



Brief paper

Privacy-preserving nonlinear cloud-based model predictive control via affine masking[☆]Kaixiang Zhang^a, Zhaojian Li^{a,*}, Yongqiang Wang^b, Nan Li^c^a Department of Mechanical Engineering, Michigan State University, East Lansing, MI 48824, USA^b Department of Electrical and Computer Engineering, Clemson University, Clemson, SC 29634, USA^c School of Automotive Studies, Tongji University, Shanghai 201804, China

ARTICLE INFO

Article history:

Received 24 September 2022

Received in revised form 5 May 2023

Accepted 22 August 2024

Available online 23 September 2024

Keywords:

Model predictive control

Cloud-based control

Privacy preservation

Output feedback

ABSTRACT

With the advent of 5G technology that presents enhanced communication reliability and ultra-low latency, there is renewed interest in employing cloud computing to perform high performance but computationally expensive control schemes like nonlinear model predictive control (MPC). Such a cloud-based control scheme, however, requires data sharing between the plant (agent) and the cloud, which raises privacy concerns. This is because privacy-sensitive information such as system states and control inputs has to be sent to/from the cloud and thus can be leaked to attackers for various malicious activities. In this paper, we develop a simple yet effective affine masking strategy for privacy-preserving nonlinear MPC. Specifically, we consider external eavesdroppers or honest-but-curious cloud servers that wiretap the communication channel and intend to infer the plant's information including state information and control inputs. An affine transformation-based privacy-preservation mechanism is designed to mask the true states and control signals while reformulating the original MPC problem into a different but equivalent form. We show that the proposed privacy scheme does not affect the MPC performance and it preserves the privacy of the plant such that the eavesdropper is unable to identify the actual value or even estimate a rough range of the private state and input signals. The proposed method is further extended to achieve privacy preservation in cloud-based output-feedback MPC. Simulations are performed to demonstrate the efficacy of the developed approaches.

© 2024 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

1. Introduction

Model predictive control (MPC) is an optimal control paradigm that can explicitly handle system constraints and has enjoyed great successes over the past decade (Allenspach & Ducard, 2021; Li et al., 2019; Liu et al., 2016; Mayne, 2014). Despite their outstanding performances, conventional MPC implementations involve solving an online optimization problem that requires substantial computation power, especially for nonlinear and complex systems. This hinders the deployment of MPC in many resource-limited cyber-physical systems with real-time constraints such as autonomous vehicles and mobile robots. Cloud-based MPC – outsourcing the heavy computation to the cloud with superior computational resources – has received renewed attention (Li

et al., 2023; Schlüter & Darup, 2020; Sultangazin & Tabuada, 2021), partly attributed to the advancement in 5G technologies that can provide reliable communication with negligible latency.

In brief, cloud computing is a unified platform that provides on-demand computing power and data storage services to users (Grossman, 2009). The cloud can offer superior computational capabilities to execute advanced (and computationally expensive) control strategies like nonlinear MPC, as well as incorporate real-time crowdsourced information as a preview to increase situational awareness and enhance system performance (Li et al., 2017, 2014, 2016; Ozatay et al., 2014). A general setup for cloud-based MPC is as follows. First, the plant sends the measured (or estimated) states to the cloud. The cloud then solves a pre-specified MPC problem and sends back the optimal control actions. The system evolves over one step and the process is then repeated. The aforementioned setup has several advantages, including high performance (if the communication has negligible latency), easy deployment, and convenient modification when needed, among others. However, the system states/measurements and control actions need to be transmitted between the cloud and the local agent, raising concerns that

[☆] This research was supported by National Science Foundation, United States under award numbers 2045436 and 2219487. The material in this paper was not presented at any conference. This paper was recommended for publication in revised form by Associate Editor Vijay Gupta under the direction of Editor Christos G. Cassandras.

* Corresponding author.

E-mail addresses: zhangk64@msu.edu (K. Zhang), lizhaoj1@egr.msu.edu (Z. Li), yongqiw@clemson.edu (Y. Wang), li_nan@tongji.edu.cn (N. Li).

outsourcing computation to a cloud might leak private information (e.g., sensor measurements and system states) to an eavesdropper or an untrusted cloud. In fact, several studies have shown that exposing local agent's information to connectivity can indeed lead to security vulnerabilities and various malicious activities (Munteanu et al., 2018; Petit & Shladover, 2015; Xu et al., 2021).

Considering the aforementioned concerns and the growing awareness of security in cyber-physical systems, it is imperative to protect the privacy of agents if cloud-based control is used. As such, several privacy preservation schemes for cloud-based MPC have been proposed, which can be mainly categorized into homomorphic encryption-based methods (Alexandru et al., 2018; Darup, Redder, Quevedo, 2018; Darup, Redder, Shames et al., 2018; Schlüter & Darup, 2020) and algebraic transformation based methods (Naseri et al., 2022; Sultangazin & Tabuada, 2021; Xu & Zhu, 2015, 2017). The homomorphic encryption-based methods exploit cryptography to mask privacy-sensitive information (e.g., system states) while still enabling the cloud to perform the MPC computation with encrypted data. In Darup, Redder, Shames et al. (2018), homomorphic encryption is used to design a secure explicit MPC scheme for linear systems with state and input constraints. The encrypted fast gradient method and proximal gradient method are developed in Alexandru et al. (2018) and Darup, Redder, Quevedo (2018), respectively, to achieve implicit MPC for linear systems with input constraints. Despite strong privacy guarantees for the cloud-based MPC, the induced encryption and decryption procedures are quite computationally heavy, which is thus not suitable for systems with limited onboard resources and stringent real-time constraints.

Different from the homomorphic encryption-based methods, the algebraic transformation-based approaches rely on introducing transformation maps that act as masks, rendering the real signals of a local agent indiscernible by the attacker. More specifically, the main idea of the algebraic transformation methods is to design appropriate transformation maps to protect privacy-sensitive signals and construct a different but equivalent MPC problem. Without knowing the original MPC problem, the cloud will solve the equivalent MPC problem and provide the plant with the corresponding optimal control action. By using inverse transformation maps, the plant can recover the optimal control action to the original problem. This idea has been initially applied to accomplish privacy preservation in optimization (Mangasarian, 2011; Wang et al., 2011; Weeraddana et al., 2013; Weeraddana & Fischione, 2017) and then extended to cloud-based MPCs. For example, in Xu and Zhu (2015), non-singular matrices are utilized to produce a transformation mechanism for linear MPC in the networked control system. In Xu and Zhu (2017), orthogonal matrices are combined with homomorphic encryption to design a hybrid privacy preservation scheme for output-feedback MPC. In Naseri et al. (2022), random transformations are utilized to achieve privacy preservation for set-theoretic MPC. Furthermore, isomorphisms and symmetries are adopted in Sultangazin and Tabuada (2021) as a source of transformation to protect the privacy of system signals.

In this paper, a privacy-preserving cloud-based nonlinear MPC framework is developed to protect system privacy (e.g., states, inputs) via an affine transformation scheme (which is a form of algebraic transformation). We first show that if the cloud is an honest-but-curious adversary or there exists an external eavesdropper, the conventional cloud-based MPC architecture cannot protect the private information of the plant. An affine transformation-based privacy mechanism is then designed to mask the real system state and input signals. With the affine transformation, we reformulate the original MPC problem into a different but equivalent one, which is solved by the cloud.

The solution to the equivalent MPC problem is then received by the local agent and transformed via simple inverse affine transformation to recover the solution to the original problem. A privacy definition is introduced to show that the proposed affine transformation scheme can protect the private system state and input signals from being inferred by the attacker.

The major contributions of this paper include the following. First, we develop a privacy-preserving cloud-based MPC for a class of nonlinear systems. While studies on privacy-preserving cloud MPC for linear systems exist (see e.g., Alexandru et al. (2018), Darup, Redder, Quevedo (2018), Darup, Redder, Shames et al. (2018), Naseri et al. (2022), Schlüter and Darup (2020), Sultangazin and Tabuada (2021) and Xu and Zhu (2015, 2017)), to the authors' best knowledge, this is the first work on privacy-preserving cloud MPC for a class of nonlinear systems with general constraints. Using cloud computing for nonlinear and complex systems makes the most practical sense, as recent advances in compact and powerful onboard computation units are enabling real-time implementations for linear MPCs (but still not for nonlinear MPCs) (Bemporad et al., 2018). We mask the privacy-sensitive signals via affine transformation and reformulate a compatible nonlinear MPC that is equivalent to the original problem, thus with no performance degradation. Furthermore, the affine transformation method is light-weight in computation, which makes it easily applicable to cloud-based control. Second, a new privacy definition, ∞ -diversity with unbounded diameter, is introduced that is suitable for the considered real-time cyber-physical systems. Third, we extend the developed framework to cloud-based nonlinear output-feedback MPC to achieve privacy preservation for nonlinear systems with only output feedback. Finally, simulation examples are presented to demonstrate the efficacy of the developed framework. The proposed approach draws inspiration from algebraic transformation-based methods developed for linear systems (Sultangazin & Tabuada, 2021; Xu & Zhu, 2015, 2017), but there exist significant differences between our work and these references. The scheme proposed in Xu and Zhu (2015) is limited to special objective functions and linear input constraints, and in Xu and Zhu (2017), neither state nor input constraints are considered. In contrast, our approach is designed to address more general MPC problems, encompassing nonlinear systems, objective functions described by general quadratic form, and accounting for state and input constraints. To conceal sensitive information, we employ an affine transformation mechanism and communication protocol similar to that presented in Sultangazin and Tabuada (2021). However, different from the work of Sultangazin and Tabuada (2021) which quantifies privacy via the dimension of the manifold that describes the diversity experienced by the adversary, we tailor the privacy notion for cloud-based nonlinear state-feedback and output-feedback MPC by using set cardinality and diameter. Note that the set dimension-based privacy quantification in Sultangazin and Tabuada (2021) is derived based on the state/input/output matrices of linear systems and cannot be directly applied to nonlinear systems. Our privacy notion works for general nonlinear systems, and it requires that after observing the released data, the adversary has infinite uncertainties on each of its interested entries and the difference between the possible uncertainties could be arbitrarily large.

The rest of this paper is organized as follows. Section 2 introduces the problem formulation including cloud-based MPC and the attack model. Section 3 presents the developed privacy preservation scheme via affine transformations. We then extend the scheme for output-feedback MPC in Section 4. Simulations are presented in Section 5, and finally Section 6 concludes this paper.

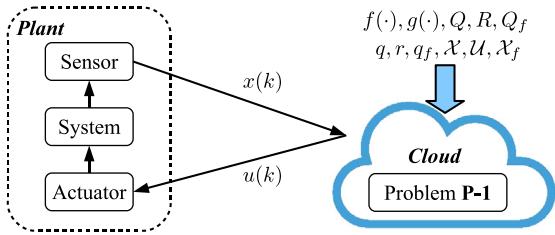


Fig. 1. Cloud-based conventional MPC architecture.

2. Problem formulation

In this section, we present the relevant background of the considered privacy-preserving cloud-based MPC problem. Specifically, we first introduce the conventional cloud-based MPC framework with no privacy protection, followed by a description of the privacy attack model considered in this paper.

2.1. Cloud-based MPC

We consider a class of nonlinear systems which can be described by the following control-affine discrete-time model:

$$x(k+1) = \Phi(x(k), u(k)) = f(x(k)) + g(x(k))u(k), \quad (1)$$

where $x(k) \in \mathbb{R}^n$ is the system state, $u(k) \in \mathbb{R}^m$ is the control input, $\Phi(\cdot, \cdot) \in \mathbb{R}^n$, $f(\cdot) \in \mathbb{R}^n$ and $g(\cdot) \in \mathbb{R}^{n \times m}$ are nonlinear continuous functions characterizing the system dynamics. At each sampling instant k , the following nonlinear MPC problem is solved:

$$\begin{aligned} \mathbf{P - 1} : \min_{u_k} J_N(x(k), \mathbf{U}_k) = & \sum_{i=0}^{N-1} (x_{i|k}^\top Q x_{i|k} + q^\top x_{i|k} + u_{i|k}^\top R u_{i|k} + r^\top u_{i|k}) \\ & + x_{N|k}^\top Q_f x_{N|k} + q_f^\top x_{N|k}, \\ \text{s.t. } & x_{i+1|k} = f(x_{i|k}) + g(x_{i|k})u_{i|k}, i = 0, \dots, N-1, \\ & x_{i|k} \in \mathcal{X}, i = 1, \dots, N-1, \\ & u_{i|k} \in \mathcal{U}, i = 0, \dots, N-1, \\ & x_{N|k} \in \mathcal{X}_f, \\ & x_{0|k} = x(k), \mathbf{U}_k = [u_{0|k}^\top, \dots, u_{N-1|k}^\top]^\top, \end{aligned} \quad (2)$$

which is a receding horizon optimal control problem with state and input constraints. In (2), $J_N(\cdot, \cdot) \in \mathbb{R}$ is the cost function with $Q \in \mathbb{R}^{n \times n}$, $q \in \mathbb{R}^n$, $R \in \mathbb{R}^{m \times m}$, $r \in \mathbb{R}^m$, $Q_f \in \mathbb{R}^{n \times n}$ and $q_f \in \mathbb{R}^n$ being weighting matrices and vectors; $x_{i|k}$ and $u_{i|k}$ are, respectively, the predicted system state and the input i time steps ahead of current time instant k ; $N \in \mathbb{N}_+$ is the prediction horizon; $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{U} \subset \mathbb{R}^m$ are state and input constraint sets, respectively, and $\mathcal{X}_f \subset \mathbb{R}^n$ is the terminal set.

In a conventional MPC, the optimization problem (2) is solved at each time step based on the current state $x(k)$, and the first element of the optimal input sequence $\mathbf{U}_k^* = [u_{0|k}^{*\top}, \dots, u_{N-1|k}^{*\top}]^\top$ is applied to the system, i.e., $u(k) = u_{0|k}^*$, and the system evolves over one step. The process is then repeated. With gentle assumptions and by appropriately selecting the weighting matrix Q_f and terminal set \mathcal{X}_f , the resulting closed-loop system can achieve guaranteed recursive feasibility and asymptotical stability (Rawlings et al., 2017).

The optimization problem in (2) is a nonlinear programming problem that requires significant computation power, which is very challenging to solve onboard considering limited onboard

computation and stringent real-time constraints for many cyber-physical systems. This challenge is exacerbated when the dimension of the system state and the prediction horizon are large. To address this problem, cloud-based MPC is a viable framework where complex computations are outsourced to the cloud that has superior computational power. Specifically, the common cloud-based MPC architecture is shown in Fig. 1, which includes the following two phases:

- Handshaking Phase: The plant sends $\{f(\cdot), g(\cdot), Q, q, R, r, Q_f, q_f, \mathcal{X}, \mathcal{U}, \mathcal{X}_f\}$ to the cloud, that is, the necessary information for the cloud to set up the nonlinear programming problem in (2).
- Execution Phase: At each time step k , the plant first sends its state $x(k)$ to the cloud. Then the cloud computes $u(k)$ by solving the optimization problem (2) and sends the resultant $u(k)$ to the plant. Finally, the plant applies $u(k)$ to the actuators and the system evolves over one step.

2.2. Attack model

As described above, for the conventional cloud-based MPC, the plant needs to provide the cloud with the system state, dynamic model, objective function, and constraints, which may contain confidential information that needs to be protected from an external eavesdropper or the untrusted cloud. In this paper, we consider the following two attack models:

- *Eavesdropping attacks* are attacks in which an external eavesdropper wiretaps communication channels to intercept exchanged messages in an attempt to learn the information about sending parties.
- *Honest-but-curious attacks* are attacks in which the untrusted cloud follows all protocol steps correctly but is curious and collects all received intermediate data in an attempt to learn the information about the plant.

In particular, we consider the case that the privacy-sensitive information is contained in the system state $x(k)$ and input $u(k)$. When cloud-based MPC is adopted in some specific areas, such as intelligent vehicles and smart grids, the disclosure of the system state and input information may induce safety risk (McDaniel & McLaughlin, 2009; Petit & Shladover, 2015). For example, for cooperative control of multiple connected vehicles, the system state and input usually include vehicles' location and velocity messages, which should be well protected to prevent adversaries from using such information to secretly track a vehicle (Corser et al., 2016; Dotzer, 2005) and from engaging in further malicious activities (Hubaux et al., 2004; Xue et al., 2014). In its Readiness Report, the National Highway Traffic Safety Administration acknowledges various privacy issues that must be addressed when implementing vehicle communications, including preventing location tracking (National Highway Traffic Safety Administration, 2014). It is clear that the attacker can successfully eavesdrop on the messages $x(k)$ and $u(k)$ when the conventional cloud-based MPC architecture introduced in Section 2.1 is adopted. The objective of this paper is to develop a masking mechanism to redesign the exchanged information between the plant and the cloud such that an equivalent MPC problem is solved without affecting system performance while preventing the external eavesdropper or untrusted cloud from inferring the system state and input.

3. Main results

In this section, we present our privacy-preserving cloud-based nonlinear MPC framework. We first show that by applying affine

masking on the states and controls, and transforming the cost terms and system dynamics accordingly, the transformed nonlinear MPC problem solved on the cloud is equivalent to the original problem. We then show that this affine transformation can protect the privacy of the system states and inputs by virtue of indistinguishability.

3.1. Affine masking and problem reformulation

Inspired by the works (Sultangazin & Tabuada, 2021; Xu & Zhu, 2015, 2017) that exploit linear transformations for *linear* MPCs, in this section, we design affine transformation maps to accomplish the privacy protection for the considered cloud-based *nonlinear* MPC. Specifically, two invertible affine maps $\iota_x(\cdot) := \{P_x, t_x\}$ and $\iota_u(\cdot) := \{P_u, t_u\}$ are introduced to transform the state $x(k)$ and input $u(k)$ to the new state $\bar{x}(k)$ and input $\bar{u}(k)$, as follows:

$$\begin{aligned}\bar{x}(k) &= \iota_x(x(k)) = P_x(x(k) + t_x), \\ \bar{u}(k) &= \iota_u(u(k)) = P_u(u(k) + t_u),\end{aligned}\quad (3)$$

where $P_x \in \mathbb{R}^{n \times n}$, $P_u \in \mathbb{R}^{m \times m}$ are arbitrary invertible matrices, and $t_x \in \mathbb{R}^n$, $t_u \in \mathbb{R}^m$ are arbitrary non-zero vectors with compatible dimensions. From (1) and (3), it follows that the transformed system state evolves according to the following dynamics:

$$\bar{x}(k+1) = \bar{\Phi}(\bar{x}(k), \bar{u}(k)) = \bar{f}(\bar{x}(k)) + \bar{g}(\bar{x}(k))\bar{u}(k), \quad (4)$$

where $\bar{f}(\cdot) \in \mathbb{R}^n$ and $\bar{g}(\cdot) \in \mathbb{R}^{n \times m}$ are defined as

$$\begin{aligned}\bar{f}(\bar{x}(k)) &= P_x(f \circ \iota_x^{-1}(\bar{x}(k)) - g \circ \iota_x^{-1}(\bar{x}(k))t_u + t_x), \\ \bar{g}(\bar{x}(k)) &= P_x g \circ \iota_x^{-1}(\bar{x}(k))P_u^{-1},\end{aligned}\quad (5)$$

with \circ denoting function composition and $\iota_x^{-1}(\cdot)$ being the inverse operation of $\iota_x(\cdot)$, i.e., $\iota_x^{-1}(\bar{x}(k)) = P_x^{-1}\bar{x}(k) - t_x$. As will be shown below, the affine maps are able to mask the real system state $x(k)$ and input $u(k)$ to protect the privacy, and in the cloud a new optimization problem with respect to $\bar{x}(k)$, $\bar{u}(k)$, and the new system dynamics (4) is solved. Specifically, with the affine maps $\{P_x, t_x\}$ and $\{P_u, t_u\}$, one can show that **P-1** can be transformed into the following problem:

$$\begin{aligned}\mathbf{P-2:} \min_{\bar{u}_k} \bar{J}_N(\bar{x}(k), \bar{U}_k) &= \sum_{i=0}^{N-1} (\bar{x}_{i|k}^\top \bar{Q} \bar{x}_{i|k} + \bar{q}^\top \bar{x}_{i|k} + \bar{u}_{i|k}^\top \bar{R} \bar{u}_{i|k} + \bar{r}^\top \bar{u}_{i|k}) \\ &\quad + \bar{x}_{N|k}^\top \bar{Q}_f \bar{x}_{N|k} + \bar{q}_f^\top \bar{x}_{N|k}, \\ \text{s.t. } \bar{x}_{i+1|k} &= \bar{f}(\bar{x}_{i|k}) + \bar{g}(\bar{x}_{i|k})\bar{u}_{i|k}, i = 0, \dots, N-1, \\ \bar{x}_{i|k} &\in \bar{\mathcal{X}}, i = 1, \dots, N-1, \\ \bar{u}_{i|k} &\in \bar{\mathcal{U}}, i = 0, \dots, N-1, \\ \bar{x}_{N|k} &\in \bar{\mathcal{X}}_f, \\ \bar{x}_{0|k} &= \bar{x}(k), \bar{U}_k = [\bar{u}_{0|k}^\top, \dots, \bar{u}_{N-1|k}^\top]^\top,\end{aligned}\quad (6)$$

where $\bar{Q} \in \mathbb{R}^{n \times n}$, $\bar{q} \in \mathbb{R}^n$, $\bar{R} \in \mathbb{R}^{m \times m}$, $\bar{r} \in \mathbb{R}^m$, $\bar{Q}_f \in \mathbb{R}^{n \times n}$, and $\bar{q}_f \in \mathbb{R}^n$ are defined as

$$\begin{aligned}\bar{Q} &= P_x^{-\top} Q P_x^{-1}, \quad \bar{q} = P_x^{-\top} q - 2P_x^{-\top} Q t_x, \\ \bar{R} &= P_u^{-\top} R P_u^{-1}, \quad \bar{r} = P_u^{-\top} r - 2P_u^{-\top} R t_u, \\ \bar{Q}_f &= P_x^{-\top} Q_f P_x^{-1}, \quad \bar{q}_f = P_x^{-\top} q_f - 2P_x^{-\top} Q_f t_x.\end{aligned}\quad (7)$$

Moreover, in (6), $\bar{\mathcal{X}}$, $\bar{\mathcal{X}}_f$ and $\bar{\mathcal{U}}$ are the corresponding constraint sets of \mathcal{X} , \mathcal{X}_f and \mathcal{U} under the affine maps $\{P_x, t_x\}$ and $\{P_u, t_u\}$, respectively. This indicates that $\forall x \in \mathcal{X}$, $\iota_x(x) = P_x(x + t_x) \in \bar{\mathcal{X}}$; vice versa $\forall \bar{x} \in \bar{\mathcal{X}}$, $\iota_x^{-1}(\bar{x}) = P_x^{-1}\bar{x} - t_x \in \mathcal{X}$ (similarly for \mathcal{X}_f , $\bar{\mathcal{X}}_f$ and \mathcal{U} , $\bar{\mathcal{U}}$).

After introducing the affine maps, compared to the conventional cloud-based MPC in Section 2.1, our privacy-preserving cloud-based nonlinear MPC architecture is modified as shown in Fig. 2:

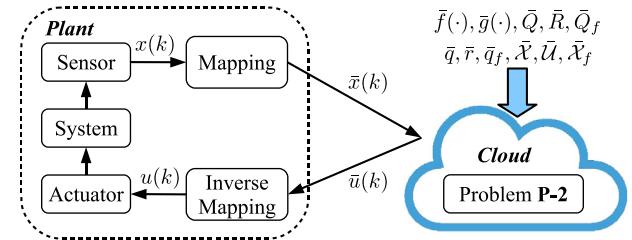


Fig. 2. Cloud-based privacy-preserving MPC architecture with affine masking.

- **Handshaking Phase:** Given the affine maps $\{P_x, t_x\}$ and $\{P_u, t_u\}$, the plant transforms its system dynamics, objective function, and constraint sets into

$$\{\bar{f}(\cdot), \bar{g}(\cdot), \bar{Q}, \bar{q}, \bar{R}, \bar{r}, \bar{Q}_f, \bar{q}_f, \bar{\mathcal{X}}, \bar{\mathcal{U}}, \bar{x}(k), \bar{u}(k)\}$$

and sends them to the cloud to provide necessary information for the cloud to set up the nonlinear programming problem (6).

- **Execution Phase:** At each time step k , the plant first encodes $x(k)$ into $\bar{x}(k)$ with $\{P_x, t_x\}$ and sends $\bar{x}(k)$ to the cloud. Then the cloud computes $\bar{u}(k)$ by solving the optimization problem (6) and sends the solution $\bar{u}(k)$ to the plant. Finally, the plant uses $\{P_u, t_u\}$ to decode $\bar{u}(k)$, i.e., $u(k) = \iota_u^{-1}(\bar{u}(k)) = P_u^{-1}\bar{u}(k) - t_u$, and then applies the resultant $u(k)$ to the actuators. The system then evolves over one step.

Remark 1. Compared to the conventional MPC architecture, the privacy-preserving MPC architecture requires the plant to mask the real state $x(k)$ into $\bar{x}(k)$ and decode $\bar{u}(k)$ into $u(k)$ by using the affine transformation. The affine transformation relies on matrix multiplication, whose time complexity is no greater than $O(n^3)$, where n is the dimension of the transformed variables.

Note that under the privacy-preserving cloud-based MPC architecture, the exchanged information between the plant and the cloud during the execution phase is $\bar{x}(k)$ and $\bar{u}(k)$, instead of the actual system state $x(k)$ and input $u(k)$. In the sequel, we first show that the transformed MPC problem solved on the cloud is equivalent to the original MPC problem, and we then show that the privacy of $x(k)$ and $u(k)$ is protected.

Assumption 1. Both the external eavesdropper and untrusted cloud can get access to the exchanged information between the plant and the cloud, i.e., $\bar{x}(k)$ and $\bar{u}(k)$, but they do not have any prior knowledge about the dynamic system, affine transformation scheme, and the affine maps $\{P_x, t_x\}$ and $\{P_u, t_u\}$.

Lemma 1. Under the affine transformation mechanism, the optimization problem **P-2** is equivalent to **P-1**, i.e., if $\bar{U}_k^* = [\bar{u}_{0|k}^{*\top}, \dots, \bar{u}_{N-1|k}^{*\top}]^\top$ is a local (resp. global) minimizer of **P-2**, then the transformed control via inverse mapping $U_k^* = [u_{0|k}^{*\top}, \dots, u_{N-1|k}^{*\top}]^\top = [(P_u^{-1}\bar{u}_{0|k}^{*\top} - t_u)^\top, \dots, (P_u^{-1}\bar{u}_{N-1|k}^{*\top} - t_u)^\top]^\top$ is a local (resp. global) minimizer of **P-1**.

Proof. Let $\bar{X}_k^* = [\bar{x}_{0|k}^{*\top}, \dots, \bar{x}_{N|k}^{*\top}]^\top$ and $X_k^* = [x_{0|k}^{*\top}, \dots, x_{N|k}^{*\top}]^\top$ be the state sequences corresponding to \bar{U}_k^* and U_k^* , respectively. As \bar{U}_k^* is a minimizer of **P-2**, \bar{X}_k^* and \bar{U}_k^* satisfy the dynamic model (4) and the constraints described by $\{\bar{\mathcal{X}}, \bar{\mathcal{U}}, \bar{\mathcal{X}}_f\}$. According to (3) and the formulation of problem **P-1** and **P-2**, it can be concluded that if U_k^* is the inverse mapping of \bar{U}_k^* under $\{P_u, t_u\}$, then X_k^* and U_k^* are the state and input sequences of dynamic

system (1) and \mathbf{X}_k^* is the inverse mapping of $\bar{\mathbf{X}}_k^*$ under $\{P_x, t_x\}$. In problem **P-2**, $\bar{\mathcal{X}}$, $\bar{\mathcal{X}}_f$ and $\bar{\mathcal{U}}$ are defined as the corresponding constraint sets of \mathcal{X} , \mathcal{X}_f and \mathcal{U} under the affine maps $\{P_x, t_x\}$ and $\{P_u, t_u\}$, respectively. Therefore, if $\bar{\mathbf{X}}_k^*$ and $\bar{\mathbf{U}}_k^*$ satisfy the constraints described by $\{\bar{\mathcal{X}}, \bar{\mathcal{U}}, \bar{\mathcal{X}}_f\}$, then \mathbf{X}_k^* and \mathbf{U}_k^* will satisfy the constraints described by $\{\mathcal{X}, \mathcal{U}, \mathcal{X}_f\}$.

With our designed state and control transformations in (3), the cost term transformations in (7), and the definitions of $J_N(\cdot, \cdot)$ and $J_N(\cdot, \cdot)$, it can be shown that

$$J_N(x(k), \mathbf{U}_k) = \bar{J}_N(\bar{x}(k), \bar{\mathbf{U}}_k) + \varrho, \quad (8)$$

where $\varrho = \sum_{i=1}^{N-1} (t_x^\top Q t_x - q^\top t_x + t_u^\top R t_u - r^\top t_u) + t_x^\top Q_f t_x - q_f^\top t_x \in \mathbb{R}$ is a constant. We now use proof by contradiction, that is, we assume that $\bar{\mathbf{U}}_k^*$ is a local (resp. global) minimizer of problem **P-2** within domain $\bar{\mathcal{U}}_{\text{local}}$ but \mathbf{U}_k^* is not a local (resp. global) minimizer of problem **P-1** within domain $\mathcal{U}_{\text{local}}$, where $\mathcal{U}_{\text{local}}$ is the corresponding domain of $\mathcal{U}_{\text{local}}$ under the affine map $\{P_u, t_u\}$. This means that there exists an optimal sequence (other than \mathbf{U}_k^*) $\mathbf{U}_k^{**} = [u_{0|k}^{**\top}, \dots, u_{N-1|k}^{**\top}]^\top \in \mathcal{U}_{\text{local}}$ such that

$$J_N(x(k), \mathbf{U}_k^{**}) < J_N(x(k), \mathbf{U}_k^*). \quad (9)$$

Let $\bar{\mathbf{U}}_k^{**} = [\bar{u}_{0|k}^{**\top}, \dots, \bar{u}_{N-1|k}^{**\top}]^\top = [(P_u(u_{0|k}^{**} + t_u))^\top, \dots, (P_u(u_{N-1|k}^{**} + t_u))^\top]^\top \in \bar{\mathcal{U}}_{\text{local}}$. According to (8), (9) can be rewritten as

$$\bar{J}_N(\bar{x}(k), \bar{\mathbf{U}}_k^{**}) + \varrho < \bar{J}_N(\bar{x}(k), \bar{\mathbf{U}}_k^*) + \varrho, \quad (10)$$

which contradicts the assumption that $\bar{\mathbf{U}}_k^*$ is a local (resp. global) minimizer of problem **P-2**. The proof is complete. \square

Lemma 1 reveals that the transformed MPC problem is a different yet equivalent form of the original MPC problem. Thus, if the original MPC ensures properties such as recursive feasibility and asymptotical stability, then the transformed formulation preserves these theoretical guarantees.

3.2. Privacy preservation

We next discuss the privacy notion used in this paper. As mentioned in the previous section, the attacker aims to infer the system state $x(k)$ and control input $u(k)$. Under the privacy-preserving cloud-based MPC architecture discussed above, the attacker will have access to $\bar{x}(k)$ and $\bar{u}(k)$ at each time step k , and we need to show that for any $\kappa \in \mathbb{N}_+$, $x_{[0,\kappa]} = \{x(0), \dots, x(\kappa)\}$ and $u_{[0,\kappa]} = \{u(0), \dots, u(\kappa)\}$ cannot be identified from $\bar{x}_{[0,\kappa]} = \{\bar{x}(0), \dots, \bar{x}(\kappa)\}$ and $\bar{u}_{[0,\kappa]} = \{\bar{u}(0), \dots, \bar{u}(\kappa)\}$. To facilitate the following development, two triples Ω and $\bar{\Omega}$ are defined as

$$\begin{aligned} \Omega &= \{f(\cdot, g(\cdot)), J_N(\cdot, \cdot), \{\mathcal{X}, \mathcal{U}, \mathcal{X}_f\}\}, \\ \bar{\Omega} &= \{\bar{f}(\cdot, \bar{g}(\cdot)), \bar{J}_N(\cdot, \cdot), \{\bar{\mathcal{X}}, \bar{\mathcal{U}}, \bar{\mathcal{X}}_f\}\}. \end{aligned} \quad (11)$$

It can be seen that the triples Ω and $\bar{\Omega}$ can be used to define the optimization problem in (2) and (6), respectively. We call $\{x_{[0,\kappa]}, u_{[0,\kappa]}\}$ a solution to the optimization problem (2) defined by Ω if $\{x_{[0,\kappa]}, u_{[0,\kappa]}\}$ is a trajectory of the nonlinear system $\{f(\cdot, g(\cdot))\}$ where the control input $u(k)$ at each time step k is solved by minimizing objective function $J_N(\cdot, \cdot)$ under constraints described by $\{\mathcal{X}, \mathcal{U}, \mathcal{X}_f\}$. Moreover, we use $\Omega \xrightarrow{\{P_x, t_x, P_u, t_u\}} \bar{\Omega}$ to denote that $\bar{\Omega}$ is the transformed triple of Ω under the affine maps $\{P_x, t_x\}$ and $\{P_u, t_u\}$.

Given $\bar{\Omega}$, for any feasible input sequence $\bar{x}_{[0,\kappa]}$ and output sequence $\bar{u}_{[0,\kappa]}$ received by the attacker, the set $\Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})$ is defined as

$$\begin{aligned} \Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}) &= \{x_{[0,\kappa]}, u_{[0,\kappa]} : \exists \{P_x, t_x, P_u, t_u\} \text{ and } \Omega \\ &\text{s.t. } \bar{x}(k) = P_x(x(k) + t_x), \bar{u}(k) = P_u(u(k) + t_u), \\ &\Omega \xrightarrow{\{P_x, t_x, P_u, t_u\}} \bar{\Omega}, \text{ and } \{x_{[0,\kappa]}, u_{[0,\kappa]}\} \text{ is the solution to } \Omega\}. \end{aligned} \quad (12)$$

Essentially, the set $\Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})$ includes all possible values of $\{x_{[0,\kappa]}, u_{[0,\kappa]}\}$ that can be transformed into $\{\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}\}$ with corresponding affine maps $\{P_x, t_x, P_u, t_u\}$. The diameter of $\Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})$, a metric that measures the distance (dissimilarity) between its elements, is defined as

$$\text{Diam}_{\Delta_{\bar{\Omega}}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}) = \sup_{w, w' \in \Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})} |w - w'|_{\min}, \quad (13)$$

where $|w - w'|_{\min} = \min_{l \in \{1, \dots, (n+m)(\kappa+1)\}} |w_l - w'_l|$ with w_l and w'_l being the l th element of w and w' , respectively. Note that $|w - w'|_{\min}$ is used to quantify the minimum element difference between w and w' . If $|w - w'|_{\min} = \delta$, where δ is an arbitrarily positive constant, then $\forall l \in \{1, \dots, (n+m)(\kappa+1)\}$, we have $|w_l - w'_l| \geq \delta$.

Definition 1 (∞ -Diversity with Unbounded Diameter). The privacy of the actual system state $x_{[0,\kappa]}$ and input $u_{[0,\kappa]}$ is preserved if (1) the cardinality of the set $\Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})$ is infinite, and (2) $\text{Diam}_{\Delta_{\bar{\Omega}}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}) = \infty$.

In the ∞ -Diversity with Unbounded Diameter privacy defined above, the first condition requires that there are infinitely many sets of $\{x_{[0,\kappa]}, u_{[0,\kappa]}\}$, $\{P_x, t_x, P_u, t_u\}$ and Ω that can generate the same $\{\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}\}$ received by the attacker. As a result, it is impossible for the attacker to use $\{\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}\}$ to infer the actual system state and input information. Moreover, the second condition requires that the difference between the possible values of each element in $\{x_{[0,\kappa]}, u_{[0,\kappa]}\}$ could be arbitrarily large, and thus the attacker cannot even approximately estimate (e.g., find a finite range or uniquely determine a portion of) the private signals.

We now show that the affine transformation mechanism can achieve privacy preservation based on **Definition 1**.

Theorem 1. Under the affine masking mechanism described in Section 3.1, the system states and control inputs are ∞ -diversity-with-unbounded-diameter private, that is, the attacker cannot infer the actual system state $x(k)$ and input $u(k)$ with any guaranteed accuracy.

Proof. We prove **Theorem 1** by proving the two conditions in **Definition 1**. We first show that under the affine masking scheme, the cardinality of the set $\Delta(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})$ is infinite. Specifically, given the sequence $\{\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}\}$ and $\bar{\Omega}$ accessible to the attacker, for arbitrary affine maps $\iota'_x(\cdot) := \{P'_x, t'_x\}$ and $\iota'_u(\cdot) := \{P'_u, t'_u\}$ such that P'_x and P'_u are invertible, a sequence $\{x'_{[0,\kappa]}, u'_{[0,\kappa]}\}$ and Ω' can be uniquely determined. Recall that $\{x'_{[0,\kappa]}, u'_{[0,\kappa]}\}$ should satisfy $\bar{x}(k) = P'_u(x'(k) + t'_x)$ and $\bar{u}(k) = P'_u u'_k + t'_u$, which indicates that the sequence $\{x'_{[0,\kappa]}, u'_{[0,\kappa]}\}$ can be determined by

$$\begin{aligned} x'(k) &= \iota'^{-1}_x(\bar{x}(k)) = (P'_x)^{-1} \bar{x}(k) - t'_x, \\ u'(k) &= \iota'^{-1}_u(\bar{u}(k)) = (P'_u)^{-1} \bar{u}(k) - t'_u. \end{aligned} \quad (14)$$

Based on (14) and $\bar{\Omega}$, Ω' can be further obtained by following the similar procedure introduced in Section 3.1. As there exist infinitely many such affine maps $\{P'_x, t'_x, P'_u, t'_u\}$, there exist infinitely many $\{x'_{[0,\kappa]}, u'_{[0,\kappa]}\}$ and Ω' such that via proper affine transformations, the attacker will receive the same accessed information: $\{\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}\}$ and $\bar{\Omega}$, which thus satisfies the first condition in **Definition 1**.

We now prove the second condition in **Definition 1**. For any $w, w' \in \Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})$ (i.e., $\{x_{[0,\kappa]}, u_{[0,\kappa]}\}, \{x'_{[0,\kappa]}, u'_{[0,\kappa]}\} \in \Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})$) with $\{P_x, t_x, P_u, t_u\}$ and $\{P'_x, t'_x, P'_u, t'_u\}$ being the corresponding affine maps, we have

$$\begin{aligned} x(k) &= P_x^{-1} \bar{x}(k) - t_x, \quad x'(k) = (P'_x)^{-1} \bar{x}(k) - t'_x, \\ u(k) &= P_u^{-1} \bar{u}(k) - t_u, \quad u'(k) = (P'_u)^{-1} \bar{u}(k) - t'_u. \end{aligned} \quad (15)$$

Based on (15), it can be obtained that

$$\begin{aligned}
 |w - w'|_{\min} &= \left| \begin{array}{l} (P_x^{-1} - (P'_x)^{-1})\bar{x}(0) - (t_x - t'_x) \\ \vdots \\ (P_x^{-1} - (P'_x)^{-1})\bar{x}(\kappa) - (t_x - t'_x) \\ (P_u^{-1} - (P'_u)^{-1})\bar{u}(0) - (t_u - t'_u) \\ \vdots \\ (P_u^{-1} - (P'_u)^{-1})\bar{u}(\kappa) - (t_u - t'_u) \end{array} \right|_{\min} \\
 &= \left| \begin{array}{l} (I_{\kappa+1} \otimes (P_x^{-1} - (P'_x)^{-1}))\bar{x}_{[0,\kappa]} - 1_{\kappa+1} \otimes (t_x - t'_x) \\ (I_{\kappa+1} \otimes (P_u^{-1} - (P'_u)^{-1}))\bar{u}_{[0,\kappa]} - 1_{\kappa+1} \otimes (t_u - t'_u) \end{array} \right|_{\min} \\
 &\geq \left| 1_{\kappa+1} \otimes (t_x - t'_x) \right|_{\min} - \left| (I_{\kappa+1} \otimes (P_x^{-1} - (P'_x)^{-1}))\bar{x}_{[0,\kappa]} \right|_{\max} \\
 &\quad - \left| (I_{\kappa+1} \otimes (P_u^{-1} - (P'_u)^{-1}))\bar{u}_{[0,\kappa]} \right|_{\max}, \tag{16}
 \end{aligned}$$

where \otimes is the Kronecker product, $I_{\kappa+1} \in \mathbb{R}^{(\kappa+1) \times (\kappa+1)}$ is the identity matrix, and $1_{\kappa+1} \in \mathbb{R}^{\kappa+1}$ is the column vector with all the entries being ones. Furthermore, by using (13) and (16), the diameter of the set $\Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})$ can be derived as follows:

$$\begin{aligned}
 \text{Diam}_{\Delta_{\bar{\Omega}}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}) &= \sup_{w, w' \in \Delta_{\bar{\Omega}}(\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]})} |w - w'|_{\min} \\
 &\geq \sup_{t_x, t'_x \in \mathbb{R}^n, t_u, t'_u \in \mathbb{R}^m} \left| \begin{array}{l} 1_{\kappa+1} \otimes (t_x - t'_x) \\ 1_{\kappa+1} \otimes (t_u - t'_u) \end{array} \right|_{\min} \\
 &\quad - \inf_{P_x, P'_x \in \mathbb{R}^{n \times n}, P_u, P'_u \in \mathbb{R}^{m \times m}} \left| \begin{array}{l} (I_{\kappa+1} \otimes (P_x^{-1} - (P'_x)^{-1}))\bar{x}_{[0,\kappa]} \\ (I_{\kappa+1} \otimes (P_u^{-1} - (P'_u)^{-1}))\bar{u}_{[0,\kappa]} \end{array} \right|_{\max} \\
 &= \infty. \tag{17}
 \end{aligned}$$

Thus, the second condition in Definition 1 is satisfied. \square

By following the arguments from the proof of Theorem 1, it is clear that there exist infinitely many sets of Ω (i.e., system dynamics, cost function, and constraint sets) such that via proper affine transformation, the attacker will receive the same accessed information $\bar{\Omega}$. Therefore, the attacker cannot exploit $\bar{\Omega}$ to uniquely determine the actual Ω . Due to the complicated structure of Ω , defining metrics to quantify the difference between the accessible valuations of Ω is non-trivial and needs to be further studied.

Remark 2. Due to communication overhead or resource constraint, elements in the affine maps $\{P_x, t_x\}$ and $\{P_u, t_u\}$ cannot be arbitrary large numbers in practical applications. To disguise the real state and input information, it is beneficial for the plant to choose suitable affine maps such that the transformed data $\{\bar{x}_{[0,\kappa]}, \bar{u}_{[0,\kappa]}\}$ is quite different from the actual one $\{x_{[0,\kappa]}, u_{[0,\kappa]}\}$. Generally, within a bounded set confined by communication overhead or resource constraint, the plant can choose P_x, P_u and t_x, t_u that are distant (in the sense of Frobenius norm, for example) from the identity matrix and zero vector, respectively, to achieve this purpose.

3.3. Discussion on privacy notion and protection scheme

Definition 1 is an extension to the l -diversity (Machanavajjhala et al., 2007) which has been widely adopted in formal privacy analysis on attribute privacy of tabular datasets and has recently been extended to define privacy in linear dynamic networks (Lu & Zhu, 2020). Essentially, l -diversity requires that there are at least l different possible values for the privacy-sensitive data attributes, and a greater l indicates greater indistinguishability. **Definition 1** extends the l -diversity notion by requiring that there exist infinitely many possible sets of states/inputs and affine transformation combinations that can generate the same accessible information for the adversary (∞ -diversity). In addition, the

difference of the states/inputs in these sets can be arbitrarily large (unbounded diameter). This makes the adversary unable to identify the actual value or even estimate a rough range or a portion of the private parameters. Furthermore, the conventional l -diversity works for discrete-valued setting, whereas **Definition 1** is tailored to the considered cloud-based nonlinear MPC with sensitive attributes being continuous-valued. In the following, we discuss the differences between the proposed privacy definition/scheme and other existing privacy notions/schemes (e.g., differential privacy, homomorphic encryption, and affine transformation).

The privacy notions based on statistics or information theory have been widely utilized in the security community, such as differential privacy, entropy, and mutual information. Differential privacy approaches inject random noises into private data in such a way that the adversary cannot infer the private data with high probability (Dwork & Roth, 2014; Huang et al., 2012). For cloud-based nonlinear MPC, such persistent noise injection mechanism will inevitably deteriorate system performance and potentially lead to the violation of state and input constraints, while the proposed privacy definition with the affine masking scheme does not affect system performance as the transformed problem is equivalent to the original one as shown in Section 3.1. Moreover, both entropy and mutual information-based privacy preservation rely on explicit statistical models of source data and side information (Nekouei et al., 2019; Sankar et al., 2013), which, however, are not generally available in the considered cloud-based MPC problem as the state and input signals of the system may not follow any probabilistic distribution.

Various homomorphic encryption-based methods have been designed for privacy-preserving linear MPC, and both semantic security and secret sharing have been used to define privacy. Semantic security requires that no additional information about a plaintext can be inferred using its ciphertext by the adversary. It is worth noting that the encryption techniques with semantic security guarantees (Alexandru et al., 2018; Darup, Redder, Quevedo, 2018; Darup, Redder, Shames et al., 2018; Schlüter & Darup, 2020) only allow the cloud (which has the public key but not the private key) to perform simple linear mathematical operations on encrypted data, making them applicable only for linear systems and difficult, if not impossible, to be extended to the considered nonlinear system with complicated operations. In addition, secret sharing allows to divide and reconstruct secret data in such a way that the individual shareholders reveal nothing about the secret (Shamir, 1979). It is an effective tool to achieve privacy-preserving cloud-based control (Darup & Jager, 2019; Schlor et al., 2021) but it requires using multiple shareholders/clouds that are not colluding, resulting in a more complex system structure. Instead of relying on data division and sharing to multiple shareholders, the proposed method exploits affine transformation to mask sensitive system information, which does not require multiple clouds to facilitate the design of privacy preservation scheme.

The proposed affine masking scheme is inspired by the algebraic transformation-based works designed for linear systems but there exist several differences. In Xu and Zhu (2015), the linear MPC problem with linear input constraints is first transformed into a quadratic problem, and then a transformation mechanism based on non-singular matrices is designed to mask sensitive information. Xu and Zhu (2017) combine orthogonal matrices with homomorphic encryption to design a hybrid privacy preservation scheme for non-constrained linear MPC. Note that their transformation mechanisms are designed for specific linear MPC forms and thus cannot be applied to the considered nonlinear MPC with state and input constraints. In Sultangazin and Tabuada (2021), the dimension of the manifold describing the diversity experienced by the adversary is used as a measure

of privacy. The derivation of the set dimension based privacy notion relies on the system's linear characteristics, and thus it is difficult (if not impossible) to extend this notion to nonlinear systems. In this paper, we exploit the set cardinality and diameter to quantify the privacy for cloud-based nonlinear MPC, which applies to general nonlinear systems with constraints and can provide stronger privacy guarantees. Furthermore, in Naseri et al. (2022), the transformation-based technique is incorporated into set-theoretic MPC to protect the privacy of a linear system subject to bounded disturbance, while no rigorous notion is introduced to analyze the privacy guarantees. Our work focuses on privacy preservation in nonlinear MPC and the development of the privacy notion.

Although the proposed method circumvents some issues that arise in existing privacy notions, it has certain limitations. One limitation is that it does not consider the case in which the external eavesdropper or untrusted cloud has auxiliary information about the dynamic system and the affine transformation scheme (see Assumption 1), whereas differential privacy and semantic security are immune to arbitrary auxiliary information.

Remark 3. In summary, the proposed affine masking strategy for nonlinear state-feedback MPC makes two technical contributions. First, we tailor the affine masking technique to conceal sensitive information and reformulate the original nonlinear MPC into an equivalent formulation, achieving privacy preservation without compromising control performance. Different from homomorphic encryption-based methods that are limited to linear MPC and incur tedious encryption and decryption procedures, the proposed strategy is applicable to a class of control-affine nonlinear systems and is computationally efficient. Second, we introduce a new privacy definition that uses both set cardinality and diameter to facilitate the privacy quantification for nonlinear MPC. Existing transformation-based approaches rely on linear system characteristics, while our privacy notion extends the existing approaches to preserve privacy for nonlinear systems and employs the set cardinality and diameter to measure the uncertainties on each element of interest to the adversary, making it applicable to nonlinear systems with constraints.

4. Extension to output-feedback MPC

The aforementioned cloud-based MPC methods require that all system states are measurable to perform the state-feedback control. However, for some systems, not all states are accessible but an output vector is available for output feedback control designs. Therefore, in this section, we extend the privacy-preserving cloud-based MPC design to the output-feedback case. Specifically, let $y(k) \in \mathbb{R}^p$ be the system output described by

$$y(k) = Cx(k), \quad (18)$$

where $C \in \mathbb{R}^{p \times n}$. We assume that the system is observable and the state $x(k)$ can be estimated via a high-gain observer (Khalil, 2002) in the following form:

$$\hat{x}(k+1) = \Phi(\hat{x}(k), u(k)) + H(y(k) - C\hat{x}(k)), \quad (19)$$

where $\hat{x}(k) \in \mathbb{R}^n$ is the estimate of $x(k)$ and $H \in \mathbb{R}^{n \times p}$ is the gain matrix. The estimated state $\hat{x}(k)$ is then fed into the MPC problem (2) to obtain the solutions. Under the output-feedback case, the conventional cloud-based MPC is typically implemented as follows:

- Handshaking Phase: The plant sends

$$\{f(\cdot), g(\cdot), Q, q, R, r, Q_f, q_f, \mathcal{X}, \mathcal{U}, \mathcal{X}_f, H, C\}$$

to the cloud, which are necessary information for the cloud to perform state estimation and subsequent MPC based on the estimated state.

- Execution Phase: At each time step k , the plant first sends $y(k)$ to the cloud. Then the cloud estimates the system state \hat{x}_k via (19), computes $u(k)$ based on \hat{x}_k by solving the optimization problem shown in (2) and sends $u(k)$ to the plant. Finally, the plant applies $u(k)$ to the actuators and the system evolves over one step.

The objective now is to avoid leaking the privacy-sensitive information $y(k)$, $\hat{x}(k)$, and $u(k)$ to the attacker. Similar to (3), an invertible affine map is introduced to mask $y(k)$ as follows:

$$\bar{y}(k) = \iota_y(y(k)) = P_y(y(k) + t_y), \quad (20)$$

where $P_y \in \mathbb{R}^{p \times p}$ is an invertible matrix and $t_y \in \mathbb{R}^p$ is an offset vector. According to (3), (18) and (20), it can be obtained that

$$\bar{y}(k) = \bar{C}\bar{x}(k) + \bar{\sigma}_y, \quad (21)$$

with $\bar{C} \in \mathbb{R}^{p \times n}$ and $\bar{\sigma}_y \in \mathbb{R}^p$ being defined as

$$\bar{C} = P_y C P_x^{-1}, \quad \bar{\sigma}_y = P_y(-Ct_x + t_y). \quad (22)$$

Moreover, from (4), (19) and (21), it can be shown that $\bar{x}(k)$ can be estimated with $\bar{y}(k)$ via the following observer:

$$\hat{\bar{x}}(k+1) = \bar{\Phi}(\hat{\bar{x}}(k), \bar{u}(k)) + \bar{H}(\bar{y}(k) - \bar{C}\hat{\bar{x}}(k) - \bar{\sigma}_y), \quad (23)$$

where $\bar{H} \in \mathbb{R}^{n \times p}$ is given by

$$\bar{H} = P_x H P_y^{-1}. \quad (24)$$

The cloud-based privacy-preserving MPC under the output-feedback setup can then be performed with the following modified procedures:

- Handshaking Phase: Given the affine maps $\{P_x, t_x\}$, $\{P_u, t_u\}$ and $\{P_y, t_y\}$, the plant transforms its system dynamics, objective function, constraint sets and observer into

$$\{\bar{f}(\cdot), \bar{g}(\cdot), \bar{Q}, \bar{q}, \bar{R}, \bar{r}, \bar{Q}_f, \bar{q}_f, \bar{\mathcal{X}}, \bar{\mathcal{U}}, \bar{\mathcal{X}}_f, \bar{H}, \bar{C}, \bar{\sigma}_y\}$$

and sends them to the cloud.

- Execution Phase: At each time step k , the plant first encodes $y(k)$ into $\bar{y}(k) = P_y(y(k) + t_y)$ and sends $\bar{y}(k)$ to the cloud. Then the cloud estimates the system state via (23), computes $\bar{u}(k)$ by solving the optimization problem (6) and sends $\bar{u}(k)$ to the plant. Finally, the plant uses $\{P_u, t_u\}$ to decode $\bar{u}(k)$ (i.e., $u(k) = \iota_u^{-1}(\bar{u}(k)) = P_u^{-1}\bar{u}(k) - t_u$) and then applies $u(k)$ to the actuators. The system evolves over one step.

Theorem 2. Under the affine masking mechanism described in this subsection, the system outputs, states, and control inputs are ∞ -diversity-with-unbounded-diameter private, that is, the attacker cannot infer the actual outputs $y(k)$, system state $x(k)$ and input $u(k)$ with any guaranteed accuracy.

Proof. The proof follows similar arguments in Theorem 1. \square

Remark 4. In contrast to Section 3, which employs affine maps to conceal real state and input information in state-feedback MPC, the privacy preservation scheme for output-feedback MPC introduces an additional affine map to mask the real system output. This process also entails reformulating the original high-gain observer into a compatible form, enabling estimation of the transformed system state with the transformed output. The combination of the affine masking strategy and observer reformulation is crucial to ensure that the original output-feedback MPC is shaped into a different but equivalent one, which guarantees that the private information is protected with no performance degradation.

5. Simulation results

In this section, we perform numerical simulations to demonstrate the efficacy of the developed approach. All computations are performed in MATLAB 2022a on a laptop with an Intel i7-10710U CPU with 6 cores, 1.6 GHz clock rate, and 16 GB RAM. We consider the regulation control problem of a quadrotor aerial vehicle. The system state and input of the quadrotor aerial vehicle are defined as $x = [\xi^\top, \eta^\top, \dot{\xi}^\top, \dot{\eta}^\top]^\top \in \mathbb{R}^{12}$ and $u = [u_1, u_2, u_3, u_4]^\top \in \mathbb{R}^4$, respectively, where $\xi = [\xi_x, \xi_y, \xi_z]^\top \in \mathbb{R}^3$ represents the position of the quadrotor mass center expressed in the inertial frame, $\eta = [\phi, \theta, \psi]^\top \in \mathbb{R}^3$ represents the roll, pitch, and yaw angles, and u_i ($i = 1, 2, 3, 4$) represents the squared angular velocity of the i th rotor. The continuous-time model of the quadrotor can be described by (Raffo et al., 2010):

$$\ddot{\xi} = -e_3 g + \frac{Re_3}{m} U_1, \quad \ddot{\eta} = M(\eta)^{-1} (\tau - C(\eta, \dot{\eta})\dot{\eta}), \quad (25)$$

where $e_3 = [0, 0, 1]^\top$, $g = 9.81 \text{ m/s}^2$ is the gravity acceleration, $m = 2 \text{ kg}$ is the quadrotor mass, $R \in \mathbb{SO}(3)$ is the rotation matrix, $M(\eta)$ is the state-dependent inertia matrix, and $C(\eta, \dot{\eta})$ is the Coriolis matrix. Detailed expressions of R , $M(\eta)$ and $C(\eta, \dot{\eta})$ can be found in Raffo et al. (2010). In addition, U_1 denotes the total thrust of the rotors, and τ denotes the torques in the roll, pitch, and yaw angular directions. U_1 and τ are formulated with u , as follows:

$$U_1 = \alpha(u_1 + u_2 + u_3 + u_4), \quad \tau = \begin{bmatrix} l\alpha(-u_2 + u_4) \\ l\alpha(-u_1 + u_3) \\ \beta(-u_1 + u_2 - u_3 + u_4) \end{bmatrix}, \quad (26)$$

where $l = 0.25 \text{ m}$ is the distance between the rotor and the center of mass, $\alpha = 1$ is the lift constant, and $\beta = 0.2$ is the drag constant. We discretize the continuous-time model (25) with a sampling time of $\Delta T = 0.1 \text{ s}$ by using Euler's method. The control objective is to regulate the plant from the initial state $x_0 = [-1, 1, 1.5, 0, 0, 0, 0, 0, 0, 0, 0, 0]^\top$ to the desired state $x_d = 0_{12}$ by using the cloud-based MPC schemes. For the MPC formulation, the weighting matrices and vectors are selected as $Q = Q_f = \text{diag}(300, 300, 300, 300, 300, 300, 0, 0, 0, 0, 0, 0)$, $q = q_f = 0_{12}$, $R = \text{diag}(0.1, 0.1, 0.1, 0.1)$ and $r = 0_4$, and the system state and input are subjected to the constraints $-10 \leq \xi \leq 10$ and $0 \leq u \leq 10$, respectively. Moreover, the affine maps $\{P_u, t_u\}$ and $\{P_x, t_x\}$ are chosen as

$$P_u = \begin{bmatrix} -2 & -3 & 0 & 0 \\ 0.3 & -1.6 & 0 & 0 \\ 0 & 0 & -2 & -1 \\ 0 & 0 & 1.2 & -3 \end{bmatrix}, \quad t_u = \begin{bmatrix} 15 \\ -8 \\ 10 \\ -12 \end{bmatrix},$$

$$P_x = \begin{bmatrix} P_{x,\xi} & 0 & 0 & 0 \\ 0 & P_{x,\eta} & 0 & 0 \\ 0 & 0 & P_{x,\dot{\xi}} & 0 \\ 0 & 0 & 0 & P_{x,\dot{\eta}} \end{bmatrix}, \quad t_x = \begin{bmatrix} t_{x,\xi} \\ t_{x,\eta} \\ t_{x,\dot{\xi}} \\ t_{x,\dot{\eta}} \end{bmatrix},$$

$$P_{x,\xi} = \begin{bmatrix} -3 & 1.5 & 0.1 \\ -1 & -2 & 0.2 \\ 0.5 & -1 & -2.5 \end{bmatrix}, \quad t_{x,\xi} = \begin{bmatrix} 2 \\ -3 \\ 1 \end{bmatrix}, \quad P_{x,\dot{\xi}} = -2I_3, \quad t_{x,\dot{\xi}} = \begin{bmatrix} -1 \\ -8 \\ 7 \end{bmatrix},$$

$$P_{x,\eta} = \begin{bmatrix} -2 & -1.5 & 0 \\ -0.8 & -2.5 & 0 \\ 0 & -1.6 & -2 \end{bmatrix}, \quad t_{x,\eta} = \begin{bmatrix} -5 \\ 4 \\ 6 \end{bmatrix}, \quad P_{x,\dot{\eta}} = -1.5I_3, \quad t_{x,\dot{\eta}} = \begin{bmatrix} -3 \\ -4 \\ 3 \end{bmatrix}.$$

The state and input signals of the quadrotor are privacy-sensitive since the eavesdropper can use them to infer the quadrotor's position and velocity information and then track or

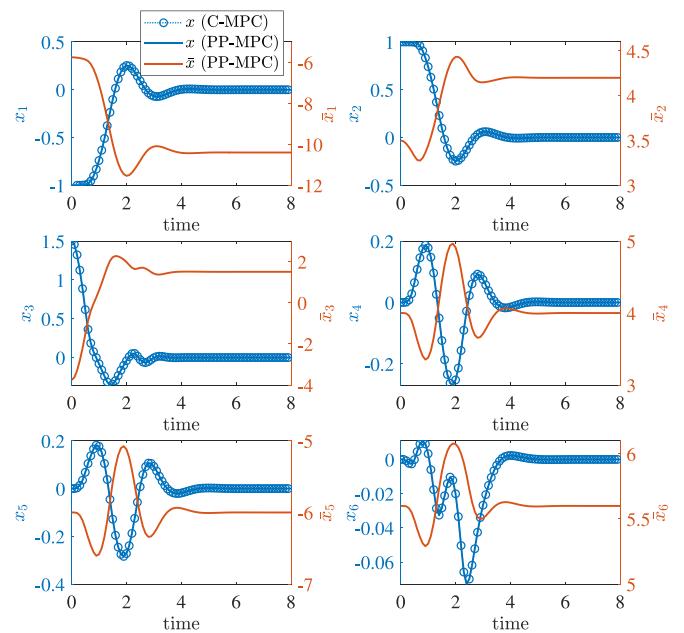


Fig. 3. System state evolution of conventional and privacy-preserving state-feedback MPC. C-MPC refers to conventional MPC and PP-MPC refers to privacy-preserving MPC.

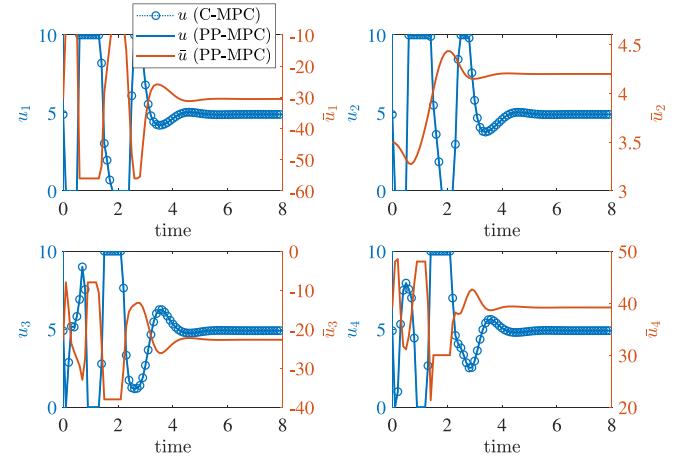


Fig. 4. System input evolution of conventional and privacy-preserving state-feedback MPC.

attack the quadrotor. We evaluate the conventional and privacy-preserving MPC schemes with state feedback. The simulation results are presented in Figs. 3 and 4. Fig. 3 (4) illustrates the state (input) trajectory under conventional MPC and the real and transformed state (input) trajectories under privacy-preserving MPC. It is clear that the state and input trajectories obtained from the privacy-preserving MPC are identical to the ones obtained by the conventional MPC. This aligns with the theoretical findings concluded in Lemma 1, affirming that the affine transformation mechanism maintains control performance equivalent to conventional MPC. Meanwhile, as shown in Figs. 3 and 4, under the privacy-preserving MPC, the state and input information collected by the cloud diverges significantly from the actual one. This observation underscores the efficacy of our proposed method in privacy preservation. According to Definition 1 and Theorem 1, the proposed method ensures the existence of infinitely many sets of states/inputs capable of generating the same accessible information (i.e., $\bar{x}(k)$ and $\bar{u}(k)$) for the adversary. The difference

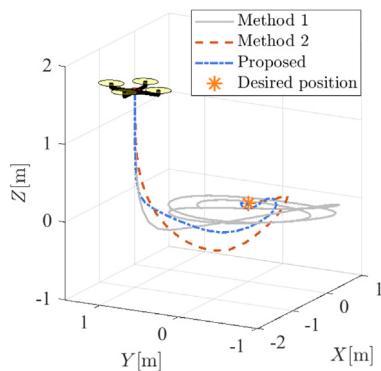


Fig. 5. Trajectory evolution of the quadrotor, resulting from Method 1 (Darup, Redder, Quevedo, 2018), Method 2 (Sultangazin & Tabuada, 2021), and the proposed method.

Table 1
Comparison of accumulative cost and average computation time.

	Method 1	Method 2	Proposed
Accumulative cost [$\times 10^4$]	2.9260	1.2116	1.1091
Average computation time [ms]	69.9666	0.0510	0.0499

The average computation time refers to the time required by the plant for implementing operations under different privacy-preserving methods.

among these sets could be arbitrarily large, which makes the adversary unable to infer $x(k)$ and $u(k)$.

For comparison, two existing privacy-preserving methods, i.e., Method 1 (Darup, Redder, Quevedo, 2018) and Method 2 (Sultangazin & Tabuada, 2021), are tested in this simulation scenario. Method 1 (Darup, Redder, Quevedo, 2018) uses homomorphic encryption to conceal sensitive information, while Method 2 (Sultangazin & Tabuada, 2021) employs transformation-based techniques to prevent privacy leakage. Since both methods are designed for linear MPC, the nonlinear system (25) is linearized at the desired position to facilitate implementation. The motion trajectories of the quadrotor under different control schemes are illustrated in Fig. 5. It is clear that the proposed method can effectively regulate the quadrotor to the desired position with minimal trajectory fluctuations. Moreover, Table 1 presents the accumulative cost (i.e., $\sum(x^\top Qx + q^\top x + u^\top Ru + r^\top u)$) and the average computation time required by the plant to implement the operations for different privacy-preserving methods. The proposed affine masking strategy achieves better closed-loop performance compared to Methods 1 and 2. Both the proposed strategy and Method 2 utilize similar transformation-based techniques to mask actual information, and they are more computationally efficient compared to Method 1 which relies on complicated encryption and decryption procedures.

6. Conclusion

This paper developed an affine masking-based privacy-preserving cloud-based nonlinear MPC framework. We considered eavesdroppers and honest-but-curious adversaries who intend to infer the plant's system state and input and the ∞ -diversity with unbounded diameter privacy notion was adopted. A simple yet effective affine transformation mechanism was designed to enable privacy preservation without affecting the MPC calculation. Furthermore, the proposed method was successfully extended to output-feedback MPC. Simulation results showed that by using the proposed method, the MPC problem

can be addressed without disclosing private information to the cloud.

One thing we would like to note is that although the models are transformed in the cloud MPC implementations, one can show that the current privacy preservation scheme cannot protect the poles/zeros of the linearized system. Our future work will enhance the privacy scheme to address this issue. We will also extend this framework for systems with uncertainties (e.g., robust and stochastic MPCs), explore other metrics for privacy definition, and analyze its resilience/vulnerability to different attackers and side-knowledge.

References

Alexandru, A. B., Morari, M., & Pappas, G. J. (2018). Cloud-based MPC with encrypted data. In *Proceedings of the IEEE conference on decision and control* (pp. 5014–5019).

Allenspach, M., & Ducard, G. J. J. (2021). Nonlinear model predictive control and guidance for a propeller-tilting hybrid unmanned air vehicle. *Automatica*, 132, Article 109790.

Bemporad, A., Bernardini, D., Long, R., & Verdejo, J. (2018). *Model predictive control of turbocharged gasoline engines for mass production: Technical report*, SAE Technical Paper.

Corser, G. P., Fu, H., & Banihani, A. (2016). Evaluating location privacy in vehicular communications and applications. *IEEE Transactions on Intelligent Transportation Systems*, 17(9), 2658–2667.

Darup, M. S., & Jager, T. (2019). Encrypted cloud-based control using secret sharing with one-time pads. In *Proceedings of the IEEE conference on decision and control* (pp. 7215–7221).

Darup, M. S., Redder, A., & Quevedo, D. E. (2018). Encrypted cloud-based MPC for linear systems with input constraints. *IFAC-PapersOnLine*, 51(20), 535–542.

Darup, M. S., Redder, A., Shames, I., Farokhi, F., & Quevedo, D. E. (2018). Towards encrypted MPC for linear constrained systems. *IEEE Control Systems Letters*, 2(2), 195–200.

Dotzer, F. (2005). Privacy issues in vehicular ad hoc networks. In *Proceedings of international workshop on privacy enhancing technologies* (pp. 197–209).

Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4), 211–407.

Grossman, R. L. (2009). The case for cloud computing. *IT Professional*, 11(2), 23–27.

Huang, Z., Mitra, S., & Dullerud, G. (2012). Differentially private iterative synchronous consensus. In *Proceedings of the ACM workshop on privacy in the electronic society* (pp. 81–90).

Hubaux, J. P., Capkun, S., & Luo, J. (2004). The security and privacy of smart vehicles. *IEEE Security & Privacy*, 2(3), 49–55.

Khalil, H. K. (2002). *Nonlinear systems*. Upper Saddle River, NJ: Prentice-Hall.

Li, N., Girard, A., & Kolmanovsky, I. (2019). Stochastic predictive control for partially observable Markov decision processes with time-joint chance constraints and application to autonomous vehicle control. *Journal of Dynamic Systems, Measurement, and Control*, 141(7), Article 071007.

Li, Z., Kolmanovsky, I. V., Atkins, E. M., Lu, J., Filev, D. P., & Bai, Y. (2017). Road disturbance estimation and cloud-aided comfort-based route planning. *IEEE Transactions on Cybernetics*, 47(11), 3879–3891.

Li, Z., Kolmanovsky, I., Atkins, E., Lu, J., Filev, D., & Michelini, J. (2014). Cloud aided semi-active suspension control. In *Proceeding of the IEEE symposium on computational intelligence in vehicles and transportation systems* (pp. 76–83).

Li, Z., Kolmanovsky, I., Atkins, E., Lu, J., Filev, D. P., & Michelini, J. (2016). Road risk modeling and cloud-aided safety-based route planning. *IEEE Transactions on Cybernetics*, 46(11), 2473–2483.

Li, N., Zhang, K., Li, Z., Srivastava, V., & Yin, X. (2023). Cloud-assisted nonlinear model predictive control for finite-duration tasks. *IEEE Transactions on Automatic Control*, 68(9), 5287–5300.

Li, M., Shi, Y., & Liu, X. (2016). Distributed MPC of aggregated heterogeneous thermostatically controlled loads in smart grid. *IEEE Transactions on Industrial Electronics*, 63(2), 1120–1129.

Lu, Y., & Zhu, M. (2020). On privacy preserving data release of linear dynamic networks. *Automatica*, 115, Article 108839.

Machanavajjhala, A., Kifer, D., Gehrke, J., & Venkatasubramanian, M. (2007). ℓ -diversity: Privacy beyond k -anonymity. *ACM Transactions on Knowledge Discovery from Data*, 1(1), 3–14.

Mangasarian, O. L. (2011). Privacy-preserving linear programming. *Optimization Letters*, 5(1), 165–172.

Mayne, D. Q. (2014). Model predictive control: Recent developments and future promise. *Automatica*, 50, 2967–2986.

McDaniel, P., & McLaughlin, S. (2009). Security and privacy challenges in the smart grid. *IEEE Security & Privacy*, 7(3), 75–77.

Munteanu, A., Muradore, R., Merro, M., & Fiorini, P. (2018). On cyber-physical attacks in bilateral teleoperation systems: An experimental analysis. In *Proceedings of the IEEE industrial cyber-physical systems* (pp. 159–166).

Naseri, A. M., Lucia, W., & Youssef, A. (2022). A privacy preserving solution for cloud-enabled set-theoretic model predictive control. In *Proceedings of the European control conference* (pp. 894–899). IEEE.

National Highway Traffic Safety Administration (2014). Vehicle-to-vehicle communications: Readiness of V2V technology for application.

Nekouei, E., Tanaka, T., Skoglund, M., & Johansson, K. H. (2019). Information-theoretic approaches to privacy in estimation and control. *Annual Reviews in Control*, 47, 412–422.

Ozatay, E., Onori, S., Wollaeger, J., Ozguner, U., Rizzoni, G., Filev, D., Michelini, J., & Di Cairano, S. (2014). Cloud-based velocity profile optimization for everyday driving: A dynamic-programming-based solution. *IEEE Transactions on Intelligent Transportation Systems*, 15(6), 2491–2505.

Petit, J., & Shladover, S. E. (2015). Potential cyberattacks on automated vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 16(2), 546–556.

Raffo, G. V., Ortega, M. G., & Rubio, F. R. (2010). An integral predictive/nonlinear H_∞ control structure for a quadrotor helicopter. *Automatica*, 46(1), 29–39.

Rawlings, J. B., Mayne, D. Q., & Diehl, M. (2017). *Model predictive control: theory, computation, and design*. Nob Hill Publishing.

Sankar, L., Rajagopalan, S. R., & Poor, H. V. (2013). Utility-privacy trade-offs in databases: An information-theoretic approach. *IEEE Transactions on Information Forensics and Security*, 8(6), 838–852.

Schlör, S., Hertneck, M., Wildhagen, S., & Allgöwer, F. (2021). Multi-party computation enables secure polynomial control based solely on secret-sharing. In *Proceedings of the IEEE conference on decision and control* (pp. 4882–4887).

Schlüter, N., & Darup, M. S. (2020). Encrypted explicit MPC based on two-party computation and convex controller decomposition. In *Proceedings of the IEEE conference on decision and control* (pp. 5469–5476).

Shamir, A. (1979). How to share a secret. *Communications of the ACM*, 22(11), 612–613.

Sultangazin, A., & Tabuada, P. (2021). Symmetries and isomorphisms for privacy in control over the cloud. *IEEE Transactions on Automatic Control*, 66(2), 538–549.

Wang, C., Ren, K., & Wang, J. (2011). Secure and practical outsourcing of linear programming in cloud computing. In *Proceedings of IEEE INFOCOM* (pp. 820–828).

Weeraddana, P. C., Athanasiou, G., Fischione, C., & Baras, J. S. (2013). Per-se privacy preserving solution methods based on optimization. In *Proceedings of the IEEE conference on decision and control* (pp. 206–211).

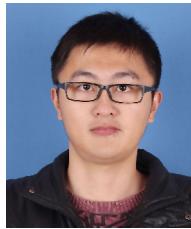
Weeraddana, P. C., & Fischione, C. (2017). On the privacy of optimization. *IFAC-PapersOnLine*, 50(1), 9502–9508.

Xu, Y., Deng, G., Zhang, T., Qiu, H., & Bao, Y. (2021). Novel denial-of-service attacks against cloud-based multi-robot systems. *Information Sciences*, 576, 329–344.

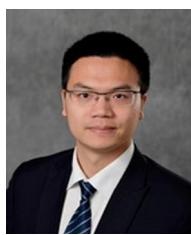
Xu, Z., & Zhu, Q. (2015). Secure and resilient control design for cloud enabled networked control systems. In *Proceedings of the first ACM workshop on cyber-physical systems-security and/or privacy* (pp. 31–42).

Xu, Z., & Zhu, Q. (2017). Secure and practical output feedback control for cloud-enabled cyber-physical systems. In *Proceedings of the IEEE conference on communications and network security* (pp. 416–420).

Xue, M., Wang, W., & Roy, S. (2014). Security concepts for the dynamics of autonomous vehicle networks. *Automatica*, 50(3), 852–857.



Kaixiang Zhang received the B.E. degree in automation from Xiamen University, Xiamen, China, in 2014, and the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2019. From 2017 to 2018, he was a visiting doctoral student with the National Institute for Research in Computer Science and Automation, Rennes, France. He was a research associate with the Department of Mechanical Engineering at Michigan State University, where he is currently a fixed-term Assistant Professor. His research interests include visual servoing, robotics, and control theory and application.



Zhaojian Li is a Red Cedar Associate Professor in the Department of Mechanical Engineering at Michigan State University. He obtained M.S. (2013) and Ph.D. (2015) in aerospace engineering (flight dynamics and control) at the University of Michigan, Ann Arbor. Dr. Li worked as an algorithm engineer at General Motors from January 2016 to July 2017. His research interests include learning-based control, nonlinear and complex systems, and robotics and automated vehicles. He is the author of more than 50 top journal articles and several patents. He is currently the associate editor for *IEEE Transactions on Intelligent Vehicles*, *Journal of Evolving Systems*, *American Control Conference*, and *ASME Dynamics and Control Conference*. He is a senior member of IEEE and a recipient of the NSF CAREER award.



Yongqiang Wang was born in Shandong, China. He received the B.S. degree in electrical engineering and automation, the B.S. degree in computer science and technology from Xi'an Jiaotong University, Xi'an, Shaanxi, China, in 2004, and the M.Sc. and Ph.D. degrees in control science and engineering from Tsinghua University, Beijing, China, in 2009. From 2007 to 2008, he was with the University of Duisburg-Essen, Germany, as a visiting student. He was a Project Scientist at the University of California, Santa Barbara before joining Clemson University, SC, USA, where he is currently an Associate Professor. His current research interests include distributed control, optimization, and learning, with emphasis on privacy protection. He currently serves as an associate editor for *IEEE Transactions on Automatic Control* and *IEEE Transactions on Control of Network Systems*.



Nan Li received the B.E. degree in vehicle engineering from Tongji University, Shanghai, China, in 2014, the M.S. degree in mechanical engineering, the M.S. degree in mathematics, and the Ph.D. degree in aerospace engineering from the University of Michigan, Ann Arbor, MI, USA, in 2016, 2020, and 2021, respectively. He is now a Professor with the School of Automotive Studies, Tongji University, Shanghai, China. Prior to joining Tongji University, he was a Postdoctoral Research Fellow with the Department of Aerospace Engineering at the University of Michigan, Ann Arbor, MI, USA, from 2021 to 2022, and an Assistant Professor with the Department of Aerospace Engineering at Auburn University, AL, USA, from 2022 to 2024. His research interests include optimal and predictive control, data-driven and learning-based control, multi-agent systems, and automotive and aerospace applications. He has authored more than 80 articles.