using partially homomorphic encryption," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, Dec 2016, pp. 5053–5058.

[16] A. B. Alexandru, M. Morari, and G. J. Pappas, "Cloud-based MPC with Encrypted Data," *ArXiv e-prints*, 2018.

[17] A. B. Alexandru, K. Gatsis, Y. Shoukry, S. A. Seshia, P. Tabuada, and G. J. Pappas, "Cloud-based Quadratic Optimization with Partially Homomorphic Encryption," *arXiv e-prints*, Sep. 2018.

[18] M. Schulze Darup, A. Redder, I. Shames, F. Farokhi, and D. Quevedo, "Towards encrypted mpc for linear constrained systems," *IEEE Control Systems Letters*, vol. 2, no. 2, pp. 195–200, April 2018.

[19] J. Cortes, G. E. Dullerud, S. Han, J. L. Ny, S. Mitra, and G. J. Pappas, "Differential privacy in control and network systems," in *2016 IEEE 55th Conference on Decision and Control*, Dec 2016, pp. 4252–4272.

[20] F. Koufogiannis and G. J. Pappas, "Differential privacy for dynamical sensitive data," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, Dec 2017, pp. 1118–1125.

[21] C. Dwork, "Differential privacy," in *Automata, Languages and Programming*, M. Bugliesi, B. Preneel, V. Sassone, and I. Wegener, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 1–12.

[22] P. Weeraddana and C. Fischione, "On the privacy of optimization," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 9502 – 9508, 2017, 20th IFAC World Congress.

[23] Z. Xu and Q. Zhu, "Secure and resilient control design for cloud enabled networked control systems," in *Proceedings of the First ACM Workshop on Cyber-Physical Systems-Security and/or PrivaCy*, ser. CPS-SPC '15, 2015, pp. 31–42.

[24] D. Wu, B. C. Lesieutre, P. Ramanathan, and B. Kakunoori, "Preserving privacy of AC optimal power flow models in multi-party electric grids," *IEEE Transactions on Smart Grid*, vol. 7, no. 4, pp. 2050–2060, July 2016.

[25] A. Sultangazin and P. Tabuada, "Towards the use of symmetries to ensure privacy in control over the cloud," in *2018 IEEE 57th Conference on Decision and Control*, Dec 2018, pp. 5008–5–13.

[26] A. Sultangazin, S. Diggavi, and P. Tabuada, "Protecting the privacy of networked multi-agent systems controlled over the cloud," in *2018 27th International Conference on Computer Communication and Networks (ICCCN)*, July 2018, pp. 1–7.

[27] W. Respondek, "Symmetries and minimal flat outputs of nonlinear control systems," in *New Trends in Nonlinear Dynamics and Control and their Applications*, W. Kang, C. Borges, and M. Xiao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, pp. 65–86.

[28] M. A. Beitia, J. M. Gracia, and I. de Hoyos, "A linear matrix equation: a criterion for block similarity," *Linear and Multilinear Algebra*, vol. 31, pp. 93–118, 1992.

[29] J. M. Lee, *Introduction to Smooth Manifolds*, ser. Graduate Texts in Mathematics. Springer-Verlag New York, 2003.

[30] B. Vandereycken, P. A. Absil, and S. Vandewalle, "Embedded geometry of the set of symmetric positive semidefinite matrices of fixed rank," in *2009 IEEE/SP 15th Workshop on Statistical Signal Processing*, Aug 2009, pp. 389–392.