Secure and Private Implementation of Dynamic Controllers Using Semi-Homomorphic Encryption

Carlos Murguia, Farhad Farokhi, and Iman Shames

***Abstract*—This paper presents a secure and private implementation of linear time-invariant dynamic controllers using Paillier's encryption, a semi-homomorphic encryption method. To avoid overflow or underflow within the encryption domain, the state of the controller is reset periodically. A control design approach is presented to ensure stability and optimize performance of the closed-loop system with encrypted controller.**

I. Introduction

Internet of Things (IoT) has brought opportunities for flexibility of deployment and efficiency improvements. However, it threatens security and privacy of individuals and businesses as IoT devices, by design, share their information for processing over the cloud. This information can be secured from adversaries over the network by using encrypted communication channels [1]. This approach, although effective and necessary, does not address vulnerability of the data on servers running cloud-computing services. These services themselves can use the data for targeted advertisement or can be hacked for malicious purposes. Therefore, there is a need for a more secure methodology that addresses the security and privacy of data while being processed.

Thankfully secure cloud computing is possible with the use of homomorphic encryption methods – encryption methods that allow computation over plain data by performing appropriate computations on the encrypted data [2]–[4]. The use of homomorphic encryption allows a controller to be remotely realised without needing to openly sharing private and sensitive data (and consenting to its use in an unencrypted manner). This paper specifically discusses secure and private implementation of linear time-invariant dynamic controllers with the aid of the Paillier's encryption [3], a semi-homomorphic encryption method.

The use of homomorphic encryption for secure control has been studied previously [5]–[12]. However, all these studies consider static controllers. This is because, when dealing with dynamical control laws (with an encrypted memory/state that must be maintained remotely), the number of bits required for representing the state of the controller grows linearly with the number of iterations. This renders the memory useless after a few iterations due to an overflow or an underflow (i.e., number of fractional bits required for representing a number becomes larger than the number of fractional bits in the fixed-point number basis). In fact, using rough calculations, it can be seen that for a system with sampling time of 10 milliseconds, 16 bits quantized controller parameters and measurements, and within an encryption space of 2048 bits[1], the state of the controller becomes incorrect after roughly 1.2 seconds due to an overflow or underflow. The unstabilizing effect of restricting the memory of controllers to finite rings is illustrated in Section IV for the key length of 2048 bits using a controller that can easily stabilize a batch chemical reactor in the absence of encryption.

There are multiple ways to deal with this issue:
1) We should decrypt the state of the encrypted controller, project it into the desired set of fixed-point rational numbers, and encrypt it again. To avoid this issue, the encrypted state can be sent to a trust third-party (e.g., an IoT device) to be decrypted, rounded, encrypted, and transmitted back. This adds unnecessary communication overhead and overburdens the computational units of the IoT device. Furthermore, by decrypting the state, the risk of a security breach increases.
2) We should restrict the controller parameters so that the state of the dynamic controller remains within the set of fixed-point rational numbers. This could make the problem of designing the controller into a mixed-integer optimization problem, which is computationally exhaustive. However, a robust control approach can be taken to ensure that converting non-integer controllers to integer ones does not ruin stability [13]. An alternative approach with a promising prospect was recently pursued in [14] based on coordinate transformation and controller reencryption to implement linear dynamic controllers over an infinite horizon.
3) We should reset the controller, i.e., the state of the controller is set to a publicly known number (e.g., zero) periodically. In this case, the controller must be redesigned to ensure stability/performance, which this paper shows to remain a tractable optimisation problem. This is the approach chosen by the current paper.

Here, we only focus on encrypting the outputs of the system and the state of the controller. This is because the parameters of the controller are often not sensitive in practice. For instance, in autonomous vehicles, location and velocity are sensitive as they reveal private information about users,

C. Murguia is with the Department of Mechanical Engineering, Eindhoven University of Technology, The Netherlands. e-mail: C.G.Murguia@tue.nl.

F. Farokhi and I. Shames are with the Department of Electrical and Electronic Engineering at the University of Melbourne, Australia. e-mails: {ffarokhi,ishames}@unimelb.edu.au.

When working on this paper, F. Farokhi was also affiliated with CSIRO's Data61. He is thankful for their support during that time.

The work of I. Shames was funded by NATO Science for Peace and Security (SPS) PROGRAMME SPS.SFP G5479. The work of F. Farokhi was funded by McKenzie Fellowship and Melbourne School of Engineering.

[1]Encryption keys with the length of 2048 bits is recommended by National Institute of Standards and Technology (NIST) for data over 2016-2030; see https://www.keylength.com/en/4/.

e.g., home/work address and travel habits, while the controller parameters are implicitly related to dynamics of the vehicle.

Resetting controllers have been previously studied in [15]–[21]. However, the synthesis approach in this paper is more general than those studies and further it is designed to accommodate challenges associated with the implementation of dynamical controllers over the cipher space. Particularly, majority of existing work on reset controllers focus on state dependent triggers. Due to the nature of our problem, where the controller cannot access to the unencrypted state, those results are not applicable. Along the same lines, since we always have to reset the controller to the same state regardless of the state of the plant, the existing results for switched systems seem to be not applicable.

The rest of the paper is organized as follows. Preliminary materials on homomorphic encryption are presented in Section II. The design and implementation of the controller is discussed in Section III. Finally, numerical results are presented in Section IV and the paper is concluded in Section V.

II. Preliminary Material

In this paper, a tuple $(\mathbb{P}, \mathbb{C}, \mathbb{K}, \mathfrak{E}, \mathfrak{D})$ denotes a public key encryption scheme, where $\mathbb{P}$ is the set of plaintexts, $\mathbb{C}$ is the set of ciphertexts, $\mathbb{K}$ is the set of keys, $\mathfrak{E}$ is the encryption algorithm, and $\mathfrak{D}$ is the decryption algorithm. Each $\kappa = (\kappa_p, \kappa_s) \in \mathbb{K}$ is composed of a public key κ_p (which is shared with and used by everyone for encrypting plaintexts) and a private key κ_s (which is maintained only by the trusted parties for decryption). Algorithms $\mathfrak{E}$ and $\mathfrak{D}$ are publicly known while the keys, which set the parameters of these algorithms, are generated and used in each case. The use of the term "algorithm", instead of mapping or function, is due to the presence of random[2] elements in the encryption procedure possibly resulting in one plaintext being mapped to multiple ciphertexts. A necessary requirement for the encryption scheme is to be invertible, i.e., $\mathfrak{D}(\mathfrak{E}(x, \kappa_p), \kappa_p, \kappa_s) = x$ for all $x \in \mathbb{P}$ given $\kappa = (\kappa_p, \kappa_s) \in \mathbb{K}$.

Definition 1 (Homomorphic Property) *Assume there exist operators $\circ$ and $\diamond$ such that $(\mathbb{P}, \circ)$ and $(\mathbb{C}, \diamond)$ form groups. A public key encryption $(\mathbb{P}, \mathbb{C}, \mathbb{K}, \mathfrak{E}, \mathfrak{D})$ is called called homomorphic if $\mathfrak{D}(\mathfrak{E}(x_1, \kappa_p) \diamond \mathfrak{E}(x_1, \kappa_p), \kappa_p, \kappa_s) = x_1 \circ x_2$ for all $x_1, x_2 \in \mathbb{P}$ and $\kappa \in \mathbb{K}$.*

Throughout this paper, $|\mathbb{A}|$ denotes the cardinality of any set $\mathbb{A}$. Further, we define the notation $\mathbb{Z}_q := \{0, \dots, q-1\} = \{n \bmod q : \forall n \in \mathbb{Z}\}$ for all positive integers $q \in \mathbb{N}$. In this paper, we assume that $\mathbb{P} = \mathbb{Z}_{n_p}$ and $\mathbb{C} = \mathbb{Z}_{n_c}$ with $n_p = |\mathbb{P}|$ and $n_c = |\mathbb{C}|$. A public key encryption $(\mathbb{P}, \mathbb{C}, \mathbb{K}, \mathfrak{E}, \mathfrak{D})$ is additively homomorphic if there exists an operator $\diamond$ such that Definition 1 is satisfied when the operator $\circ$ is defined as $x_1 \circ x_2 := (x_1 + x_2) \bmod n_p$ for all $x_1, x_2 \in \mathbb{P}$. For additively homomorphic schemes, in this paper, the notation $\oplus$ is used to denote the equivalent operator in the ciphertext domain ($\diamond$ in the definition above). Similarly, a public key encryption is multiplicatively homomorphic if there exists an operator $\diamond$ such that Definition 1 is satisfied with $\circ$ defined as $x_1 \circ x_2 := (x_1 x_2) \bmod n_p$ for all $x_1, x_2 \in \mathbb{P}$. If a public key encryption is both additively and multiplicatively homomorphic, it is fully homomorphic but, if only one of these conditions is satisfied, it is semi-homomorphic. Homomorphism shows there exist operations over ciphertexts that can generate encrypted versions of sumed or multiplied plaintexts without the need of decrypting their corresponding cuphertexts. An example of additively homomorphic encryption scheme is the Paillier's encryption method [3]. ElGamal is an example of multiplicatively homomorphic encryption schemes [4]. Recently, several fully homomorphic encryption methods have been also developed, see, e.g., [2].

Now, we define semantic security, borrowed from [22]. A key $\kappa = (\kappa_p, \kappa_s) \in \mathbb{K}$ is randomly generated. A probabilistic polynomial time-bounded adversary proposes $x_1, x_2 \in \mathbb{P}$. The agent chooses x at random from $\{x_1, x_2\}$ with equal probability, encrypts x according to $y = \mathfrak{E}(x, \kappa_p)$, and sends y to the adversary (along with the public key κ_p). The adversary produces x', which is an estimate of x based on all the avialable information (everything except κ_s, i.e., x_1, x_2, y, $\mathfrak{E}$, $\mathfrak{D}$, κ_p). The adversary's advantage (in comparison to that of a pure random number generator) is given by $\mathrm{Adv}(|\mathbb{K}|) := |\mathbb{P}\{x = x'\} - 1/2|$. The public key encryption $(\mathbb{P}, \mathbb{C}, \mathbb{K}, \mathfrak{E}, \mathfrak{D})$ is semantically secure (alternatively known as indistinguishability under chosen plaintext attack) if Adv is negligible[3].

In this paper, the results are presented for the Paillier's encryption method. It is noteworthy that the Paillier's encryption method is semantically secure under the *Decisional Composite Residuosity Assumption*, i.e., it is "hard" to decide whether there exists $y \in \mathbb{Z}_{N^2}$ such that $x = y^N \bmod N$ for $N \in \mathbb{Z}$ and $x \in \mathbb{Z}_{N^2}$. More information regarding the assumption can be found in [3], [23]. This can be used to establish the security of the proposed framework.

The Paillier's encryption scheme is as follows. First the public and private keys are generated. To do so, large prime numbers p and q are selected randomly and independently of each other such that $\gcd(pq, (1-p)(1-q)) = 1$, where $\gcd(a, b)$ refers to the greatest common divisor of integers a and b. The public key (which is shared with all the parties and is used for encryption) is $\kappa_p = pq$. The private key (which is only available to the entity that needs to decrypt the data) is $\kappa_s = (\lambda, \mu)$ with $\lambda = \mathrm{lcm}(p-1, q-1)$ and $\mu = \lambda^{-1} \bmod \kappa_p$, where $\mathrm{lcm}(a, b)$ is the least common multiple of integers a and b. The ciphertext of plain message $x \in \mathbb{P} = \mathbb{Z}_{\kappa_p}$ is $\mathfrak{E}(x, \kappa_p) = (\kappa_p + 1)^x r^{\kappa_p} \bmod \kappa_p^2$, where r is randomly selected with uniform probability from $\mathbb{Z}^*_{\kappa_p} := \{x \in \mathbb{Z}_{\kappa_p} | \gcd(x, \kappa_p) = 1\}$. Finally, to decrypt any ciphertext $c \in \mathbb{C} = \mathbb{Z}_{\kappa_p^2}$, $\mathfrak{D}(c, \kappa_p, \kappa_s) = (L(c^\lambda \bmod \kappa_p^2)\mu) \bmod \kappa_p$, where $L(z) = (z-1)/\kappa_p$.

Proposition 1 [3] *1) For $r, r' \in \mathbb{Z}^*_{\kappa_p}$ and $t, t' \in \mathbb{P}$ such that $t + t' \in \mathbb{P}$, $\mathfrak{E}(t, \kappa_p)\mathfrak{E}(t', \kappa_p) \bmod \kappa_p^2 = \mathfrak{E}(t + t', \kappa_p)$; 2) For $r \in \mathbb{Z}^*_{\kappa_p}$ and $t, t' \in \mathbb{P}$ such that $tt' \in \mathbb{P}$, $\mathfrak{E}(t, \kappa_p)^{t'} \bmod \kappa_p^2 = \mathfrak{E}(t't, \kappa_p)$.*

[2]These random elements are replaced with pseudo-random ones when implementing encryption and decryption algorithms.

[3]A function $f : \mathbb{N} \to \mathbb{R}_{\geq 0}$ is called negligible if, for any $c \in \mathbb{N}$, there exists $n_c \in \mathbb{N}$ such that $f(n) \leq 1/n^c$ for all $n \geq n_c$.

Proposition 1 shows that the Paillier's encryption is a semi-homomorphic encryption scheme, i.e., algebraic manipulation of the plain data is possible without decryption using appropriate operations over the encrypted data. The Paillier's encryption is additively homomorphic with operator $\oplus$ being defined as $x_1 \oplus x_2 = (x_1 x_2) \bmod \kappa_p^2$ for all $x_1, x_2 \in \mathbb{C}$. Note that the Paillier's method is not multiplicatively homomorphic as t' in the identity $\mathfrak{E}(t,\kappa_p)^{t'} \bmod \kappa_p^2 = \mathfrak{E}(t't,\kappa_p)$ in Proposition 1 is not encrypted. Define $\triangle$ such that $x_1 \triangle x_2 = x_1^{x_2} \bmod N^2$ for all $x_1 \in \mathbb{C}$ and $x_2 \in \mathbb{P}$. Note that $\triangle$ is not an operator (in the mathematical sense) as its operands belong to two difference sets; it is just a mapping.

III. Dynamic Controller Implementation

Consider the discrete-time linear time invariant system

$$\mathcal{P}: \begin{cases} x[k+1] = Ax[k] + Bu[k], \quad x[0] = x_0, \\ y[k] = Cx[k], \end{cases} \tag{1}$$

with $k \in \mathbb{N}$, state $x[k] \in \mathbb{R}^{n_x}$, control input $u[k] \in \mathbb{R}^{n_u}$, and output $y[k] \in \mathbb{R}^{n_y}$. Many linear time-invariant systems cannot be stabilized by static output feedback controllers [24]–[26]. Therefore, dynamic output feedback controllers have been used for decades to stabilize system using only output measurements, e.g., standard Kalman-filter (or Luenberger observer) based linear regulators [27], and general dynamic output feedback controllers for quadratic performance [28]. System (1) is controlled by a dynamic output feedback controller of the form

$$\mathcal{C}: \begin{cases} x_c[k+1] = \begin{cases} A_c x_c[k] + B_c y[k], & (k+1) \bmod T \neq 0, \\ 0, & (k+1) \bmod T = 0, \end{cases} \\ \qquad u[k] = C_c x_c[k] + D_c y[k], \end{cases} \tag{2}$$

with controller state $x_c[k] \in \mathbb{R}^{n_c}$. It is assumed that the state of the controller resets every T time steps, i.e., $x_c[\ell T] = 0$ for all $\ell \in \mathbb{N}$. *This is because implementing encrypted controllers over an infinite horizon is impossible due to memory issues (by multiplication of fractional numbers, the number of bits required for representing fractional and integer parts grow).*

Remark 1 (Observer-Based Controller) *A class of dynamic output controllers are observers plus static feedback controllers. For implementing the observer, we have two options:*

- *Encrypted Observer at a Remote Location: We remark that when implementing the observer in an encrypted form, we would encounter the same issues as implementing any other dynamic output controller:* under/over flow of the state in the encrypted domain. *Therefore, we would have to implement a resetting policy for the observer too. In this case, however, we do not have the extra degrees of freedom of selecting all the parameters of the controller; we can only select the observer/feedback gains. This makes the analysis of the performance harder and the controller more conservative (surely by having more free parameters, the controller could perform better).*
- *Without Encryption at the Sensor: If there are more than one sensor, they need to select a trusted sensor for implementation of the observer. This sensor should have enough computational and communication capability for a few more multiplications and summations (for the static state feedback controller) to avoid the need for remote control using third-party distrusted servers (which place this case out of the scope of the paper).*

Combining the dynamics in (1) and (2) results in the augmented system:

$$z[k+1] = \begin{cases} F(\mathcal{P},\mathcal{C}) z[k], & (k+1) \bmod T \neq 0, \\ \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} F(\mathcal{P},\mathcal{C}) z[k], & (k+1) \bmod T = 0, \end{cases} \tag{3}$$

where $z[k] := \begin{bmatrix} x[k]^\top & x_c[k]^\top \end{bmatrix}^\top$ and

$$F(\mathcal{P},\mathcal{C}) := \begin{bmatrix} A + BD_cC & BC_c \\ B_cC & A_c \end{bmatrix}. \tag{4}$$

The following theorem provides a sufficient condition for the asymptotic stability of the origin of (1) in feedback with the resetting controller (2).

Theorem 1 *The closed-loop dynamics* (1)-(2) *is globally asymptotically stable if there exist* $P \in \mathbb{R}^{(n_c+n_x)\times(n_c+n_x)}$, $\varepsilon \in (0,1)$, $\mu \in [-1,0)$, $\delta \in [1,\infty)$, *and* $\epsilon \in (0,\infty)$ *satisfying:*

$$P \succ \epsilon I, \tag{5a}$$

$$F(\mathcal{P},\mathcal{C})^\top P F(\mathcal{P},\mathcal{C}) \preceq (1+\mu) P, \tag{5b}$$

$$F(\mathcal{P},\mathcal{C})^\top \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} P \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} F(\mathcal{P},\mathcal{C}) \preceq \delta P, \tag{5c}$$

$$\delta (1+\mu)^{T-1} < \varepsilon. \tag{5d}$$

Proof: The proofs are removed due to page limit and are presented in a technical report. See [29]. ∎

The following result provides a sufficient condition for the stabilizability of the system using the resetting controller.

Proposition 2 *If* $n_c \geq n_x$, (A,B) *is stabilizable, and* (A,C) *is detectable, there exist* $\mu = \mu^* \in [-1,0)$ *and* $\epsilon = \epsilon^* \in (0,\infty)$ *such that* (5a) *and* (5b) *are satisfied.*

Proof: See [29]. ∎

For $\mu^* \in [-1,0)$ and $\epsilon^* \in (0,\infty)$ in Proposition 2, the following problem can be solved to find the smallest resetting horizon T for the dynamical controller:

$$\min_{\substack{T \in \mathbb{N} \\ \varepsilon \in (0,1) \\ \delta \in [1,\infty)}} \min T, \tag{6a}$$

$$\text{s.t.} \quad P \succ \epsilon^* I, \quad \delta (1+\mu^*)^{T-1} < \varepsilon, \tag{6b}$$

$$F(\mathcal{P},\mathcal{C})^\top P F(\mathcal{P},\mathcal{C}) \preceq (1+\mu^*) P, \tag{6c}$$

$$F(\mathcal{P},\mathcal{C})^\top \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} P \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} F(\mathcal{P},\mathcal{C}) \preceq \delta P. \tag{6d}$$

Note that the conditions in Theorem 1, or optimization problem (6), are sufficient but not necessary. This is always the case when working with Lyapunov-based techniques for stability of dynamical systems [28], [30]. In the next subsection, we provide change of variables to cast these conditions as linear matrix inequalities that can be solved off-line only once and passed to the cloud for real-time control.

A. Synthesis of Resetting Controllers

In this subsection, we use appropriate change of variables to linearize the matrix inequalities in Theorem 1 without generating conservatism. We provide tools for designing full order ($n_c = n_x$) resetting controllers of the form (2) satisfying (5).

That is, we look for matrices (A_c, B_c, C_c, D_c) satisfying the inequalities in (5) for some positive definite $P \in \mathbb{R}^{2n_x \times 2n_x}$, $\mu \in [-1, 0)$, $\delta \in (0, \infty)$, $\varepsilon \in (0, 1)$, and $T \in \mathbb{N}$. Let $n_c = n_x$ and P be positive definite. Consider $F(\mathcal{P}, \mathcal{C})$ in (4) and define:

$$\tilde{F}(\mathcal{P}, \mathcal{C}) := \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} F(\mathcal{P}, \mathcal{C}) = \begin{bmatrix} A + BD_cC & BC_c \\ 0 & 0 \end{bmatrix}. \quad (7)$$

For simplicity of notation, in this subsection, $F(\mathcal{P}, \mathcal{C})$ and $\tilde{F}(\mathcal{P}, \mathcal{C})$ are denoted by F and $\tilde{F}$, respectively. Then, (5b) and (5c) can be written as

$$F^\top P F - (1+\mu)P \preceq 0, \quad \tilde{F}^\top P \tilde{F} - \delta P \preceq 0, \quad (8)$$

where 0 denotes the zero matrix of appropriate dimensions. Using properties of the Schur complement, inequalities (8) are fulfilled if and only if the following is satisfied:

$$\mathcal{L} := \begin{bmatrix} (1+\mu)P & F^\top P \\ PF & P \end{bmatrix} \succeq 0, \tilde{\mathcal{L}} := \begin{bmatrix} \delta P & \tilde{F}^\top P \\ P\tilde{F} & P \end{bmatrix} \succeq 0. \quad (9)$$

Note that the blocks PF and $P\tilde{F}$ are nonlinear functions of (P, A_c, B_c, C_c, D_c). In what follows, we propose a change of variables: $(P, A_c, B_c, C_c, D_c) \to \nu$, so that, in the new variables ν, we can obtain *affine* matrix inequalities equivalent to (9). In particular, for positive definite P and *nonlinear* matrix inequalities $\mathcal{L} \geq 0$ and $\tilde{\mathcal{L}} \geq 0$, we aim at finding two invertible matrices $\mathcal{Y}$ and $\mathcal{T}$, and variables ν such that the congruence transformations $P \to \mathcal{Y}^\top P \mathcal{Y}$, $\mathcal{L} \to \mathcal{T}^\top \mathcal{L} \mathcal{T}$, and $\tilde{\mathcal{L}} \to \mathcal{T}^\top \tilde{\mathcal{L}} \mathcal{T}$ lead to new Linear Matrix Inequalities (LMIs) $\mathcal{Y}^\top P \mathcal{Y} > 0$, $\mathcal{T}^\top \mathcal{L} \mathcal{T} \geq 0$, and $\mathcal{T}^\top \tilde{\mathcal{L}} \mathcal{T} \geq 0$ in the variables ν. Let P be positive definite and partitioned as follows:

$$P := \begin{bmatrix} X & U \\ U^\top & \tilde{X} \end{bmatrix}, \quad (10)$$

with $X, U, \tilde{X} \in \mathbb{R}^{n_x \times n_x}$ and positive definite $X, \tilde{X}$. Define

$$P^{-1} =: \begin{bmatrix} Y & V \\ V^\top & \tilde{Y} \end{bmatrix}, \ \mathcal{Y} := \begin{bmatrix} Y & I \\ V^\top & 0 \end{bmatrix}, \ \mathcal{Z} := \begin{bmatrix} I & 0 \\ X & U \end{bmatrix}. \quad (11)$$

Using block matrix inversion formulas, it can be verified that $YX + VU^\top = I$ and $YU + V\tilde{X} = 0$, which yields $\mathcal{Y}^\top P = \mathcal{Z}$. Then, $P \to \mathcal{Y}^\top P \mathcal{Y}$ takes the form:

$$\mathcal{Y}^\top P \mathcal{Y} = \mathcal{Z}\mathcal{Y} = \begin{bmatrix} Y & I \\ I & X \end{bmatrix} =: \mathbf{P}(\nu). \quad (12)$$

Define $\mathcal{T} := \text{diag}[\mathcal{Y}, \mathcal{Y}]$ with $\mathcal{Y}$ as introduced in (11). Then, the transformations $\mathcal{L} \to \mathcal{T}^\top \mathcal{L} \mathcal{T}$ and $\tilde{\mathcal{L}} \to \mathcal{T}^\top \tilde{\mathcal{L}} \mathcal{T}$ can be written as

$$\mathcal{T}^\top \mathcal{L} \mathcal{T} = \begin{bmatrix} (1+\mu)\mathbf{P}(\nu) & \mathcal{Y}^\top F^\top \mathcal{Z}^\top \\ \mathcal{Z} F \mathcal{Y} & \mathbf{P}(\nu) \end{bmatrix}, \quad (13)$$

$$\mathcal{T}^\top \tilde{\mathcal{L}} \mathcal{T} = \begin{bmatrix} \delta \mathbf{P}(\nu) & \mathcal{Y}^\top \tilde{F}^\top \mathcal{Z}^\top \\ \mathcal{Z} \tilde{F} \mathcal{Y} & \mathbf{P}(\nu) \end{bmatrix}. \quad (14)$$

Using the structure of F and $\tilde{F}$ and the change of variables:

$$\begin{pmatrix} K_1 - XAY & K_2 \\ K_3 & K_4 \end{pmatrix} := \begin{pmatrix} U & XB \\ 0 & I_{n_u} \end{pmatrix} \begin{pmatrix} A_c & B_c \\ C_c & D_c \end{pmatrix} \begin{pmatrix} V^\top & 0 \\ CY & I_{n_y} \end{pmatrix}, \quad (15)$$

the blocks $\mathcal{Z} F \mathcal{Y}$ and $\mathcal{Z} \tilde{F} \mathcal{Y}$ can be written as

$$\mathcal{Z} F \mathcal{Y} = \begin{bmatrix} AY + BK_3 & A + BK_4C \\ K_1 & XA + K_2C \end{bmatrix} =: \mathbf{F}(\nu), \quad (16)$$

$$\mathcal{Z} \tilde{F} \mathcal{Y} = \begin{bmatrix} AY + BK_3 & A + BK_4C \\ XBK_3 + XAY & XA + XBK_4C \end{bmatrix} =: \tilde{\mathbf{F}}(\nu). \quad (17)$$

Therefore, under $\mathcal{T}$ and the change of variables in (15), we can write $\mathcal{T}^\top \mathcal{L} \mathcal{T}$ and $\mathcal{T}^\top \tilde{\mathcal{L}} \mathcal{T}$ as follows:

$$\mathcal{T}^\top \mathcal{L} \mathcal{T} = \begin{bmatrix} (1+\mu)\mathbf{P}(\nu) & \mathbf{F}(\nu)^\top \\ \mathbf{F}(\nu) & \mathbf{P}(\nu) \end{bmatrix} =: \mathbf{L}(\nu), \quad (18)$$

$$\mathcal{T}^\top \tilde{\mathcal{L}} \mathcal{T} = \begin{bmatrix} \delta \mathbf{P}(\nu) & \tilde{\mathbf{F}}(\nu)^\top \\ \tilde{\mathbf{F}}(\nu) & \mathbf{P}(\nu) \end{bmatrix} =: \mathbf{S}(\nu), \quad (19)$$

with $\mathbf{P}(\nu), \mathbf{F}(\nu)$, and $\tilde{\mathbf{F}}(\nu)$ as defined in (12), (16), and (17), respectively. Therefore, the original matrix inequality, $\mathcal{L} \succeq 0$ defined in (9), that depends non-linearly on the decision variables (P, A_c, B_c, C_c, D_c) is transformed into a new inequality, $\mathbf{L}(\nu) \succeq 0$, that is an affine function of the variables ν. Note, however, that $\mathbf{S}(\nu) \succeq 0$ (the block $\tilde{\mathbf{F}}(\nu)$) is still nonlinear in the new variables ν. In the following lemma, we give a sufficient condition, in terms of an affine inequality $\tilde{\mathbf{L}}(\nu) \succeq 0$, for $\mathbf{S}(\nu)$ to be positive semidefinite.

Lemma 1 *Consider $\mathbf{P}(\nu)$ and $\mathbf{S}(\nu)$ defined in* (12) *and* (19)*, respectively. Define the matrices:*

$$\mathbf{R}(\nu) := \begin{pmatrix} AY + BK_3 & A + BK_4C \end{pmatrix}, \quad (20)$$

$$\tilde{\mathbf{L}}(\nu) := \begin{bmatrix} \delta \mathbf{P}(\nu) & \mathbf{R}(\nu)^\top \\ \mathbf{R}(\nu) & 2I_n - X \end{bmatrix}. \quad (21)$$

Then, $\tilde{\mathbf{L}}(\nu) \succeq 0 \Rightarrow \mathbf{S}(\nu) \succeq 0$.

Proof: See [29]. ■

Lemma 1 provides a sufficient condition, $\tilde{\mathbf{L}}(\nu) \succeq 0$, for the nonlinear matrix $\mathbf{S}(\nu)$ to be positive semidefinite. This $\tilde{\mathbf{L}}(\nu)$ is an affine function of ν. Note, however, that finding ν satisfying $(\mathbf{P}(\nu) \succ 0, \mathbf{L}(\nu) \succeq 0, \tilde{\mathbf{L}}(\nu) \succeq 0)$ might not be sufficient to guarantee the existence of (P, A_c, B_c, C_c, D_c) satisfying $(P \succ 0, \mathcal{L} \succeq 0, \tilde{\mathcal{L}} \succeq 0)$. For this to be true, matrices $\mathcal{Y}$ and $\mathcal{T}$ must be invertible so that the transformations $P \to \mathcal{Y}^\top P \mathcal{Y} = \mathbf{P}(\nu)$, $\mathcal{L} \to \mathcal{T}^\top \mathcal{L} \mathcal{T} = \mathbf{L}(\nu)$, and $\tilde{\mathcal{L}} \to \mathcal{T}^\top \tilde{\mathcal{L}} \mathcal{T} = \mathbf{S}(\nu)$ are congruence transformations; and ν must render the change of variables in (15) invertible.

Lemma 2 *Consider matrices $\mathcal{Y}$ and $\mathbf{P}(\nu)$ defined in* (11) *and* (12)*, respectively. Let (X, Y) be such that $\mathbf{P}(\nu) \succ 0$. Then, $\mathcal{Y}$ and $\mathcal{T} = diag(\mathcal{Y}, \mathcal{Y})$ are nonsingular and the change of variables in* (15) *is invertible.*

Proof: See [29]. ■

Therefore, by Lemma 2, if $\mathbf{P}(\nu) \succ 0$, the transformations $P \to \mathbf{P}(\nu)$, $\mathcal{L} \to \mathbf{L}(\nu)$, and $\tilde{\mathcal{L}} \to \mathbf{S}(\nu)$ are congruence transformations. The latter and the fact that (by Lemma 1) $\tilde{\mathbf{L}}(\nu) \succeq 0 \Rightarrow \mathbf{S}(\nu) \succeq 0$ imply that

$$\begin{cases} (\mathbf{P}(\nu) \succ 0, \ \mathbf{L}(\nu) \succeq 0, \ \text{and } \tilde{\mathbf{L}}(\nu) \succeq 0) \\ \Downarrow \\ (P \succ 0, \ \mathcal{L} \succeq 0, \ \text{and } \tilde{\mathcal{L}} \succeq 0), \end{cases} \quad (22)$$

for $P = \mathcal{Y}^{-\top} \mathbf{P}(\nu) \mathcal{Y}^{-1}$ and the controller matrices in

$$\begin{pmatrix} A_c & B_c \\ C_c & D_c \end{pmatrix} = \begin{pmatrix} U & XB \\ 0 & I_{n_u} \end{pmatrix}^{-1} \begin{pmatrix} K_1 - XAY & K_2 \\ K_3 & K_4 \end{pmatrix} \times \begin{pmatrix} V^\top & 0 \\ CY & I_{n_y} \end{pmatrix}^{-1}, \quad (23)$$

obtained by inverting (15). In the following lemma, we summarize the discussion presented above.

Lemma 3 *For given system matrices (A, B, C). If there exist matrices $\nu = (X, Y, K_1, K_2, K_3, K_4)$, $K_2 \in \mathbb{R}^{n_x \times n_u}$,*

$K_3 \in \mathbb{R}^{n_y \times n_x}$, $K_4 \in \mathbb{R}^{n_u \times n_y}$, $X, Y, K_1 \in \mathbb{R}^{n_x \times n_x}$ *satisfying* $\mathbf{P}(\nu) \succ 0$, $\mathbf{L}(\nu) \succeq 0$, *and* $\tilde{\mathbf{L}}(\nu) \succeq 0$ *with* $\mathbf{P}(\nu)$, $\mathbf{L}(\nu)$, *and* $\tilde{\mathbf{L}}(\nu)$ *as defined in* (12), (18), *and* (21), *respectively; then, there exist* (P, A_c, B_c, C_c, D_c) *satisfying* $P \succ 0$, $\mathcal{L} \succeq 0$, *and* $\tilde{\mathcal{L}} \succeq 0$ *with* P, $\mathcal{L}$, *and* $\tilde{\mathcal{L}}$ *as defined in* (9) *and* (10), *respectively. Moreover, for every* ν *such that* $\mathbf{P}(\nu) \succ 0$, $\mathbf{L}(\nu) \succeq 0$, *and* $\mathbf{L}(\nu) \succeq 0$, *the change of variables in* (15) *and matrix* $\mathcal{Y}$ *in* (11) *are invertible and the* (P, A_c, B_c, C_c, D_c) *obtained by inverting* (12) *and* (15) *are unique and satisfy the analysis inequalities* (8).

Proof: See [29]. ∎

By Lemma 3, the matrices (P, A_c, B_c, C_c, D_c) obtained by inverting (12) and (15) satisfy inequalities (8) (and thus also (5b) and (5c)). Moreover, because the reconstructed P is positive definite, inequality (5a) is satisfied with $\epsilon = \lambda_{\min}(P)$, where $\lambda_{\min}(P) \in \mathbb{R}_{>0}$ denotes the smallest eigenvalue of P. Next, we give the synthesis result corresponding to Theorem 1.

Theorem 2 *For given system matrices* (A, B, C) *and constants* $\varepsilon \in (0, 1)$, $\mu \in [-1, 0)$, $\delta \in [1, \infty)$, *and* $T \in \mathbb{N}$ *satisfying* (5d), *if there exist matrices* $\nu = (X, Y, K_1, K_2, K_3, K_4)$ *satisfying* $\mathbf{P}(\nu) \succ 0$, $\mathbf{L}(\nu) \succeq 0$, *and* $\tilde{\mathbf{L}}(\nu) \succeq 0$; *then,* $P = \mathcal{Y}^{-\top}\mathbf{P}(\nu)\mathcal{Y}$ *and the controller matrices in* (23) *satisfy the analysis inequalities* (5a)-(5c) *and thus render the closed-loop dynamics* (1)-(2) *asymptotically stable.*

Proof: See [29]. ∎

Controller Reconstruction. For given ν satisfying the synthesis inequalities ($\mathbf{P}(\nu) \succ 0, \mathbf{L}(\nu) \succeq 0$, $\tilde{\mathbf{L}}(\nu) \succeq 0$):

1) For given X and Y, compute via singular value decomposition a full rank factorization $VU^\top = I - YX$ with square and nonsingular V and U.
2) For given ν and invertible V and U, solve the system of equations $\mathcal{Y}^\top P \mathcal{T} = \mathbf{P}(\nu)$ and (15) to obtain the matrices (P, A_c, B_c, C_c, D_c).

Remark 2 *Note that* ε, μ, δ, *and* T, *in Theorem 2 must be fixed before looking for feasible solutions* ν *satisfying the synthesis LMIs:* $\mathbf{P}(\nu) \succ 0$, $\mathbf{L}(\nu) \succeq 0$, *and* $\tilde{\mathbf{L}}(\nu) \succeq 0$. *However, for any* $\mu \in [-1, 0)$ *and* $\delta \in [1, \infty)$, *there always exist* $\varepsilon \in (0, 1)$ *and* $T \in \mathbb{N}$ *satisfying* (5d). *Moreover, the synthesis LMIs depend on* ν, δ, *and* μ *but not on* ε *and* T. *Therefore, to find feasible controllers, we only have to fix* (μ, δ) *and look for* ν *satisfying the synthesis LMIs. The constants* (μ, δ) *are, in fact, variables of the synthesis problem; however, to linearize some of the constraints, we fix their value and search over* $\mu \in [-1, 0)$ *and* $\delta \in [1, \infty)$ *to find feasible* ν. *The latter increases the computations needed to find controllers; however, we can perform a bisection search over* $\delta \in [1, \infty)$ *and, because* $\mu \in [-1, 0)$ *(a bounded set), a grid search over* μ *to decrease the required computations.*

Finally, note that the characteristics (e.g., unstable poles) of the system in (1) make the feasibility of the design LMIs $\mathbf{P}(\nu) \succ 0$, $\mathbf{L}(\nu) \succeq 0$, and $\tilde{\mathbf{L}}(\nu) \succeq 0$ "easier or harder" for fixed resetting horizon T and constants ε, μ, δ. Exploring this dependence, in general, is an avenue for future research.

B. Dynamic Controller Implementation

In this subsection, we present the necessary transformations required for implementing encrypted dynamic control laws. Before stating the next result, we introduce some notation. Define $\|A\|_{\max} := \max_{i,j} |a_{ij}|$, where a_{ij} denotes the entry in i-th row and j-th column of matrix A, and $\mathbb{Q}(n, m) := \{b \,|\, b = -b_n 2^{n-m-1} + \sum_{i=1}^{n-1} 2^{i-m-1} b_i, b_i \in \{0, 1\}\, \forall i \in \{1, \ldots, n\}\}$. For any $x \in \mathbb{R}^q$ and $\mathbb{A} \subseteq \mathbb{R}^q$, let $\operatorname{proj}(x, \mathbb{A}) \in \arg\min_{x' \in \mathbb{A}} \|x' - x\|_\infty$ and $\operatorname{dist}(x, \mathbb{A}) := \min_{x' \in \mathbb{A}} \|x' - x\|_\infty$. The quantization of $x \in \mathbb{R}^q$ is $\operatorname{proj}(x, \mathbb{Q}(n, m)^q)$ and the quantization error is $\|\operatorname{proj}(x, \mathbb{Q}(n, m)^q) - x\|_\infty = \operatorname{dist}(x, \mathbb{Q}(n, m)^q)$. The quantization of $X \in \mathbb{R}^{p \times q}$ is defined as $\operatorname{proj}(x, \mathbb{Q}(n, m)^{p \times q}) \in \arg\min_{x' \in \mathbb{A}} \|x' - x\|_{\max}$ and the quantization error as $\|\operatorname{proj}(x, \mathbb{Q}(n, m)^{p \times q}) - x\|_{\max}$. Please refer to [6] for details about the quantization scheme.

Remark 3 (Floating-Point vs Fixed-Point Representation) *Instead of quantizing the data and model parameters by projection into equally-spaced points, capturing the fixed-pint representation, we can encrypt the significant digits and exponents separately. This results in a floating-point representation of the number in which the exponent signifies the location of the floating point. In this case, however, the number of digits required for storage of the significant digits potentially grows with each summation and multiplication. Floating-point processors continuously trim these numbers to keep within the storage limits. However, this is not possible in the ciphertext unless we decrypt the state of the controller every few iterations and trim it. This adds unnecessary communication overhead and overburdens the computational units of the IoT device. Furthermore, by decrypting the state, the risk of a privacy or security breach increases.*

Theorem 3 *Let*

$$\bar{A}_c = \operatorname{proj}(A_c, \mathbb{Q}(n, m)^{n_c \times n_c}), \tag{24a}$$
$$\bar{B}_c = \operatorname{proj}(B_c, \mathbb{Q}(n, m)^{n_c \times n_y}), \tag{24b}$$
$$\bar{C}_c = \operatorname{proj}(C_c, \mathbb{Q}(n, m)^{n_u \times n_x}), \tag{24c}$$
$$\bar{D}_c = \operatorname{proj}(D_c, \mathbb{Q}(n, m)^{n_u \times n_y}). \tag{24d}$$

Then, there exists $\bar{n} \geq \bar{m} > 0$ *such that* $F(\mathcal{P}, \mathcal{C})$ *satisfies* (5) *if and only if* $F(\mathcal{P}, \bar{\mathcal{C}})$, *where* $\bar{\mathcal{C}}$ *denotes the controller in* (2) *with quantized parameters* $\bar{A}_c$, $\bar{B}_c$, $\bar{C}_c$, *and* $\bar{D}_c$ *in* (24), *satisfies* (5) *with the same* P *for all* $n \geq \bar{n}$ *and* $m \geq \bar{m}$.

Proof: The proof follows from continuity of the eigenvalues. Note that, by increasing n and m, the quantization error decreases (actually, it tends to zero). ∎

In what follows, we discuss the implementation of quantized resetting controllers using homomorphic encryption schemes and quantized sensor measurements. The controller, in this case, is given by

$$\bar{\bar{\mathcal{C}}}: \begin{cases} x_c[k+1] = \begin{cases} \bar{A}_c x_c[k] + \bar{B}_c \bar{y}[k], & (k+1) \bmod T \neq 0, \\ 0, & (k+1) \bmod T = 0, \end{cases} \\ u[k] = \bar{C}_c x_c[k] + \bar{D}_c \bar{y}[k], \end{cases} \tag{25}$$

where $\bar{A}_c$, $\bar{B}_c$, $\bar{C}_c$, and $\bar{D}_c$ are defined in (24) and

$$\bar{y}[k] \in \underset{y \in \mathbb{Q}(n,m)^{n_y}}{\arg\min} \|y - y[k]\|_\infty. \tag{26}$$

The difference between $\bar{\bar{\mathcal{C}}}$ in (25) and $\bar{\mathcal{C}}$ in Theorem 3 is the quantization of the output measurements $y[k]$. The following standing assumption is made in this paper to ensure the stability of the closed-loop system.

Assumption 1 *$n \geq \bar{n}$ and $m \geq \bar{m}$ where $\bar{n}$ and $\bar{m}$ are given in Theorem 3.*

The following theorem proves the stability of the system $\mathcal{P}$ with the quantized resetting controller $\bar{\bar{\mathcal{C}}}$. Note that, for any $r \in \mathbb{R}_{\geq 0}$, $\mathbb{B}(r) := \{x \,|\, \|x\|_2^2 \leq r\}$.

Theorem 4 *Under Assumption 1, if there exist $\varepsilon \in (0,1)$, $\mu \in [-1,0)$, $\delta \in [1,\infty)$, $T \in \mathbb{N}$, and $\epsilon \in (0,\infty)$ such that inequalities in* (5) *are satisfied and*

$$n > \log_2 \left(\frac{\lambda_{\max}(C^\top C)}{\epsilon} x_0^\top \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} P \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} x_0 \right) + 1; \tag{27}$$

then, the system dynamics (1) *with the quantized resetting controller in* (25) *is stable and, for some constant*[4] *$\varrho > 0$, $\lim_{k\to\infty} \operatorname{dist}(x[k], \mathbb{B}(\varrho 2^{-m})) = 0$.*

Proof: See [29]. ∎

Theorem 4 implies that the state of the system converges to a vicinity of the origin (instead of the origin itself) due to quantization effects. The volume of the this area can be arbitrarily reduced by increasing m and thus the performance of the system can be arbitrarily improved.

Lemma 4 *For the resetting quantized controller in* (25)*, $x_c[k] \in \mathbb{Q}((n_c+1)(k \bmod T - 1) + n_y + n(k \bmod T + 1), m(k \bmod T + 1))^{n_c}$ and $u_c[k] \in \mathbb{Q}((n_c+1)(k \bmod T) + n_y + n(k \bmod T + 2), m(k \bmod T + 2))^{n_u}$.*

Proof: See [29]. ∎

Using the change of variables:

$$\tilde{A}_c = (2^m \bar{A}_c) \bmod 2^{\tilde{n}}, \tag{28a}$$
$$\tilde{B}_c[k] = (2^{m(k \bmod T+1)} \bar{B}_c) \bmod 2^{\tilde{n}}, \tag{28b}$$
$$\tilde{C}_c = (2^m \bar{C}_c) \bmod 2^{\tilde{n}}, \tag{28c}$$
$$\tilde{D}_c[k] = (2^{m(k \bmod T+1)} \bar{D}_c) \bmod 2^{\tilde{n}}, \tag{28d}$$
$$\tilde{x}_c[k] = (2^{m(k \bmod T+1)} \bar{x}_c[k]) \bmod 2^{\tilde{n}}, \tag{28e}$$
$$\tilde{y}[k] = (2^{m(k \bmod T+1)} \bar{y}[k]) \bmod 2^{\tilde{n}}, \tag{28f}$$
$$\tilde{u}[k] = (2^{m(k \bmod T+2)} \bar{u}[k]) \bmod 2^{\tilde{n}}, \tag{28g}$$

with $\tilde{n} > (n_c+1)T + n_u + n(T+2)$, the resetting quantized controller in (25) can be rewritten as

$$\tilde{\mathcal{C}}: \begin{cases} \tilde{x}_c[k+1] = \begin{cases} \tilde{A}_c \tilde{x}_c[k] + \tilde{B}_c[k] \tilde{y}[k], & (k+1) \bmod T \neq 0, \\ 0, & (k+1) \bmod T = 0, \end{cases} \\ \tilde{u}[k] = \tilde{C}_c \tilde{x}_c[k] + \tilde{D}_c[k] \tilde{y}[k]. \end{cases} \tag{29}$$

Note that, by Lemma 4, $\tilde{A}_c, \tilde{B}_c, \tilde{C}_c, \tilde{D}_c, \tilde{x}_c, \tilde{y}, \tilde{u}$ are *positive integers*. This is useful because the Paillier's scheme can only work with finite ring of positive integers. Therefore, the update equation can now be implemented using Paillier's encryption scheme. The correctness of this implementation follows from the results of [6] on fixed-point rational numbers.

First, the public and private keys must be generated such that $\kappa_p \geq 2^{\tilde{n}+1}$ to ensure that no unintended overflow occurs when using the encrypted numbers. The sensors measure, quantize, and encrypt the output to obtain

$$\check{y}_i[k] := \mathfrak{E}(\tilde{y}_i[k], \kappa_p). \tag{30}$$

[4]See [29] for a description of this constant.

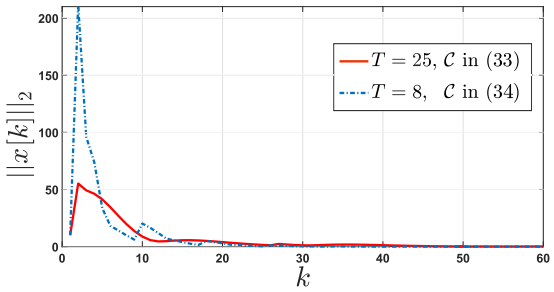

Fig. 1. Norm of the state of the closed-loop system $\|x[k]\|_2$ with quantized controller (25) and quantizer resolution $(n, m) = (24, 14)$.

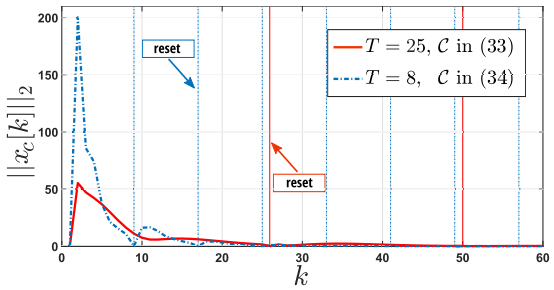

Fig. 2. Norm of the state of the quantized controller $\|x_c[k]\|_2$ in (25) with quantizer resolution $(n, m) = (24, 14)$.

The controller follows the encrypted version of (29) to update its state and compute the actuation signal as

$$(\check{x}_c)_i[k+1] = \begin{cases} \left(\oplus_{j=1}^{n_x} (\check{x}_c)_j[k] \triangle (\tilde{A}_c)_{ij} \right) \\ \quad \oplus \left(\oplus_{j=1}^{n_y} \check{y}_j[k] \triangle (\tilde{B}_c)_{ij} \right), & (k+1) \bmod T \neq 0, \\ \mathfrak{E}(0, \kappa_p), & (k+1) \bmod T = 0, \end{cases} \tag{31}$$

$$\check{u}_i[k] = \left(\oplus_{j=1}^{n_c} (\check{x}_c)_j[k] \triangle (\tilde{C}_c)_{ij} \right) \oplus \left(\oplus_{j=1}^{n_y} (\check{y})_j[k] \triangle (\tilde{D}_c)_{ij} \right). \tag{32}$$

Finally, the actuator extract the control signal by $\tilde{u}_i[k] = \mathfrak{D}(\check{u}_i[k], \kappa_p, \kappa_s) \bmod 2^{\tilde{n}}$, and implements $u_i[k] = 2^{-m(k \bmod T+2)} (\tilde{u}_i[k] - 2^{\tilde{n}} \mathbb{1}_{\tilde{u}_i[k] \geq 2^{\tilde{n}-1}})$.

Remark 4 *National Institute of Standards and Technology (NIST) recommends the use of key length of 2048 bits for factoring-based asymmetric encryption to guarantee that brute-force attacks are not physically possible during the life-time of the services based on projections of computing technologies. This high standard might not be necessary for some applications, such as remote control of autonomous vehicles. To demonstrate this, consider RSA, which is a similar encryption methodology and also a semi-homomorphic encryption relying on hardness of prime number factorization. RSA encryption has been attacked repeatedly using a brute-force methodology; see RSA Challenge*[5]*. Factorization of 663 bit numbers has been shown to take approximately 55 CPU-Years*[6] *[31]. Using IBM Watson (used recently for natural*

[5]https://en.wikipedia.org/wiki/RSA_Factoring_Challenge

[6]A CPU-Year is the amuont of computing work done by a 1 Giga Floating Point Operations Per Second (FLOP) reference machine in a year of dedicated service (8760 hours).

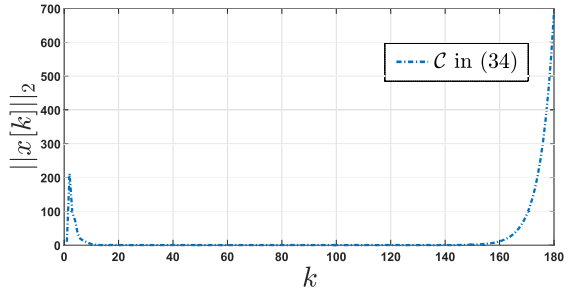

Fig. 3. Norm of the state of the closed-loop system $\|x[k]\|_2$ with quantized controller (36) and quantizer resolution $(n, m) = (24, 14)$.

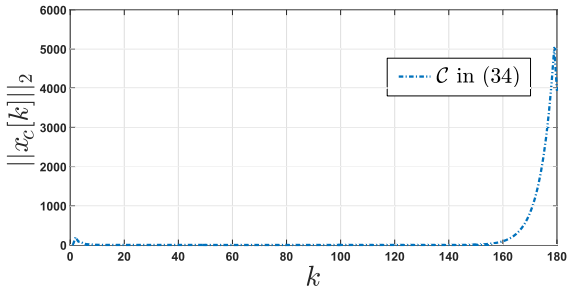

Fig. 4. Norm of the state of the quantized controller $\|x_c[k]\|_2$ in (36) with quantizer resolution $(n, m) = (24, 14)$.

language processing to win quiz show Jeopardy), factorization of 663 bit numbers takes approximately 2.5 years. These numbers are certainly not safe for use in finance or military. However, for remote control of autonomous vehicles, these keys may provide strong-enough guarantees as, by the time that an adversary breaks the code, the autonomous vehicle is in an entirely different location.

IV. Case Study of a Chemical Batch Reactor

We illustrate the performance of our results through a case study of a batch chemical reactor. This case study has been developed over the years as a benchmark example for networked control systems, see, e.g., [32]–[34]. The reactor considered here is open-loop unstable, has one input, and two outputs (please refer to [32]–[34] for details about the system dynamics). We exactly discretize the reactor dynamics introduced in [33] with sampling period $h = 0.1$. The resulting discrete-time linear system is of the form (1) with matrices A, B, C as follows:

$$\left[\begin{array}{c|c|c} A & B & C^\top \end{array}\right] = \left[\begin{array}{cccc|c|cc} 1.18 & 0 & 0.51 & -0.40 & 0 & 1 & 0 \\ -0.05 & 0.66 & -0.01 & 0.06 & 0.47 & 0 & 1 \\ 0.08 & 0.34 & 0.56 & 0.38 & 0.21 & 1 & 0 \\ 0 & 0.34 & 0.09 & 0.85 & 0.21 & -1 & 0 \end{array}\right]. \quad (33)$$

Note that $\text{eig}[A] = \{1.22, 1.01, .60, .42\}$; thus the system is open-loop unstable. Moreover, it can be verified (e.g., using the tools in [26, Theorem 3.3]) that there does not exist a static output feedback controllers of the form $u[k] = Ly[k]$, $L \in \mathbb{R}^{1\times 2}$, stabilizing system (1) with matrices (A, B, C) as in (33). First, using the synthesis results in Section III, we design switching dynamic output feedback controllers of the form (2). Using Theorem 2, and conducting a bisection in $\delta \in [1, \infty)$, and a line search in $\mu \in [-1, 0)$, we look for the smallest δ for which there exist $\mu \in [-1, 0)$ and ν satisfying the synthesis LMIs in Theorem 2. The obtained δ is given by $\delta = \delta^* = 55.0$, the corresponding μ is $\mu = \mu^* = -0.15$, the resetting horizon is $T^* = \text{argmin}_{T\in\mathbb{N}}\ \delta^*(1+\mu^*)^T = 25$, and the reconstructed A_c, B_c, C_c, D_c (see Section III) are given in

$$\left[\begin{array}{c|c} A_c & B_c \\ \hline C_c & D_c \end{array}\right] = \left[\begin{array}{cccc|cc} 0.26 & -0.03 & -0.29 & 0.31 & -0.52 & -0.03 \\ -0.32 & 1.24 & 1.40 & -3.05 & 5.46 & 1.25 \\ -0.45 & 0.02 & 0.87 & -0.75 & 2.32 & -0.01 \\ -0.05 & -0.04 & 0.72 & -0.51 & 2.28 & -0.08 \\ \hline 1.02 & -2.65 & -2.65 & 6.28 & -11.3 & -4.09 \end{array}\right]. \quad (34)$$

This controller satisfies the original inequalities in (5) with $\epsilon = 0.0026$ and any $\varepsilon \in (0.9459, 1)$. For comparison, let $\mu = \mu^* = -0.65$ and search for the smallest δ for which there exists ν satisfying the synthesis LMIs in Theorem 2. In this case, $\delta^* = 3000$, the smallest resetting horizon is $T^* = \text{argmin}_{T\in\mathbb{N}}\ \delta^*(1+\mu^*)^T = 8$, the reconstructed A_c, B_c, C_c, D_c are in

$$\left[\begin{array}{c|c} A_c & B_c \\ \hline C_c & D_c \end{array}\right] = \left[\begin{array}{cccc|cc} -0.18 & -0.01 & -0.77 & 0.84 & -1.11 & -0.01 \\ 9.17 & 0.43 & 13.4 & -16.2 & 22.8 & 0.42 \\ 1.24 & 0.10 & 3.82 & -4.22 & 7.81 & 0.06 \\ 1.32 & 0.10 & 3.47 & -3.87 & 7.89 & 0.06 \\ \hline -19.6 & -0.93 & -28.8 & 34.9 & -49.0 & -2.33 \end{array}\right]. \quad (35)$$

This controller satisfies the original inequalities in (5) with $\epsilon = 2.8 \times 10^{-5}$, and $\varepsilon \in (0.6756, 1)$.

Next, we quantize the controller matrices according to (24) to obtain $(\bar{A}_c, \bar{B}_c, \bar{C}_c, \bar{D}_c)$ with quantizer resolution $(n, m) = (24, 14)$. It can be verified that for A_c, B_c, C_c, D_c in (34) and (35), the corresponding $\bar{A}_c, \bar{B}_c, \bar{C}_c, \bar{D}_c$ satisfy the conditions of Theorem 4 with $(n, m) = (24, 14)$. We quantize sensor measurements $y[k]$ according to (26) with the same resolution $(n, m) = (24, 14)$, and close the system dynamics with the quantized controller in (25). By Theorem 4, the quantizer resolution must satisfy inequality (27) to ensure practical stability of (1) in feedback with (25) in the sense of Theorem 4. Inequality (27), with initial condition $[x(0)^T, x_c(0)^T] = [-6.83, -5.18, -4.05, -3.12, 0, 0, 0, 0]$, amounts to $n > 17$ for the controller in (34) and to $n > 23$ for the controller in (35). Therefore, $(n, m) = (24, 14)$ is enough for practical stabilization using the controllers in (34) and (35). Figures 1 and 2 show $\|x(k)\|_2$ and $\|x_c(k)\|_2$ of the closed-loop dynamics for quantized controllers corresponding to the controllers in (34) and (35) with $(n, m) = (24, 14)$.

To illustrate the need for the proposed resetting controller, we naively implement a standard quantized dynamic controller of the form:

$$\left\{\begin{array}{l} \bar{x}_c[k+1] = \bar{A}_c\bar{x}_c[k] + \bar{B}_c\bar{y}[k], \\ \bar{u}[k] = \bar{C}_c\bar{x}_c[k] + \bar{D}_c\bar{y}[k]. \end{array}\right. \quad (36)$$

We use the same quantizer resolution $(n, m) = (24, 14)$, and compute the matrices $(\bar{A}_c, \bar{B}_c, \bar{C}_c, \bar{D}_c)$ using (24) with (A_c, B_c, C_c, D_c) from (35). This controller is stabilizing even without resets. Note that the Paillier's encryption only

works over the ring of positive integers $\mathbb{Z}_{\kappa_p}$, and thus the controller needs to be transformed so that its states and parameters always belong to this ring. Therefore, as also required for the resetting controller, we must transform $\bar{y}[k]$ and $(\bar{A}_c, \bar{B}_c, \bar{C}_c, \bar{D}_c)$ into positive integers, which can be done using the change of variables in (28) replacing $k \bmod T$ with k (as there is no resetting after T steps in this case). Let the integer representations of $\bar{y}[k]$ and $(\bar{A}_c, \bar{B}_c, \bar{C}_c, \bar{D}_c)$ be similarly denoted by $\tilde{y}[k]$ and $(\tilde{A}_c, \tilde{B}_c, \tilde{C}_c, \tilde{D}_c)$. The equivalent controller in the integer domain is then given by

$$\begin{cases} \tilde{x}_c[k+1] = \big(\tilde{A}_c\tilde{x}_c[k] + \tilde{B}_c\tilde{y}[k]\big) \bmod 2^{\tilde{n}}, \\ \tilde{u}[k] = \big(\tilde{C}_c\tilde{x}_c[k] + \tilde{D}_c\tilde{y}[k]\big) \bmod 2^{\tilde{n}}. \end{cases} \tag{37}$$

Finally, given $\tilde{u}[k]$, the actuators implement the control action:

$$u_i[k] = 2^{-m(k+2)}(\tilde{u}_i[k] - 2^{\tilde{n}}\mathbb{1}_{\tilde{u}_i[k]\geq 2^{\tilde{n}-1}}). \tag{38}$$

Because we must ensure that $2^{\tilde{n}} \leq \kappa_p$, we need to select a large, yet finite $\tilde{n}$. Here, for illustration purposes, we selected a key length of 2048 bits and $\tilde{n} = 2014$. Figure 3 and 4 illustrate the norm of the state of the closed-loop system, $\|x[k]\|_2$, with controller (35)-(38), and the state of the controller in the quantized domain, $\|x_c[k]\|$, respectively. Note that, even though (35)-(36) is a stabilizing controller (if no under/over flow occur), when implementing (35)-(38), the closed-loop system is unstable due to under/over flows.

V. Conclusions and Future Work

A secure and private implementation of linear time-invariant dynamic controllers using the Paillier's encryption was presented. The state is reset to zero periodically to avoid data overflow or underflow within the encryption space. A control design approach was presented to ensure the stability and performance of the closed-loop system with encrypted controller. Future work can focus on nonlinear systems and controllers.

References

[1] S. C. Patel, G. D. Bhatt, and J. H. Graham, "Improving the cyber security of SCADA communication networks," *Communications of the ACM*, vol. 52, no. 7, pp. 139–142, 2009.

[2] C. Gentry, "Fully homomorphic encryption using ideal lattices," in *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, STOC '09, pp. 169–178, 2009.

[3] 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 Prague, Czech Republic, May 2–6, 1999 Proceedings* (J. Stern, ed.), pp. 223–238, Berlin, Heidelberg: Springer Berlin Heidelberg, 1999.

[4] T. ElGamal, "A public key cryptosystem and a signature scheme based on discrete logarithms," *IEEE Transactions on Information Theory*, vol. 31, no. 4, pp. 469–472, 1985.

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

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

[7] F. Farokhi, I. Shames, and N. Batterham, "Secure and private cloud-based control using semi-homomorphic encryption," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 163–168, 2016.

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

[9] J. Kim, C. Lee, H. Shim, J. H. Cheon, A. Kim, M. Kim, and Y. Song, "Encrypting controller using fully homomorphic encryption for security of cyber-physical systems," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 175–180, 2016.

[10] M. S. Darup, A. Redder, and D. E. Quevedo, "Encrypted cloud-based MPC for linear systems with input constraints," *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 535–542, 2018.

[11] Y. Lin, F. Farokhi, I. Shames, and D. Nešić, "Secure control of nonlinear systems using semi-homomorphic encryption," in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 5002–5007, IEEE, 2018.

[12] A. B. Alexandru, M. Morari, and G. J. Pappas, "Cloud-based MPC with encrypted data," *arXiv preprint arXiv:1803.09891*, 2018.

[13] J. H. Cheon, K. Han, H. Kim, J. Kim, and H. Shim, "Need for controllers having integer coefficients in homomorphically encrypted dynamic system," in *2018 IEEE Conference on Decision and Control (CDC)*, pp. 5020–5025, 2018.

[14] H. S. Junsoo Kim and K. Han, "Dynamic controller that operates over homomorphically encrypted data for infinite time horizon," technical report, arXiv:1912.07362 [eess.SY]. https://arxiv.org/abs/1912.07362.

[15] P. Tam and J. Moore, "Stable realization of fixed-lag smoothing equations for continuous-time signals," *IEEE Transactions on Automatic Control*, vol. 19, pp. 84–87, February 1974.

[16] J. Bakkeheim, T. A. Johansen, Ø. N. Smogeli, and A. J. Sorensen, "Lyapunov-based integrator resetting with application to marine thruster control," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 5, pp. 908–917, 2008.

[17] J. Clegg, "A nonlinear integrator for servomechanisms," *Transactions of the American Institute of Electrical Engineers, Part II: Applications and Industry*, vol. 77, no. 1, pp. 41–42, 1958.

[18] K. Krishnan and I. Horowitz, "Synthesis of a non-linear feedback system with significant plant-ignorance for prescribed system tolerances," *International Journal of Control*, vol. 19, no. 4, pp. 689–706, 1974.

[19] O. Beker, C. Hollot, Y. Chait, and H. Han, "Fundamental properties of reset control systems," *Automatica*, vol. 40, no. 6, pp. 905–915, 2004.

[20] C. Prieur, I. Queinnec, S. Tarbouriech, and L. Zaccarian, "Analysis and synthesis of reset control systems," *Foundations and Trends in Systems and Control*, vol. 6, no. 2-3, pp. 117–338, 2018.

[21] Y. Guo, W. Gui, C. Yang, and L. Xie, "Stability analysis and design of reset control systems with discrete-time triggering conditions," *Automatica*, vol. 48, no. 3, pp. 528–535, 2012.

[22] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*. Chapman & Hall/CRC Cryptography and Network Security Series, Taylor & Francis, 2 ed., 2014.

[23] X. Yi, R. Paulet, and E. Bertino, *Homomorphic encryption and applications*, vol. 3. Springer, 2014.

[24] V. Syrmos, C. Abdallah, P. Dorato, and K. Grigoriadis, "Static output feedback—a survey," *Automatica*, vol. 33, pp. 125 – 137, 1997.

[25] Y.-Y. Cao, J. Lam, and Y.-X. Sun, "Static output feedback stabilization: An ILMI approach," *Automatica*, vol. 34, pp. 1641 – 1645, 1998.

[26] G. I. Bara and M. Boutayeb, "Static output feedback stabilization with $\mathcal{H}_\infty$ performance for linear discrete-time systems," *IEEE Transactions on Automatic Control*, vol. 50, pp. 250–254, 2005.

[27] K. J. Aström and B. Wittenmark, *Computer-controlled Systems (3rd Ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997.

[28] C. Scherer, P. Gahinet, and M. Chilali, "Multiobjective output-feedback control via LMI optimization," *IEEE Transactions on Automatic Control*, vol. 42, pp. 896–911, 1997.

[29] C. Murguia, F. Farokhi, and I. Shames, "Secure and private implementation of dynamic controllers using semi-homomorphic encryption," technical report, arXiv:1812.04168 [math.OC]. https://arxiv.org/abs/1812.04168.

[30] H. K. Khalil, *Nonlinear Systems*. Englewood Cliffs, NJ: Prentice-Hall, 3nd ed., 2002.

[31] A. J. Elbirt, *Understanding and Applying Cryptography and Data Security*. CRC Press, 2009.

[32] D. Carnevale, A. R. Teel, and D. Nesic, "A Lyapunov proof of an improved maximum allowable transfer interval for networked control systems," *IEEE Transactions on Automatic Control*, vol. 52, pp. 892–897, 2007.

[33] G. C. Walsh, Hong Ye, and L. G. Bushnell, "Stability analysis of networked control systems," *IEEE Transactions on Control Systems Technology*, vol. 10, pp. 438–446, 2002.

[34] D. Nesic and A. R. Teel, "Input-output stability properties of networked control systems," *IEEE Transactions on Automatic Control*, vol. 49, pp. 1650–1667, 2004.