This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2017.2779473, IEEE Control Systems Letters

Towards encrypted MPC for linear constrained systems

Moritz Schulze Darup, Adrian Redder, Iman Shames, Farhad Farokhi, and Daniel Quevedo

***Abstract*—We present an encrypted model predictive control (MPC) scheme for linear constrained systems. More precisely, we show that homomorphic encryption can be used to realize a secure and private cloud-based evaluation of an MPC if the corresponding piecewise affine control law is explicitly given.**

***Index Terms*—Networked control systems, security, privacy, homomorphic encryption, predictive control for linear systems**

I. Introduction

The internet of things (IoT), cyber-physical systems, and smart grids show that the requirements for control engineering are changing. Already today, a variety of control schemes are implemented on network or cloud services. While these architectures offer many opportunities, they also increase the risk of cyberattacks. Hence, much research has focused on safe implementations of control schemes within clouds (see, e.g., [1], [2]). In this context, homomorphic encryption (HE) seems to be a promising tool (see [3], [4]). HE allows simple arithmetic operations (e.g., sums or multiplications) to be evaluated on ciphertext. It thus opens the opportunity to implement simple control schemes (e.g., linear state feedback) in an encrypted form. The general setup is as follows. First, state measurements are encrypted at the sensor. Associated ciphertexts are then transmitted to the cloud, where an encrypted control action is computed and sent to the actuator. Upon reception, the computed input is decrypted and applied to the system. Since data is never decrypted during transmission or in the cloud, eavesdropping and some other attacks are effectively prevented.

In this paper, we introduce an encrypted model predictive control (MPC) scheme for linear systems with state and input constraints. Thus, we extend the concepts from [3] and [4] to constrained systems, which is the main contribution of the paper. Technically, our approach exploits the fact that an MPC law is piecewise affine (for polytopic constraints and a quadratic performance criterion, see [5]). Under the assumption that this control law can be explicitly computed, we show that a secure and private cloud-based implementation of the controller can be realized using HE. In other words, we present an encrypted explicit MPC scheme.

In principle, encryption should not influence the performance of a controlled system since it only involves a temporary transformation of data. It turns out, however, that encryption requires quantization and it is well-known that quantization may affect the stability of controlled systems [6]. Thus, it is important to carefully address the effect of encryption on the system's stability (similar to [4]). We here use robust MPC (as in [7]) to compensate quantization errors and to guarantee robust stability. While alternative methods for handling quantization exist (see, e.g., [8]), we will show that the proposed robust MPC scheme additionally allows one to choose a suitable resolution and range of the quantization (i.e., a suitable fixed-point format). In this way, we provide another extension to [3] and [4].

M. Schulze Darup, A. Redder, and D. Quevedo are with the Automatic Control Group, Department of Electrical Engineering, Universität Paderborn, Germany. E-mails: moritz.schulzedarup@rub.de, aredder@mail.upb.de, dquevedo@ieee.org. I. Shames and F. Farokhi are with the Control and Signal Processing Lab, Department of Electrical and Electronic Engineering, University of Melbourne, Australia. E-mails: iman.shames@unimelb.edu.au, farhad.farokhi@unimelb.edu.au.

The paper is organized as follows. We state notation in the remainder of this section and provide background on encryption and quantization in Section II. We then discuss the MPC design with quantization compensation in Section III. Our main result, i.e., the encrypted cloud-based evaluation of the resulting control law, is presented in Section IV. Finally, we illustrate the proposed control scheme with an example in Section V and state conclusions in Section VI.

Notation. We denote the sets of real, rational, integer, and natural numbers by $\mathbb{R}$, $\mathbb{Q}$, $\mathbb{Z}$, and $\mathbb{N}$, respectively. We further introduce $\mathbb{N}_{[i,k]} := \{j \in \mathbb{N} \,|\, i \leq j \leq k\}$ and $\mathbb{Z}_M$ as the set of integers modulo M for $M \in \mathbb{N}$. For two sets $\mathcal{C}, \mathcal{T} \subset \mathbb{R}^n$, the operations $\mathcal{C} \oplus \mathcal{T}$ and $\mathcal{C} \ominus \mathcal{T}$ refer to the Minkowski addition and the Pontryagin difference, respectively. We frequently use the matrix norms $\|K\|_1$, $\|K\|_\infty$, and

$$\|K\|_{\max} := \max_{j \in \mathbb{N}_{[1,m]}} \max_{k \in \mathbb{N}_{[1,n]}} |K_{jk}|$$

for some matrix $K \in \mathbb{R}^{m \times n}$. The squared vector norm $\|\xi\|_P^2$ is understood as $\xi^\intercal P \xi$ for some $\xi \in \mathbb{R}^n$ and some positive definite matrix $P \in \mathbb{R}^{n \times n}$.

II. Background on encryption and quantization

A. Paillier cryptosystem

In this paper, we use the Paillier cryptosystem (PCS, see [9]) to encrypt data. As most other cryptosystems, the PCS acts on a subset of the set of integers. The cardinality of this subset depends on the chosen key for the encryption. The key generation requires the selection of two large prime numbers p_1 and p_2 such that $\gcd(p_1 p_2, (p_1 - 1)(p_2 - 1)) = 1$, where $\gcd(i, j)$ refers to the greatest common divisor of $i, j \in \mathbb{N}$. For the simplified PCS used here, p_1 and p_2 are additionally required to be of the same order, i.e., $p_1, p_2 \in [2^{j-1}, 2^j]$ for some $j \in \mathbb{N}$. The public key then is $M = p_1 p_2$. The private key (which is only known to the entity that needs to decrypt the data) is computed as $\lambda := \operatorname{lcm}(p_1 - 1, p_2 - 1)$, where $\operatorname{lcm}(i, j)$ refers to the least common multiple of $i, j \in \mathbb{N}$. For the encryption of any plaintext $t \in \mathbb{Z}_M$, we now choose a random $r \in \mathbb{Z}_M^* := \{t \in \mathbb{Z}_M \,|\, \gcd(t, M) = 1\}$ and we construct the ciphertext as

$$E_M(t, r) := (M + 1)^t r^M \bmod M^2 \in \mathbb{Z}_{M^2}.$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2017.2779473, IEEE Control Systems Letters

For any ciphertext $c \in \mathbb{Z}_{M^2}$, the corresponding plaintext results from the decryption

$$D_M(c) := L_M(c^\lambda \bmod M^2)\,\mu \bmod M,$$

where $L_M(\gamma) := (\gamma-1)\backslash M$ (with "\" denoting integer division) and where $\mu := \lambda^{-1} \bmod M$ is the so-called modular multiplicative inverse. It can be shown that encryption followed by decryption provides an integer equivalent to the plaintext. More formally, for every $t \in \mathbb{Z}_M$, we have $D_M(E_M(t,r)) = t$ for all $r \in \mathbb{Z}_M^*$. The PCS is a (partial) HE since it allows the computations of sums in encrypted form. More precisely, for every $t_1, t_2 \in \mathbb{Z}_M$ such that $t_1+t_2 \in \mathbb{Z}_M$, we have

$$E_M(t_1,r_1)E_M(t_2,r_2) \bmod M^2 = E_M(t_1+t_2, r_1 r_2) \quad (1)$$

for all $r_1, r_2 \in \mathbb{Z}_M^*$ [9, Sect. 8]. As a consequence of this property, we can also compute a semi-encrypted product. In fact, for every $t_1, t_2 \in \mathbb{Z}_M$ such that $t_1\, t_2 \in \mathbb{Z}_M$, we obtain

$$E_M(t_1,r)^{t_2} \bmod M^2 = E_M(t_1 t_2, r^{t_2}) \quad (2)$$

for all $r \in \mathbb{Z}_M^*$. Both properties will be used to evaluate encrypted MPC laws in the remainder of the paper.

B. Quantization and fixed point arithmetic

As detailed further below, the proposed application of the PCS requires quantization of the state measurements and the controller matrices. To this end, we consider signed fixed-point numbers in base 2 of the form

$$\underbrace{\beta_l}_{\text{sign bit}}\ \underbrace{\beta_{l-1}\beta_{l-2}\dots\beta_{d+1}}_{\text{integer bits}} \cdot \underbrace{\beta_d\beta_{d-1}\dots\beta_1}_{\text{fractional bits}},$$

where l and d denote the numbers of total and fractional bits, respectively. Based on this format, all numbers in the set

$$\mathbb{Q}_{l,d} := \left\{ q \in \mathbb{Q} \;\middle|\; \exists \beta \in \{0,1\}^l \colon q = -2^{l-d-1}\beta_l + \sum_{i=1}^{l-1} 2^{i-d-1}\beta_i \right\}.$$

can be described. Obviously, $\mathbb{Q}_{l,d}$ contains rational numbers between -2^{l-d-1} and $2^{l-d-1}-2^{-d}$ separated from each other by the resolution 2^{-d}. Now, the system states and controller matrices will, in general, be real-valued. We describe their quantization using the (surjective) mapping $g_{l,d} : \mathbb{R} \to \mathbb{Q}_{l,d}$ with

$$g_{l,d}(a) := \arg\min_{q \in \mathbb{Q}_{l,d}} |a-q|, \quad (3)$$

where the smallest absolute q is selected in case two optimizers exist. To prepare the application of the PCS, we further introduce the bijective mapping $f_{l,d} : \mathbb{Q}_{l,d} \to \mathbb{Z}_{2^l}$ as

$$f_{l,d}(q) := 2^d q \bmod 2^l. \quad (4)$$

Data will later be encrypted from and decrypted to the integer set $\mathbb{Z}_{2^l}$. However, the computed control actions will be values in $\mathbb{Q}_{l,d}$. We shall use the inverse $f_{l,d}^{-1} : \mathbb{Z}_{2^l} \to \mathbb{Q}_{l,d}$ with

$$f_{l,d}^{-1}(t) := \frac{1}{2^d} \cdot \begin{cases} t-2^l & \text{if} \quad t \geq 2^{l-1}, \\ t & \text{otherwise} \end{cases}$$

to reconstruct the inputs. See [4, Prop. 1]) for a proof that $q = f_{l,d}^{-1}\left(f_{l,d}(q)\right)$ for every $q \in \mathbb{Q}_{l,d}$. Within the control law evaluation, we will later have to compute sums q_1+q_2 and products $q_1 q_2$ with $q_1, q_2 \in \mathbb{Q}_{l_1,d}$ Clearly, the sums and products will also be rational numbers. However, they might not be contained in $\mathbb{Q}_{l_1,d}$. Nevertheless, it is easy to see that there exists an $l_2 \geq l_1$ such that q_1+q_2 and $2^d q_1 q_2$ are contained in $\mathbb{Q}_{l_2,d} \supseteq \mathbb{Q}_{l_1,d}$. We then find

$$\begin{aligned} f_{l_2,d}(q_1+q_2) &= 2^d(q_1+q_2) \bmod 2^{l_2} \\ &= \left(2^d q_1 \bmod 2^{l_2} + 2^d q_2 \bmod 2^{l_2}\right) \bmod 2^{l_2} \\ &= (t_1+t_2) \bmod 2^{l_2}, \end{aligned} \quad (5)$$

where $t_1 := f_{l_2,d}(q_1)$ and $t_2 := f_{l_2,d}(q_2)$ (cf. [4, Prop. 2]). In other words, a mapped summation can roughly be interpreted as a sum in $\mathbb{Z}_{2^{l_2}}$. A similar observation holds for the product (cf. [4, Prop. 4]) in terms of

$$f_{l_2,d}(2^d q_1 q_2) = 2^d \left(2^d q_1 q_2\right) \bmod 2^{l_2} = t_1 t_2 \bmod 2^{l_2}. \quad (6)$$

III. MPC DESIGN WITH QUANTIZATION COMPENSATION

Our goal is to control linear systems of the form

$$x(k+1) = A\,x(k) + B\,u(k), \quad x(0) = x_0 \quad (7)$$

with polytopic state and input constraints

$$x(k) \in \mathcal{X} \quad \text{and} \quad u(k) \in \mathcal{U} \quad \text{for every } k \in \mathbb{N}. \quad (8)$$

Throughout the paper, we assume that the pair (A,B) (with $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$) is stabilizable and that the set $\mathcal{X} \subset \mathbb{R}^n$ is a convex and compact polytope with the origin as an interior point. We further assume that $\mathcal{U} \subset \mathbb{R}^m$ is a hyperrectangle, i.e.,

$$\mathcal{U} = \left\{ u \in \mathbb{R}^m \,|\, |u_j| \leq \overline{u}_j,\ \forall j \in \mathbb{N}_{[1,m]} \right\} \quad (9)$$

for some $\overline{u} \in \mathbb{R}^m$ with positive entries. We will show that the quantizations of the state measurements and controller matrices can be understood as artificial disturbances to the system (7) (see [10, Sect. IV] for a similar concept). We counteract those disturbances by applying a robust MPC scheme as in [7]. For the moment, we therefore consider the modified system

$$x(k+1) = A\,x(k) + B\,u(k) + w(k), \quad x(0) = x_0 \quad (10)$$

with additive disturbances from the set

$$\mathcal{W} := \{ w \in \mathbb{R}^n \,|\, \|w\|_\infty \leq \overline{\omega} \}, \quad (11)$$

where $\overline{\omega} > 0$ can be understood as a design parameter. Roughly speaking, larger $\overline{\omega}$ allow larger quantization effects to be compensated. It turns out, however, that $\overline{\omega}$ cannot be chosen arbitrarily large. In fact, we have to guarantee that a matrix $K^* \in \mathbb{R}^{m\times n}$ and a nonempty set $\mathcal{R}$ exist such that $A+BK^*$ is Schur and such that

$$(A+BK^*)\,\mathcal{R} \oplus \mathcal{W} \subseteq \mathcal{R}. \quad (12)$$

Based on such a robust positively invariant (RPI) set $\mathcal{R}$ (see, e.g., [11]), we design the robust MPC scheme according to [7] and obtain the optimal control problem

$$\begin{aligned} V(x) = \min_{\substack{\tilde{x}(0),\dots,\tilde{x}(N) \\ \tilde{u}(0),\dots,\tilde{u}(N-1)}} & \|\tilde{x}(N)\|_P^2 + \sum_{k=0}^{N-1} \|\tilde{x}(k)\|_Q^2 + \|\tilde{u}(k)\|_R^2 \quad (13) \\ \text{s.t. } & x - \tilde{x}(0) \in \mathcal{R}, \\ & \tilde{x}(k+1) = A\,\tilde{x}(k) + B\tilde{u}(k), \quad \forall k \in \mathbb{N}_{[0,N-1]} \\ & \tilde{x}(k) \in \mathcal{X} \ominus \mathcal{R}, \quad \forall k \in \mathbb{N}_{[0,N-1]} \\ & \tilde{u}(k) \in \mathcal{U} \ominus K^*\mathcal{R}, \quad \forall k \in \mathbb{N}_{[0,N-1]} \\ & \tilde{x}(N) \in \mathcal{T} \end{aligned}$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2017.2779473, IEEE Control Systems Letters

that is solved in every time step for the current state x (i.e., $x(k)$). As in classical MPC, P, Q, and R denote weighting matrices and $\mathcal{T}$ is a terminal set. In contrast to classical MPC, we consider tightened state and input constraints and we allow the initial state $\tilde{x}(0)$ of the predicted state trajectory to differ from x. Based on the optimized state and input sequences $\tilde{x}^*(0), \dots, \tilde{x}^*(N)$ and $\tilde{u}^*(0), \dots, \tilde{u}^*(N-1)$, we apply the control law $\rho : \mathcal{F} \to \mathcal{U}$ with

$$\rho(x) := \tilde{u}^*(0) + K^* \left(x - \tilde{x}^*(0)\right), \tag{14}$$

where $\mathcal{F} \subseteq \mathcal{X}$ is the set of all x such that (13) is feasible. Under the assumption that Q and R are positive definite and for a sensible choice of P and $\mathcal{T}$ (see [7, Sect. 3] or Sect. V below for details), we can guarantee that the system

$$x(k+1) = A\,x(k) + B\rho(x(k)) + w(k), \quad x(0) = x_0 \tag{15}$$

is robustly stable for all $w(k) \in \mathcal{W}$ in the sense that the system will stay in $\mathcal{F}$ and converge to $\mathcal{R}$ for all initial states $x_0 \in \mathcal{F}$ [7, Thm. 1].

Now, in case $\mathcal{R}$ and $\mathcal{T}$ are chosen as polytopes (which can be done according to [11] and [12]), it is well-known that (13) can be rewritten as a strictly convex quadratic program (QP) of the form

$$V(x) = \min_{Z} \frac{1}{2} Z^\intercal H Z \quad \text{s.t.} \quad GZ \leq F\,x + d \tag{16}$$

with $Z^\intercal = (\,\tilde{x}^\intercal(0)\ \tilde{u}^\intercal(0)\ \dots\ \tilde{u}^\intercal(N-1)\,)$. As a consequence, the optimal decision variable $Z^* \in \mathbb{R}^{n+Nm}$ is a piecewise affine function of the current state x (see, e.g., [5]). This observation implies that the control law (14) is also piecewise affine. More precisely, (14) can be written as

$$\rho(x) = \begin{cases} K_1\,x + b_1 & \text{if} \quad x \in \mathcal{P}_1, \\ \vdots & \\ K_s\,x + b_s & \text{if} \quad x \in \mathcal{P}_s, \end{cases} \tag{17}$$

where the sets $\mathcal{P}_1, \dots, \mathcal{P}_s$ are convex polytopes with pairwise disjoint interiors and where $\bigcup_{i=1}^{s} \mathcal{P}_i = \mathcal{F}$. For systems with many states or for MPC schemes with long prediction horizons N, it is usually not possible to explicitly identify the structure (17) (with a reasonable numerical effort). However, for small systems with short prediction horizons, the controller matrices K_i and b_i as well as the polytopes $\mathcal{P}_i$ can be computed within seconds. Here, we assume that (17) can be computed offline and exploited online.

We will later use properties (1) and (2) to realize an encrypted evaluation of the affine control laws $K_i\,x + b_i$. Clearly, in order to apply the encryption E_M, we need to map the controller matrices and the current state to the set $\mathbb{Z}_M$. As a preparation, we consider the approximated control law

$$\hat{\rho}(x) = \begin{cases} \widehat{K}_1\hat{x} + \hat{b}_1 & \text{if} \quad x \in \mathcal{P}_1, \\ \vdots & \\ \widehat{K}_s\hat{x} + \hat{b}_s & \text{if} \quad x \in \mathcal{P}_s, \end{cases} \tag{18}$$

where $\widehat{K}_i$, $\hat{b}_i$ and $\hat{x}$ result from the element-wise quantization of K_i and b_i (offline) and x (online) via $g_{l_1,d}$. The quantized values can later be mapped to $\mathbb{Z}_M$ using (4). Two features of (18) are important. First, the quantization may lead to the problem that $\hat{u} := \hat{\rho}(x) \notin \mathcal{U}$ for some $x \in \mathcal{F}$. In order to guarantee the required input constraint satisfaction, we will additionally apply the saturation function $\sigma : \mathbb{R}^m \to \mathcal{U}$ with

$$\sigma(\hat{u}) := \arg\min_{u \in \mathcal{U}} \|\hat{u} - u\|_2^2. \tag{19}$$

Second, as detailed in Section IV, the selection of the controller segment i in (18) is carried out based on the original state x (and not $\hat{x}$). We next focus on the application of the approximated and saturated control law to the original system (7). Thus, we study the dynamics of the system

$$x(k+1) = A\,x(k) + B\,\sigma(\hat{\rho}\,(x(k))), \quad x(0) = x_0. \tag{20}$$

The following theorem states that, for a sensible design of the quantization in terms of l_1 and d, the dynamics (20) can be interpreted as a special instance of the disturbed system (15).

Theorem 1: Let $x_0 \in \mathcal{F}$ and let l_1 and d be such that

$$\max\left\{\|K_i\|_{\max}, \|b_i\|_\infty, \|x\|_\infty\right\} \leq 2^{l_1-d-1} - 2^{-d-1}, \tag{21}$$

$$\|K_i\|_1 + \|x\|_1 + n\,2^{-d-1} + 1 \leq \overline{\omega}\,\|B\|_\infty^{-1}\,2^{d+1} \tag{22}$$

for every $i \in \mathbb{N}_{[1,s]}$ and every $x \in \mathcal{F}$. Then, there exists a sequence $w(0), w(1), \dots \in \mathcal{W}$ such that the state trajectories of the systems (15) and (20) are equal.

Proof: We initially show that

$$\|B(\sigma(\hat{\rho}(x)) - \rho(x))\|_\infty \leq \overline{\omega}, \tag{23}$$

for every $x \in \mathcal{F}$. To this end, we overestimate the right-hand side in (23) by $\|B\|_\infty \|\sigma(\hat{\rho}(x)) - \rho(x)\|_\infty$. Now, since $\rho(x) \in \mathcal{U}$ holds by construction, we have

$$\|\sigma(\hat{\rho}(x)) - \rho(x)\|_\infty \leq \|\hat{\rho}(x) - \rho(x)\|_\infty$$

according to Lemma 3 in the appendix. As a consequence, (23) holds if

$$\|\hat{\rho}(x) - \rho(x)\|_\infty \leq \|B\|_\infty^{-1}\,\overline{\omega}. \tag{24}$$

Without loss of generality, we assume that x is contained in $\mathcal{P}_i$ and that $\hat{x}$ is the quantization of x based on $g_{l_1,d}$. Thus,

$$\|\hat{\rho}(x) - \rho(x)\|_\infty = \|\widehat{K}_i\hat{x} + \hat{b}_i - K_i x - b_i\|_\infty. \tag{25}$$

Clearly, (21) implies that

$$\max\left\{\|\Delta K_i\|_{\max}, \|\Delta b_i\|_\infty, \|\Delta x\|_\infty\right\} \leq 2^{-d-1},$$

where $\Delta K_i := \widehat{K}_i - K_i$, $\Delta b_i := \hat{b}_i - b_i$, and $\Delta x := \hat{x} - x$. Hence, we find

$$\begin{aligned} &\|\widehat{K}_i\hat{x} + \hat{b}_i - K_i x - b_i\|_\infty \\ &\leq \|\Delta K_i\,x + K_i\,\Delta x + \Delta K_i\,\Delta x + \Delta b_i\|_\infty \\ &\leq \|\Delta K_i\,x\|_\infty + \|K_i\,\Delta x\|_\infty + \|\Delta K_i\,\Delta x\|_\infty + \|\Delta b_i\|_\infty \\ &\leq \|x\|_1\,2^{-d-1} + \|K_i\|_1\,2^{-d-1} + n\,(2^{-d-1})^2 + 2^{-d-1}. \end{aligned} \tag{26}$$

Combining (25) and (26) with condition (22) yields (24).

We are now ready to prove the statement in the theorem. For $k = 0$, both trajectories start at the same state $x(0) = x_0 \in \mathcal{F}$. Next, based on (15) and (20), we easily verify that the disturbance $w(0) := B\,(\sigma(\hat{\rho}(x(0))) - \rho(x(0)))$ leads to equal states at time $k = 1$. Moreover, we find $w(0) \in \mathcal{W}$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2017.2779473, IEEE Control Systems Letters

by definition of $\mathcal{W}$ in (11) and according to (23). Having $w(0) \in \mathcal{W}$ further implies $x(1) \in \mathcal{F}$ according to [7, Thm. 1]. By induction, we then conclude that the disturbances

$$w(k) := B\left(\sigma(\hat{\rho}(x(k))) - \rho(x(k))\right)$$

lead to equal trajectories for all $k \in \mathbb{N}$. ∎

Theorem 1 implies that system (20) inherits all stability guarantees of system (15). More precisely, for a quantization satisfying (21) and (22), it is guaranteed that (20) stays in $\mathcal{F}$ and converges to $\mathcal{R}$ for every initial state $x_0 \in \mathcal{F}$.

IV. Encrypted control law evaluation

The implementantion of the quantized predictive controller requires to evaluate $\sigma(\hat{\rho}(x))$ for the current state x in every time step. After having identified the polytope i such that $x \in \mathcal{P}_i$, the control action results from computing

$$u = \sigma(\hat{u}) \quad \text{with} \quad \hat{u} := \hat{\rho}(x) = \widehat{K}_i \hat{x} + \hat{b}_i. \tag{27}$$

We state the following theorem to prepare a secured computation of $\hat{u}$ based on encrypted data.

Theorem 2: Let $x \in \mathcal{P}_i$, let conditions (21) and (22) be satisfied, and let l_2 and M be such that

$$(2^{l_1-1}-1)\left(n(2^{l_1-1}-1)+2^d\right)+1 \leq 2^{l_2-1} \quad \text{and} \tag{28}$$

$$2^{l_2} + n\left(2^{l_2}-1\right)^2 \leq M. \tag{29}$$

Then, the j-th component of $\hat{u}$ satisfies

$$\hat{u}_j = 2^{-d} f_{l_2,d}^{-1}\left(D_M\left(c_0 \prod_{k=1}^{n} c_k^{t_k} \bmod M^2\right) \bmod 2^{l_2}\right), \tag{30}$$

where

$$c_k := E_M(f_{l_2,d}(\hat{x}_k), r_k), \qquad \forall k \in \mathbb{N}_{[1,n]}, \tag{31}$$

$$t_k := f_{l_2,d}((\widehat{K}_i)_{jk})), \qquad \forall k \in \mathbb{N}_{[1,n]}, \tag{32}$$

$$c_0 := E_M(f_{l_2,d}(2^d(\hat{b}_i)_j), r_0) \tag{33}$$

for arbitrary $r_0, \ldots, r_n \in \mathbb{Z}_M^*$.

Proof: We obviously have

$$\hat{u}_j = 2^{-d} f_{l_2,d}^{-1}\left(f_{l_2,d}\left(2^d(\widehat{K}_i \hat{x} + \hat{b}_i)_j\right)\right)$$

under the assumption that

$$2^d(\widehat{K}_i \hat{x} + \hat{b}_i)_j \in \mathbb{Q}_{l_2,d}. \tag{34}$$

Since $\widehat{K}_i, \hat{b}_i, \hat{x} \in \mathbb{Q}_{l_1,d}$, it is easy to see that $2^d(\widehat{K}_i \hat{x} + \hat{b}_i)_j$ is an integer multiple of 2^{-d}. Thus, (34) holds if

$$\left|2^d(\widehat{K}_i \hat{x} + \hat{b}_i)_j\right| \leq 2^{l_2-d-1} - 2^{-d}. \tag{35}$$

Clearly, (21) implies

$$\left|2^d(\hat{b}_i)_j\right| \leq 2^d \|\hat{b}_i\|_\infty \leq 2^d(2^{l_1-d-1} - 2^{-d}) = 2^{l_1-1} - 1$$

and

$$\left|2^d(\widehat{K}_i \hat{x})_j\right| \leq 2^d n(2^{l_1-d-1} - 2^{-d})^2 \leq n2^{-d}(2^{l_1-1}-1)^2.$$

The two latter relations in comibination with (28) immediately prove (35). It remains to show that

$$f_{l_2,d}\left(2^d(\widehat{K}_i \hat{x} + \hat{b}_i)_j\right) = D_M(v_j) \bmod 2^{l_2},$$

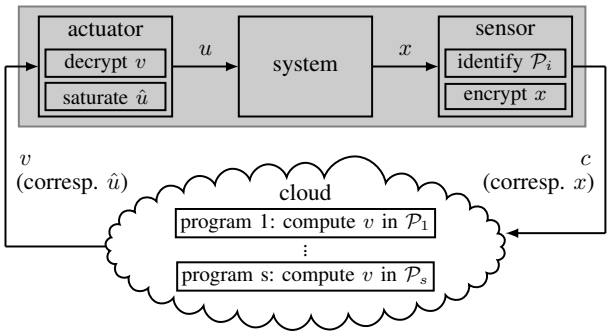

Fig. 1: Illustration of the encrypted cloud-based MPC scheme. Data is unencrypted only at the client (gray unit).

where

$$v_j := c_0 \prod_{k=1}^{n} c_k^{t_k} \bmod M^2 = c_0 \prod_{k=1}^{n} \left(c_k^{t_k} \bmod M^2\right) \bmod M^2. \tag{36}$$

Relation (36) can be rewritten using properties (1) and (2). In fact, under the assumption that $t_k f_{l_2,d}(\hat{x}_k) \in \mathbb{Z}_M$, we have

$$c_k^{t_k} \bmod M^2 = E_M(t_k f_{l_2,d}(\hat{x}_k), r_k^{t_k})$$

according to (2). Taking (1) into account and defining,

$$t_{\text{res}} := f_{l_2,d}(2^d(\hat{b}_i)_j) + \sum_{k=1}^{n} t_k f_{l_2,d}(\hat{x}_k) \quad \text{and} \quad r_{\text{res}} := r_0 \prod_{k=1}^{n} r_k^{t_k},$$

we further obtain $v_j = E_M(t_{\text{res}}, r_{\text{res}})$ under the assumption that $t_{\text{res}} \in \mathbb{Z}_M$. Since $f_{l_2,d}$ is mapping to $\mathbb{Z}_{2^{l_2}} = \mathbb{N}_{[0,2^{l_2}-1]}$, both assumptions are satisfied due to (29). We consequently find $D_M(v_j) = t_{\text{res}}$ and finally

$$\begin{aligned} t_{\text{res}} \bmod 2^{l_2} &= f_{l_2,d}(2^d(\hat{b}_i)_j) + \sum_{k=1}^{n} f_{l_2,d}(2^d(\widehat{K}_i)_{jk}\hat{x}_k) \bmod 2^{l_2} \\ &= f_{l_2,d}\left(2^d(\hat{b}_i)_j + \sum_{k=1}^{n} 2^d(\widehat{K}_i)_{jk}\hat{x}_k\right) \\ &= f_{l_2,d}\left(2^d(\widehat{K}_i \hat{x} + \hat{b}_i)_j\right), \end{aligned}$$

according to (5) and (6), which completes the proof. ∎

Theorem 2 is central for our approach. Most importantly, the right-hand side of (30) enables an encrypted cloud-based evaluation of the controller. The required computations at the sensor, in the cloud, and at the actuator are specified in the following and the resulting control scheme is illustrated in Figure 1. In this context, a proper operation of the controller is guaranteed by conditions (28) and (29) that regulate the mapping $f_{l_2,d}$ and the encryption E_M.

A. Computations at the sensor

At the sensor, we measure the current state x (i.e. $x(k)$) at every time step and we identify the polytope $\mathcal{P}_i$ such that $x \in \mathcal{P}_i$. In this context, binary search trees can be used for an efficient and easy to implement identification of the correct set $\mathcal{P}_i$ [13]. We then quantize x by applying the mapping $g_{l_1,d}$ element-wise. Afterwards, the quantization $\hat{x}$ is element-wise encrypted by evaluating c_k according to (31). The resulting

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2017.2779473, IEEE Control Systems Letters

cyphertexts $c_1, \dots, c_n \in \mathbb{Z}_{M^2}$ are then transmitted to the cloud. More precisely, corresponding to the identified region $\mathcal{P}_i$, we call the program i with the arguments $c_1, \dots, c_n$.

B. Computations in the cloud

In the cloud, s programs are uploaded offline corresponding to the s polytopic regions $\mathcal{P}_i$. Every program i computes an ecrypted version of the quantized inputs $\hat{u}_1, \dots, \hat{u}_m$ by partially evaluating (30). Let the function j in program i be responsible for evaluating $\hat{u}_j$ in $\mathcal{P}_i$. Each function j is equipped with n hard-coded parameters t_k as in (32) and the hard-coded cyphertext c_0 as in (33). In every time step, corresponding to the polytope $\mathcal{P}_i$ identified at the sensor, the program i is called with the encrypted state information $c_1, \dots, c_n$. Within the program, every function $j \in \mathbb{N}_{[1,m]}$ is called and the output v_j as in (36) is computed. Obviously, $v_j \in \mathbb{Z}_{M^2}$ is related to the encrypted input $\hat{u}_j \in \mathbb{Q}_{l_2,d}$. This can easily be seen from the decomposition (30). Finally, the ciphertexts $v_1, \dots, v_m$ are transmitted to the actuator.

C. Computations at the actuator

At the actuator, it remains to decrypt, transform, and rescale the ciphertexts $v_1, \dots, v_m$ to obtain $\tilde{u}_1, \dots, \tilde{u}_m$. According to (30), these computations are done by evaluating

$$\hat{u}_j = 2^{-d} f_{l_2,d}^{-1}\left(D_M\left(v_j\right) \bmod 2^{l_2}\right)$$

for every $j \in \mathbb{N}_{[1,m]}$. Finally, we have to guarantee input constraint satisfaction by evaluating $u = \sigma(\hat{u})$ as in (19). For the considered hyperrectangular set $\mathcal{U}$, this can be done according to (39) in the appendix. The resulting control action is then applied to the system.

V. Numerical example

We illustrate the proposed encrypted MPC scheme with a double integrator. The system matrices read

$$A = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \qquad \text{and} \qquad B = \begin{pmatrix} 0.5 \\ 1 \end{pmatrix}$$

and the constraints are $\mathcal{X} = \{x \in \mathbb{R}^2 \,|\, |x_1| \leq 25, |x_2| \leq 5\}$ and $\mathcal{U} = \{u \in \mathbb{R} \,|\, |u| \leq 1\}$. The weighting matrices $Q = I$ and $R = 0.01$, the set $\mathcal{W}$ with $\overline{\omega} = 0.1$, and the prediction horizon $N = 9$ are chosen as in [7]. We stress again that $\overline{\omega}$ is a design parameter that influences the quantization in terms of condition (22). Next, the terminal weighting matrix P is selected as the solution of the Riccati equation

$$A^\intercal(P - P\,B\,(R + B^\intercal P\,B)^{-1}B^\intercal P)\,A - P + Q = 0$$

and the stabilizing controller gain is chosen as

$$K^* = -(R + B^\intercal P\,B)^{-1}B^\intercal P\,A \approx \begin{pmatrix} -0.6609 & -1.3261 \end{pmatrix}.$$

We then use the method from [11] to compute an RPI set $\mathcal{R}$ satisfying condition (12) and obtain

$$\mathcal{R} = \operatorname{conv}\left\{ \begin{pmatrix} \mp 0.2536 \\ \pm 0.2519 \end{pmatrix}, \begin{pmatrix} \mp 0.0516 \\ \pm 0.2519 \end{pmatrix}, \begin{pmatrix} \pm 0.2536 \\ \mp 0.0499 \end{pmatrix} \right\}.$$

Finally, the terminal set $\mathcal{T}$ is chosen as

$$\mathcal{T} := \{x \in \mathbb{R}^n \,|\, (A + BK^*)^k x \in \mathcal{C},\, \forall k \in \mathbb{N}\},$$

where $\mathcal{C} := \{x \in \mathcal{X} \ominus \mathcal{R} \,|\, K^* x \in \mathcal{U} \ominus K^*\mathcal{R}\}$. Note that $\mathcal{C}$ is polytopic since $\mathcal{X}$, $\mathcal{U}$, and $\mathcal{R}$ are polytopes. As a consequence, $\mathcal{T}$ is polytopic as apparent from [12, Thm. 4.1]. We now formulate the robust MPC scheme (13) as a QP of the form (16) and solve it explicitely according to [5] using the multiparametric toolbox [14].

The paper focuses on the encrypted evaluation of the explicit MPC law within a cloud. We consequently provide a detailed implementation of the procedure from Section IV that allows to reproduce our numerical results. We first design the quantization $g_{l_1,d}$ such that the conditions (21) and (22) are satisfied. As a preparation, we compute

$$\max_{i \in \mathbb{N}_{[1,s]}} \|K_i\|_{\max} \approx 7.2737, \qquad \max_{i \in \mathbb{N}_{[1,s]}} \|K_i\|_1 \approx 7.9346,$$
$$\max_{x \in \mathcal{F}} \|x\|_\infty = 25, \qquad \max_{x \in \mathcal{F}} \|x\|_1 = 30,$$
$$\max_{i \in \mathbb{N}_{[1,s]}} \|b_i\|_\infty \approx 27.1896.$$

We then choose the smallest $d \in \mathbb{N}$ such that condition (22) holds (with $n = 2$) and obtain $d = 8$. Next, we identify the smallest $l_1 \in \mathbb{N}$ such that (21) is satisfied and find $l_1 = 14$. Afterwards, we design the mapping $f_{l_2,d}$ to $\mathbb{Z}_{2^{l_2}}$. In this context, the smallest $l_2 \in \mathbb{N}$ satisfying (28) is $l_2 = 29$. To prepare the encryption, it remains to choose suitable primes p_1 and p_2 such that $M = p_1 p_2$ satisfies (29). However, not only this inequality is important for the choice of p_1 and p_2. In fact, the encryption should also be safe which requires p_1 and p_2 to be very large (see [9, Sect. 6]). Here, for illustration purposes, we choose the relatively small primes $p_1 = 3320791807$ and $p_2 = 4075816727$ in $[2^{31}, 2^{32}]$. After computing the public key $M = p_1 p_2 = 13534938793855155689$ and the private key $\lambda = \operatorname{lcm}(p_1 - 1, p_2 - 1) = 1353493878645854715 6$, we are ready to apply our encrypted MPC scheme.

To illustrate the application, we consider the initial state

$$x_0 = \begin{pmatrix} -22.8125 - 2^{-d-1} & 3.125 + 2^{-d-1} \end{pmatrix}^\intercal. \tag{37}$$

At time step $k = 0$, the sensor measures the current state $x = x(0) = x_0$ and identifies the polytope $\mathcal{P}_i$ containing x. Afterwards, the state x is quantized using $g_{l_1,d}$ and we obtain $\hat{x} = \begin{pmatrix} -22.8125 & 3.125 \end{pmatrix}^\intercal$. Next, the mapping via $f_{l_2,d}$ and the encryption with the random numbers $r_1 = 7437696318725102472$ and $r_2 = 3862701628878047690$ results in the cyphertexts

$$c_1 = 940996016497879821937772819111425361 92,$$
$$c_2 = 259991830887541919928432649378858227 55$$

as in (31) that are transmitted to the cloud.

In the cloud, the program i (corresponding to the polytope $\mathcal{P}_i$) is called. In plaintext, it implements a quantized version of the locally affine control law

$$K_i\,x + b_i \approx \begin{pmatrix} -0.0996196 & -0.7586155 \end{pmatrix} x + 1.0699914.$$

To prepare the encrypted computation of the control law, we first compute $\widehat{K}_i = \begin{pmatrix} -0.1015625 & -0.7578125 \end{pmatrix}$ and $\hat{b}_i = 1.0703125$ offline by applying $g_{l_1,d}$ element-wise. Next, we offline calculate the plaintexts $t_1 = f_{l_2,d}((\widehat{K}_i)_1) = 536870886$ and $t_2 = f_{l_2,d}((\widehat{K}_i)_2) = 536870718$ and

$$c_0 = 430811085649344052132599179109952304 68$$

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LCSYS.2017.2779473, IEEE Control Systems Letters

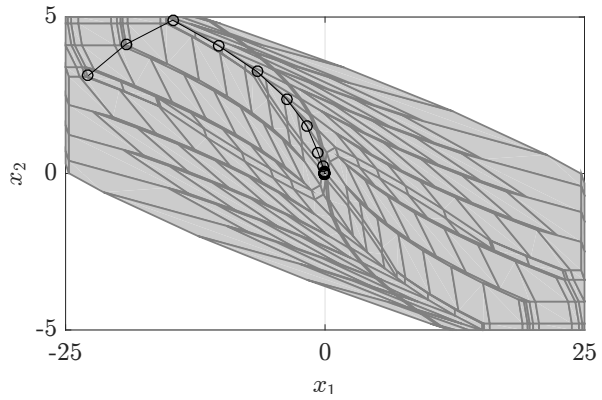

Fig. 2: Illustration of a closed-loop trajectory emanating from x_0 as in (37). The union of the 379 gray polytopes $\mathcal{P}_i$ shows the feasible set $\mathcal{F}$ of the controller.

as in (33) for $r_0 = 6170687951983498748$. These numbers are hardcoded in the function $j = 1$ within program i. For this example, due to $m = 1$, each of the s programs in the cloud contains only one function. In case $m > 1$, the previous steps are repeated for every component j of the control law. Now, online, we compute v_1 according to (36), obtain

$$v_1 = 2868333910292456070456011003298802218 0,$$

and transmit it to the actuator.

At the actuator, we first evaluate

$$\hat{u} = 2^{-d} f_{l_2,d}^{-1}\left(D_M(v_1) \bmod 2^{l_2}\right) = 1.01904296875.$$

Clearly, $\hat{u}$ violates the input constraints. However, after evaluating the saturation function, we obtain $u = \sigma(\hat{u}) = 1 \in \mathcal{U}$, which is finally applied to the system. The procedure is then repeated for every step $k \in \mathbb{N}$. The resulting trajectory is illustrated in Figure 2. As desired, the system is steered towards the origin or, more precisely, to the RPI set $\mathcal{R}$.

VI. Conclusions

We presented an encrypted implementation of an MPC scheme for linear systems with polytopic constraints. The encrypted evaluation of the control law builds on the explicit computation of the piecewise affine control law combined with homomorphic properties of the Paillier cryptosystem. A proper operation of the encrypted controller is guaranteed by a set of conditions (namely (21), (22), (28), and (29)) that regulate the underlying quantization and encryption.

The approach can be understood as a starting point for further research on encrypted MPC. In particular, two features of the proposed method need further attention. The first feature is the selection of the current controller segment i that is assumed to be carried out at the sensor. While this assumption simplifies the implementation of the encrypted controller, it also increases the computational load at the sensor. Moreover, the interceptable call of program i in the cloud leaks some information that might enable cyberattacks. Without giving details, we note that this leak can be closed at the expense of further computations at the sensor. Ideally, the bulk of computations should, however, be carried out in the cloud. Unfortunately, it is an open problem how (or if) the selection of the segment i can be realized within the cloud. The second feature that needs attention is the explicit computation of the MPC law. As discussed in Section III, this approach is only practical for systems with a few states and short prediction horizons. The extension to more complex systems therefore requires to focus on encrypted online optimization strategies similar to [15]. Clearly, this leads to a different setup. Nevertheless, we are confident that some features of the proposed scheme, e.g., the quantization compensation via robust MPC, are portable.

References

[1] A. A. Cárdenas, S. Amin, and S. Sastry, "Research challenges for the security of control systems," in *Proceedings of the 3rd Conference on hot topics in security*, pp. 1–6, 2008.

[2] Y. Mo and B. Sinopoli, "Secure control against replay attacks," in *Proc. of the 47th Annual Allerton Conference*, pp. 911–918, 2009.

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

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

[5] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, vol. 38, pp. 3–20, 2002.

[6] D. F. Delchamps, "Stabilizing a linear system with quantized state feedback," *IEEE Trans. Autom. Control*, vol. 35, pp. 916–924, 1990.

[7] D. Q. Mayne, M. M. Seron, and S. V. Raković, "Robust model predictive control of constrained linear systems with bounded disturbances," *Automatica*, vol. 41, pp. 219–224, 2005.

[8] D. E. Quevedo, G. C. Goodwin, and J. A. De Doná, "Finite constraint set receding horizon quadratic control," *Int. J. Robust Nonlinear Control*, vol. 14, pp. 355–377, 2004.

[9] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Advances in Cryptology - Eurocrypt '99: International Conference on the Theory and Application of Cryptographic Techniques* (J. Stern, ed.), vol. 1592, pp. 223–238, Springer, 1999.

[10] R. P. Aguilera and D. E. Quevedo, "Stability analysis of quadratic mpc with a discrete input alphabet," *IEEE Trans. Autom. Control*, vol. 58, pp. 3190–3196, 2013.

[11] S. V. Raković, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne, "Invariant approximations of the minimal robust positively invariant set," *IEEE Trans. Autom. Control*, vol. 50, pp. 406–410, 2005.

[12] E. G. Gilbert and K. T. Tan, "Linear systems with state and control constraints: The theory and application of maximal output admissible sets," *IEEE Trans. Autom. Control*, vol. 36, pp. 1008–1020, 1991.

[13] P. Tøndel, T. A. Johansen, and A. Bemporad, "Computation and approximation of piecewise affine control laws via binary search trees," in *Proc. of the 41st Conference on Decision and Control*, pp. 3144–3149, 2002.

[14] M. Kvasnica, P. Grieder, M. Baotić, and M. Morari, "Multi-parametric toolbox (MPT)," in *Proc. of 7th International Workshop on Hybrid Systems - Computation and Control*, pp. 448–462, 2004.

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

Appendix

Lemma 3: Let $\mathcal{U}$ and σ be defined as in (9) and (19), respectively. Further, let $\hat{u} \in \mathbb{R}^m$ and $u \in \mathcal{U}$. Then

$$\|\sigma(\hat{u}) - u\|_\infty \leq \|\hat{u} - u\|_\infty. \tag{38}$$

Proof: Due to the hyperrectangular shape of $\mathcal{U}$, we have

$$\sigma_j(\hat{u}) = \operatorname{sign}(\hat{u}_j)\min\{|\hat{u}_j|, \overline{u}_j\} \tag{39}$$

for every $j \in \mathbb{N}_{[1,m]}$. Taking into account that $|u_j| \leq \overline{u}_j$ due to $u \in \mathcal{U}$, we easily prove that $|\sigma_j(\hat{u}) - u_j| \leq |\hat{u}_j - u_j|$ for every $j \in \mathbb{N}_{[1,m]}$, which immediately leads to (38). ■