Encrypted Control System with Quantizer

Masako Kishida

***Abstract*— This paper considers the design of encrypted control systems to secure data privacy when the control systems operate over a network. In particular, we propose to combine Paillier cryptosystem with a quantizer whose sensitivity changes with the evolution of the system. This allows the encrypted control system to balance between the cipher strength and processing time. Such an ability is essential for control systems that are expected to run real-time. It also allows the closed-loop system to achieve the asymptotic stability for linear systems. Extensions to event-triggered control and nonlinear control systems are also discussed.**

I. Introduction

Networked control systems are ubiquitous [1]–[4]. The use of networks has not only reduced the deployment cost and increased the flexibility of control systems, but also allowed the systems to outsource computations of control inputs to a cloud controller when a plant does not have sufficient computational resources [5], [6]. However, this raises new privacy and security concerns; plants may want to protect the privacy of the sampled data because the cloud controller is not trustworthy, and communication networks may be vulnerable to cyber-attacks [1].

One approach to protect privacy is to use "differential privacy" [7]. Differential privacy adds noise to the data so that the contribution of a specific agent is hidden without changing the solution to the problem significantly. Although differential privacy is a relatively new notion, it has found applications in a variety of networked systems including systems and controls [8]–[10].

Another approach is to use "encryptions". "Encrypted control system" is a control architecture in which the controller computes the control input using encrypted sampled data without decrypting them. As the controller does not require a private key for decryption, encrypted control systems can not only protect the privacy of the plant data from the controller, but also enhance the cyber-security. The idea of encrypted control system was first proposed based on public-key RSA [11] and ElGamal [12] cryptosystems in [13]. Subsequently, an encrypted control system with Paillier cryptosystem [14] was considered in [15], and a solution approach to quadratic optimization with Paillier cryptosystem was proposed in [16].

This paper proposes encrypted control architectures using Paillier cryptosystem [14] *combined with quantizers*. As in [13], the proposed architectures do not require the controller to have private keys to compute the control inputs. The main contribution of this paper is to propose the augmentation of quantizers whose sensitivity changes while the system evolves. The quantizers are applied to real-valued sampled data and map to integers in $[-q_{\text{sat}}, q_{\text{sat}}]$ for a fixed saturation value q_{sat}. Thus, the plaintext space (key length) may be kept small by choosing a small q_{sat}, which *allows us to balance between cipher strength and control performance (sampling time)*. This is essential for control systems that require real-time computation of control inputs. Moreover, the use of quantizers eliminates the analysis for the fixed-arithmetic [15] to guarantee stability, and allows the linear system to achieve asymptotic stability. Other contributions include extensions to event-triggered control and nonlinear control systems. In particular, this is the first study to consider the construction of an encrypted nonlinear control system.

The rest of the paper is organized as follows. Section II provides the mathematical preliminaries and operation rules of encrypted data (ciphertext). An encrypted linear state-feedback control is presented in Section III, which is extended to event-triggered control in Section IV and nonlinear control in Section V. Finally, the paper is concluded in Section VI.

M. Kishida is with National Institute of Informatics / address: 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, Japan / phone: +81-3-4212-2231 / email: kishida@nii.ac.jp

II. Preparations

A. Notation

The sets of real numbers and integers are denoted by $\mathbb{R}$ and $\mathbb{Z}$, respectively. The set of vectors of length n is denoted by $\mathbb{R}^n$ and the set of matrices of size n by m is denoted by $\mathbb{R}^{n\times m}$. The greatest common divisor and the least common multiple of $a, b \in \mathbb{Z} \setminus \{0\}$ are denoted by $\gcd(a,b)$ and $\operatorname{lcm}(a,b)$, respectively. We define the sets of integers $\mathbb{Z}_n := \{z \in \mathbb{Z} : 0 \le z < n\}$ and $\mathbb{Z}_n^* := \{z \in \mathbb{Z}_n : \gcd(z,n) = 1\}$. For a vector $v \in \mathbb{R}^n$, the ith element of v is denoted by v_i, and the Euclidean norm is denoted by $\|v\|$. For a matrix $M \in \mathbb{R}^{n\times n}$, the i,jth element of M is denoted by m_{ij}, and the induced 2-norm and the Frobenius norm are denoted by $\|M\|$ and $\|M\|_F$, respectively. The maximum and minimum eigenvalues of a symmetric matrix $M = M^T$ are denoted by $\lambda_{\max}(M)$ and $\lambda_{\min}(M)$, respectively. The floor function is denoted by $\lfloor x \rfloor := \max\{k \in \mathbb{Z} : k < x\}$.

B. Quantizer

Paillier cryptosystem operates over the message of non-negative integers (plaintext). However, the control theory usually deals with the data of real numbers. In order to map the data to nonnegative integers, we use quantizers.

For a positive integer q_{sat} and a positive real number Δ, a

quantizer $q : \mathbb{R} \to \mathbb{Z}$ is given by [17]

$$q_\Delta(x) := \begin{cases} q_{\text{sat}} & \text{if } x > (q_{\text{sat}} + 1/2)\Delta, \\ -q_{\text{sat}} & \text{if } x \le -(q_{\text{sat}} + 1/2)\Delta, \\ \left\lfloor \dfrac{x}{\Delta} + \dfrac{1}{2} \right\rfloor & \text{if } -(q_{\text{sat}}+1/2)\Delta < x \le (q_{\text{sat}}+1/2)\Delta, \end{cases} \tag{1}$$

where Δ is the sensitivity of the quantizer and q_{sat} is the saturation value of the quantizer.

For the shorthand notation, we define

$$x_q := q_\Delta(x),\ \bar{x} := x_q \Delta,\ \tilde{x} := x - \bar{x}, \tag{2}$$

then

$$\bar{x} - \Delta/2 \le x < \bar{x} + \Delta/2,\ |\tilde{x}| \le \Delta/2. \tag{3}$$

With an abuse of notation, we write $v_q := q_\Delta(v) \in \mathbb{R}^n$ and $M_q := q_\Delta(M) \in \mathbb{R}^{n \times m}$ to denote element-wise quantization with Δ for a vector v and a matrix M, and define $\bar{v}$, $\tilde{v}$, $\bar{M}$ and $\tilde{M}$ similarly to (2). Then,

$$\|\tilde{v}\| \le \Delta\sqrt{n}/2,\ \|\tilde{M}\| \le \|\tilde{M}\|_{\text{F}} \le \Delta\sqrt{nm}/2. \tag{4}$$

C. Paillier cryptosystem

An overview of Paillier cryptosystem [14] is given below.

1) Encryption scheme:

- Key generation:
  - Choose two large prime numbers p and q randomly and independently of each other such that $\gcd(pq, (p-1)(q-1)) = 1$
  - Generate public key: $N = pq$, and $g \in \mathbb{Z}^*_{N^2}$ such that the order of g is a multiple of N
  - Generate private key: $\lambda = \operatorname{lcm}(p-1, q-1)$
- Encryption (P-encryptor): Given a message $m \in \mathbb{Z}_N$,
  - Compute ciphertext: $c = g^m \cdot r^N \bmod N^2$ with a random integer $r \in \mathbb{Z}^*_N$
- Decryption (P-decryptor): Given a ciphertext $c < N^2$,
  - Compute the message: $m = L(c^\lambda \bmod N^2)/L(g^\lambda \bmod N^2) \bmod N$, where $L(x) = (x-1)/N$

We denote the Paillier encryption of the message m by $\mathcal{E}_P(m)$, and the decryption of ciphertext c by $\mathcal{D}_P(c)$.

2) Encryption properties: Paillier cryptosystem allows us to add two encrypted values with the addition operator $\oplus$ and to multiply by a plaintext with the multiplication operator $\otimes$. Namely, for $m, m_i \in \mathbb{Z}_N$, it holds that

$$\begin{aligned} &\mathcal{D}_P\big(\mathcal{E}_P(m_1) \oplus \mathcal{E}_P(m_2) \bmod N^2\big) = m_1 + m_2 \bmod N, \\ &\mathcal{D}_P\left(a \otimes \mathcal{E}_P(m) \bmod N^2\right) = am \bmod N, \end{aligned} \tag{5}$$

therefore,

$$\mathcal{D}_P\left(\bigoplus_i a_i \otimes \mathcal{E}_P(m_i) \bmod N^2\right) = \sum_i a_i m_i \bmod N. \tag{6}$$

D. Multiplicative blinding using a random number (r-encryptor/r-decryptor)

To perform some computations that can not be performed on Paillier encrypted values $\mathcal{E}_P(m)$, we must offload such computations. To avoid direct access, as is often done, we "obfuscate" the message m by multiplying a random number $r \in \mathbb{Z}_{r_{\max}}$ for some $r_{\max} \in \mathbb{Z}/\{0\}$ to obtain $\mathcal{E}_r(m) := rm$ (multiplicative blinding [18]). The inverse operation is denoted by $\mathcal{D}_r(c) := c/r$.

E. Matrix-vector multiplication

The element-wise encryptions for a vector v and a matrix M are denoted by $\mathcal{E}_P(v)$ and $\mathcal{E}_P(M)$, respectively, and the corresponding element-wise decryptions are denoted by $\mathcal{D}_P(v)$ and $\mathcal{D}_P(M)$, respectively. The element-wise multiplicative blinding for a vector v and a matrix M are denoted by $\mathcal{E}_r(v) := rv$ and $\mathcal{E}_r(M) := rM$, respectively, and the corresponding element-wise decryptions are denoted by $\mathcal{D}_r(v) := (1/r)v$ and $\mathcal{D}_r(M) := (1/r)M$, respectively.

With these notation, for $b_i \in \mathbb{Z}_N$, observe that

$$\begin{aligned} &\mathcal{D}_P\left(\mathcal{E}_r(a) \otimes \mathcal{E}_P(b) \bmod N^2\right) = \mathcal{E}_r(a)b \bmod N, \\ &\mathcal{D}_P\left(\bigoplus_i \mathcal{E}_r(a_i) \otimes \mathcal{E}_P(b_i) \bmod N^2\right) = \sum_i \mathcal{E}_r(a_i) \otimes b_i \bmod N. \end{aligned} \tag{7}$$

Therefore, for $\mathcal{E}_r(A)$ with a matrix $A \in \mathbb{Z}^{n \times n}$ and $\mathcal{E}_P(b)$ with a vector $b \in \mathbb{Z}^n$ such that $b_i \in \mathbb{Z}_N$, it holds that

$$\mathcal{D}_P\big(\big(\mathcal{E}_r(A) \otimes \mathcal{E}_P(b)\big)_i \bmod N^2\big) = \sum_j \mathcal{E}_r(a_{ij}) b_j \bmod N, \tag{8}$$

thus

$$\mathcal{D}_P\left(\mathcal{E}_r(A) \otimes \mathcal{E}_P(b) \bmod N^2\right) = \mathcal{E}_r(Ab) \bmod N. \tag{9}$$

If $r \in \mathbb{Z}_N$ and each element of rAb is in $\mathbb{Z}_N$, we have

$$\mathcal{D}_r\left(\mathcal{D}_P(\mathcal{E}_P\left(\mathcal{E}_r(A)b \mod N\right) \bmod N^2)\right) = Ab. \tag{10}$$

F. Remarks between quantizer and encryptor

After mapping a real number to an integer using the quantizer, we need to convert this integer to an element in $\mathbb{Z}_N$. As Paillier arithmetic uses modulo N, we may take the convention that a number $x < N/3$ is positive, and that a number $x > 2N/3$ is negative. The range $N/3 < x < 2N/3$ allows for overflow detection [19]. We include this mapping from an integer to a nonnegative integer in the steps of encryption, i.e., we use $\mathcal{E}_P(x)$ to denote the Paillier encryption of $x \bmod N$. Similarly, we use $\mathcal{D}_P(x)$ to denote the Paillier decryption of x followed by subtraction of N if the Paillier decryption yields the value greater than $N/2$.

III. Encrypted State-feedback Control

In this section, we present an encrypted linear state-feedback control system that achieves asymptotic stability.

Consider the discrete-time linear system

$$x[t+1] = Ax[t] + Bu[t], \quad t = 0, 1, 2, \cdots, \tag{11}$$

where $x[t] \in \mathbb{R}^{n_x}$ is the system state, and $u[t] \in \mathbb{R}^{n_u}$ is the control input. Suppose that (A, B) is controllable, and $K \in \mathbb{R}^{n_u \times n_x}$ is found such that $A - BK$ is Schur, i.e., all the eigenvalues of $A - BK$ are inside the unit circle in the complex plane. Then, the system (11) is stabilized by the state-feedback control

$$u[t] = -Kx[t]. \tag{12}$$

Problem: Design an encrypted control system for (11)-(12) **that achieves the asymptotic stability while protecting the privacy of the sampled data** $x[t]$ **and the controller gain** K **from the controller.**

In the following subsections, the overall control design is presented, followed by the analysis and design of quantizers.

A. Overview of encryption architecture

The proposed architecture consists of the plant node and controller node between which no information from which the values of $x_q[t]$ and K_q can be identified is exchanged (Figure 1). In order to encrypt the sampled data $x[t]$ and the gain K, there are two quantizers in the system: one for $x[t]$ has the time-varying sensitivity $\Delta[t]$ and one for K has the constant sensitivity Δ_g, and both quantizers are designed not to saturate.

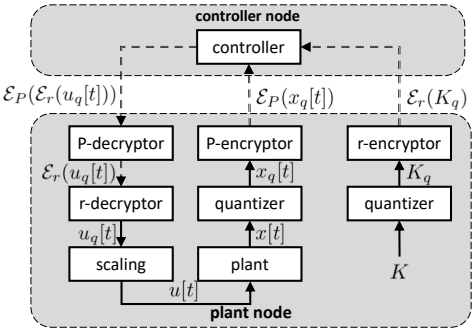

Fig. 1: Proposed encrypted control architecture. Dashed lines indicate the flow of encrypted data and solid lines indicate the flow of plaintext data.

The role of each node is summarized below (**P** and **C** denote the plant and controller nodes, respectively):

- **P:** Quantizes the gain K to K_q, obfuscates K_q to $\mathcal{E}_r(K_q)$, and then sends $\mathcal{E}_r(K_q)$ to the controller node.
- **P:** Quantizes the sampled state $x[t]$ of the plant to $x_q[t]$, encrypts $x_q[t]$ to $\mathcal{E}_P(x_q[t])$, and then sends $\mathcal{E}_P(x_q[t])$ to the controller node at every sampling time after some time t_0. Sends $\mathcal{E}_P(0)$ to the controller node at every sampling time before t_0. (The time instance t_0 is determined in III-B.2.)
- **C:** Upon receiving the obfuscated/encrypted scaled data $\mathcal{E}_r(K_q)$ and $\mathcal{E}_P(x_q[t])$, computes the encrypted scaled control inputs $\mathcal{E}_P(\mathcal{E}_r(u_q[t]))$ for $u_q[t] = K_q x_q[t]$, and sends $\mathcal{E}_P(\mathcal{E}_r(u_q[t]))$ to the plant node.
- **P:** Upon receiving the encrypted scaled control inputs $\mathcal{E}_P(\mathcal{E}_r(u_q[t]))$, decrypts $\mathcal{E}_P(\mathcal{E}_r(u_q[t]))$ to $\mathcal{E}_r(u_q[t])$ then to $u_q[t]$.
- **P:** Scales $u_q[t]$ to obtain $u[t] = u_q[t]\Delta_g\Delta[t]$ using the sensitivities Δ_g and $\Delta[t]$ and applies it to the plant.

Note that the controller uses the encrypted data of the quantizer output (scaled approximations). This means that the data sent and received among the three node is encryptions of integers between $-q_{\text{sat}}$ and q_{sat}. Thus, there are no fractional bits, which renders multiplication easy [15].

This architecture preserves the privacy of the plant state $x[t]$ ($x_q[t]$) and the controller gain K (K_q) from the controller node because the controller does not know the private key and the value of r. Thus, if quantizers are designed such that the closed-loop achieves the asymptotic stability, then the overall encrypted control system achieves asymptotic stability while protecting the privacy of $x[t]$ and K in the sense that the controller node can access only the encrypted data of $x[t]$ and K.

Remark 1:

- When generating the public key, p and q are chosen such that N for the cryptosystem satisfies $N > 3(q_{\text{sat}} + 1/2)(q_{\text{sat,g}} + 1/2)n_x r_{\max}$, where q_{sat} and $q_{\text{sat,g}}$ are the saturation values of the quantizers for the states and the controller gain, respectively, and determined in the next subsection. This guarantees that the elements of $rK_q x_q \bmod N$ are uniquely determined for each $rK_q x_q$ and vice versa. Recalling the notation for $\mathcal{E}_P$ and $\mathcal{D}_P$ in Section II-F and (10), for

$$\mathcal{E}_P\left(\mathcal{E}_r(u_q)\right) = \mathcal{E}_r(K_q) \otimes \mathcal{E}_P\left(x_q\right) = \mathcal{E}_P\left(rK_q x_q\right),$$

  we have

$$\mathcal{D}_r(\mathcal{D}_P\left(\mathcal{E}_P\left(\mathcal{E}_r(u_q)\right)\right)) = \frac{1}{r}(\mathcal{D}_P\left(\mathcal{E}_P\left(rK_q x_q\right)\right)) = K_q x_q.$$

  Similar assumptions on N are posed on later sections.
- The sensitivity of the quantizer for K can be time-varying. For example, it can be synchronized with the sensitivity of the quantizer in the plant node as long as (16) is satisfied.
- The quantization and encryption of K are required only once, but can be repeated using different random number at every sampling time.

Remark 2: Strictly speaking, the quantization and encryption of the state $x[t]$ is performed at sensor, which differs from where the quantization and encryption of the gain K is performed. Similarly, decryptions and scaling to obtain the control input $u[t]$ is performed at actuator. Those are combined in the plant node to indicate that they have common keys and quantizers.

B. Quantizer design

To analyze the effect of quantization in the design of quantizers, consider the system in Fig. 1 without encryptors and decryptors. The quantized closed-loop system for (11)-(12) is

$$x[t+1] = Ax[t] + Bu[t] = Ax[t] - B\bar{K}\bar{x}[t]. \tag{13}$$

1) Quantizer for the gain: Let us first determine the sensitivity Δ_g of the quantizer for the gain K.

As $A-BK$ is Schur, for any given $Q=Q^T>0$, there exists $P=P^T>0$ such that

$$(A-BK)^TP(A-BK)-P+Q=0. \tag{14}$$

It is guaranteed that $A-B\bar{K}$ is Schur if

$$\begin{aligned}&(A-B\bar{K})^TP(A-B\bar{K})-P\\ =&(A-BK+B\tilde{K})^TP(A-BK+B\tilde{K})-P\\ \le&-Q+(A-BK)^TPB\tilde{K}+\tilde{K}^TB^TP(A-BK)\\ &+\tilde{K}^TB^TPB\tilde{K}\le-\varepsilon I,\ \varepsilon>0.\end{aligned} \tag{15}$$

Therefore, after some computations using (4), the sensitivity Δ_g is chosen to satisfy

$$\begin{aligned}\Delta_g\le\varepsilon'+\frac{2}{\sqrt{n_xn_u}\|B^TPB\|}\big(-\|(A-BK)^TPB\|\\ +\sqrt{\|(A-BK)^TPB\|^2+\lambda_{\min}(Q)\|B^TPB\|}\Big),\end{aligned} \tag{16}$$

for any $Q=Q^T>0$ of the designer's choice and the corresponding P satisfying (14) with $\varepsilon'>0$.

Once the sensitivity is determined, the saturation value $q_{\text{sat},g}$ is selected such that the elements of K are not truncated, i.e.,

$$\max_i\max_j|K_{ij}|\le(q_{\text{sat},g}-1/2)\Delta_g. \tag{17}$$

To simplify the notation, once the quantizer is determined and the gain is quantized to $\bar{K}=K_q\Delta_g$, we choose $\bar{Q}=\bar{Q}^T>0$ and find $\bar{P}=\bar{P}^T>0$ that solves

$$(A-B\bar{K})^T\bar{P}(A-B\bar{K})-\bar{P}+\bar{Q}=0. \tag{18}$$

We use this $\bar{P}$ and $\bar{Q}$ to design the quantizer for the states.

2) Quantizer for the states: The design of quantizer for the states follows the approach proposed in [17]. Due to space limitation, we only present the summary of the quantizer and interested readers are referred to the cited reference.

The employed quantizer uses a constant saturation level q_{sat} and a time-varying sensitivity given by

$$\Delta[t]=\begin{cases}\|A\|^{2t}, & 0\le t<t_0\\ \Delta_i:=\Omega^i\Delta_0, & t_0\le t_i\le t<t_{i+1},\end{cases} \tag{19}$$

where $\Delta_0=\|A\|^{2t_0}$, Ω is a scaling factor and t_i are the time instances of sensitivity updates. Thus, the sensitivity decreases by the factor of Ω at every time of updates.

The scaling factor is given by

$$\Omega:=\Omega'\sqrt{\frac{\lambda_{\max}(\bar{P})}{\lambda_{\min}(\bar{P})}}\left(q_{\text{sat}}-\frac{1}{2}\right)^{-1}, \tag{20}$$

where

$$\Omega':=(\Theta\sqrt{n_x}+\varepsilon)\sqrt{\frac{\lambda_{\max}(\bar{P})}{\lambda_{\min}(\bar{P})}}+\sqrt{n_x}, \tag{21}$$

$$\begin{aligned}\Theta:=\frac{1}{2\lambda_{\min}(\bar{Q})}\big(\|(A-B\bar{K})^T\bar{P}B\bar{K}\|\\ +\sqrt{\|(A-B\bar{K})^T\bar{P}B\bar{K}\|^2+\lambda_{\min}(\bar{Q})\|\bar{K}^TB^T\bar{P}B\bar{K}\|}\Big),\end{aligned} \tag{22}$$

and parameters $\varepsilon>0$ and $q_{\text{sat}}\ge1$ are chosen such that $\Omega\in(0,1)$.

The time instances t_i are given by

$$\begin{aligned}t_0:=&\min\big\{t\ge1:\big\|q_{\Delta[t]}(x[t])\big\|\\ &\le\left(q_{\text{sat}}-\frac{1}{2}\right)\sqrt{\frac{\lambda_{\max}(\bar{P})}{\lambda_{\min}(\bar{P})}}-\frac{\sqrt{n_x}}{2}\Big\},\\ t_{i+1}:=&\min\left\{t\ge t_i+1:\|q_{\Delta_i}(x[t])\|\le\Omega'-\frac{\sqrt{n_x}}{2}\right\}.\end{aligned} \tag{23}$$

By construction, it holds that

$$\begin{aligned}&\|x[t_0]\|\le\Delta_0\left(q_{\text{sat}}-\frac{1}{2}\right)\sqrt{\frac{\lambda_{\min}(\bar{P})}{\lambda_{\max}(\bar{P})}},\\ &\|x[t_i]\|\le\Delta_i\Omega',\ i=1,2,\cdots.\end{aligned} \tag{24}$$

and this quantizer guarantees that

$$x[t]\in R_{i+1}:=\left\{x:x^T\bar{P}x\le\lambda_{\min}(\bar{P})\Delta_i^2\left(q_{\text{sat}}-1/2\right)^2\right\} \tag{25}$$

for $t\in[t_i,t_{i+1})$.

These quantizers lead to asymptotic stability of the closed-loop system because the rule for sensitivity updates (19) implies $\Delta[t]\to0$ as $t\to\infty$, and (25) implies that $x[t]$ approaches to 0 as $\Delta[t]\to0$.

IV. Extension to Event-triggered Control

This section presents how to augment an event-triggered control scheme to the encrypted control law developed in Section III to save communications and actuator updates. Event-triggered control takes samples of the plant state at every time instance and updates the control input only when specified conditions are satisfied [20].

Problem: Design an event-triggered encrypted control system for (11)-(12) **that achieves the asymptotic stability while protecting the privacy of the sampled data** $x[t]$ **and the controller gain** K **from the controller.**

We propose to augment an event-trigger architecture to the plant node. More specifically, we implement the event-trigger mechanism between the plant and the quantizer in the plant node. This way, the sampled data is quantized, encrypted and sent to the controller only when an event-trigger condition is met. In the following, the event-trigger condition is designed.

The event-triggered control system is given by

$$\begin{aligned}x[t+1]&=Ax[t]+Bu[t],\\ u[t]&=-\bar{K}\bar{x}[t^{(i)}],\ t^{(i)}\le t<t^{(i+1)}.\end{aligned} \tag{26}$$

where $t^{(i)}$ for $i = 1, 2, \cdots$, are the time instances of the control updates, and $\bar{x}[t^{(i)}] := x_q[t^{(i)}]\Delta[t^{(i)}]$.

Using a Lyapunov function $V[t] = x^T[t]\bar{P}x[t]$,

$$V[t+1] - V[t] \le -\lambda_{\min}(\bar{Q})\|x\|^2 + 2\|(A - B\bar{K})^T\bar{P}B\bar{K}\|\|x\|\|e\| + \|\bar{K}^T B^T\bar{P}B\bar{K}\|\|e\|^2,$$

where $e[t] = x[t] - \bar{x}[t^{(i)}]$. This Lyapunov function is negative outside the ball $\{x : \|x\| \le 2\Theta\|e\|\}$, where Θ is in (22).

Thus, an event-trigger condition can be set as

$$t^{(i+1)} = \min\left\{t \ge t^{(i)} : \|x[t]\| \le 2\Theta\|e[t]\|\right\}. \quad (27)$$

When the event-trigger condition is satisfied, check if the quantizer needs to be updated or not, and update if necessary using (23). The rest of the analysis is the same as in Section III and [17]. This is because the aforementioned event-trigger condition (27) guarantees the decrease of the Lyapunov function, based on which the analysis is developed.

This is a straightforward extension of well-known results, because the plant node knows both $x[t^{(i)}]$ and $x[t]$.

Remark 3: It is also possible to augment an event-trigger architecture to the controller node rather than the plant node. However, in order to do this, it is needed to add another node that communicates with the controller and checks the satisfaction of the event-triggered condition.

V. Extension to Nonlinear Systems

This section extends the approach in Section III to a simple nonlinear system using feedback linearization [21].

Consider the scalar nonlinear system

$$x[t+1] = ax[t] + b(u[t] - \alpha(x[t])),\ t = 0, 1, 2, \cdots, \quad (28)$$

where $x[t] \in \mathbb{R}$ is the system state, and $u[t] \in \mathbb{R}$ is the control input. Assume that $ab \neq 0$.

The feedback linearization uses the control input $u[t] = \alpha(x[t]) - v[t]$, yielding

$$x[t+1] = ax[t] - bv[t]. \quad (29)$$

If $k \in \mathbb{R}$ such that $|a - bk| < 1$ is selected, then, the system (29) is stabilized by $v[t] = kx[t]$, and (28) is stabilized by

$$u[t] = \alpha(x[t]) - kx[t]. \quad (30)$$

Problem: Design an encrypted control system for (28) **using** (30) **that achieves the practical stability while protecting the privacy of the sampled data** $x[t]$ **on a bounded set** $\mathcal{X} := [x_{\min}, x_{\max}]$ **from the controller.**

The system is said to be practically stable if $|x[0]| < c_1$, then $|x[t]| < c_2$ for $t \ge \bar{t}$ for some $\bar{t} > 0$ for given c_1 and c_2 such that $0 < c_1 < c_2$ [22]. The reason for requiring practical stability instead of asymptotic stability will become clear in the rest of this section.

A. Function approximation

In order to compute the control input using encrypted data for $x[t]$, we first approximate the nonlinear function $\alpha(x[t])$ using the quantized values. From Weierstrass approximation theorem [23], for any $\varepsilon_1' > 0$ there exist p and c_j such that $\alpha_p(x) := \sum_{j=0}^{p} c_j x^j$ satisfies

$$|\alpha_p(x) - \alpha(x)| \le \varepsilon_1',\ \forall x \in \mathcal{X}. \quad (31)$$

With a quantizer of sensitivity Δ, define

$$\bar{\alpha}_p(\bar{\mathbf{x}}) := \sum_{j=0}^{p} \bar{c}_j \bar{x}^{(j)} = \mathbf{c}_q^T \mathbf{x}_q \Delta^2, \quad (32)$$

where $c_{j,q} = q_\Delta(c_j)$, $x_q^j = q_\Delta(x^j)$ as usual, and

$$\begin{aligned} &\bar{c}_j := c_{j,q}\Delta,\ \bar{x}^{(j)} := x_q^j\Delta, \\ &\mathbf{c} := \begin{bmatrix} c_0 & c_1 & \cdots & c_p \end{bmatrix}^T,\ \mathbf{x} := \begin{bmatrix} 1 & x & \cdots & x^p \end{bmatrix}^T, \\ &\mathbf{c}_q := q_\Delta(\mathbf{c}),\ \mathbf{x}_q := q_\Delta(\mathbf{x}),\ \bar{\mathbf{x}} = \mathbf{x}_q\Delta. \end{aligned} \quad (33)$$

Then with some constant ε_2, it holds that

$$\begin{aligned} |\bar{\alpha}_p(\bar{\mathbf{x}}) - \alpha_p(x)| &\le \sum_{j=0}^{p} |\bar{c}_j - c_j||\bar{x}^{(j)}| \\ &+ |c_j - \bar{c}_j||\bar{x}^{(j)} - x^j| + |\bar{c}_j||\bar{x}^{(j)} - x^j| \le \varepsilon_2\Delta/2. \end{aligned} \quad (34)$$

With $\varepsilon_1 = 2\varepsilon_1'/\Delta$, (31) and (34) imply that

$$|\bar{\alpha}_p(\bar{\mathbf{x}}) - \alpha(x)| \le M\Delta/2,\ M := \varepsilon_1 + \varepsilon_2. \quad (35)$$

B. Overview of encryption architecture

As before, the proposed architecture consists of two nodes between which only encrypted data is exchanged (Figure 2). However, two quantizers maintain the same sensitivity $\Delta[t]$.

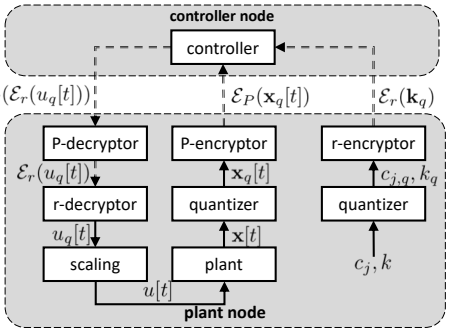

Fig. 2: Proposed encrypted control architecture. Dashed lines indicate the flow of encrypted data and solid lines indicate the flow of plaintext data.

The role of each node is summarized below (**P** and **C** denote the plant and controller nodes, respectively):

- **P:** Quantizes the gain k and the coefficients c_j to k_q and $c_{j,q}$, respectively, and constructs a vector $\mathbf{k_q} = \mathbf{c_q} - k_q\mathbf{e}_2$, where $\mathbf{e}_2$ is the second column of the identity matrix of size $p+1$. Then, obfuscates $\mathbf{k_q}$ to $\mathcal{E}_r(\mathbf{k_q})$, and then sends $\mathcal{E}_r(\mathbf{k_q})$ to the controller node (at every time the sensitivity changes).
- **P:** Quantizes the polynomial basis of the sampled state $\mathbf{x}[t]$ to $\mathbf{x}_q[t]$, and encrypts $\mathbf{x}_q[t]$ to $\mathcal{E}_P(\mathbf{x}_q[t])$, and then sends

$\mathcal{E}_P(\mathbf{x}_q[t])$ to the controller node (at every sampling time).

C: Upon receiving the obfuscated/encrypted data, computes the scaled encrypted control inputs $\mathcal{E}_P(\mathcal{E}_r(u_q[t]))$ for $u_q[t] = \mathbf{k_q}^T\mathbf{x}_q$.

P: Upon receiving the encrypted scaled control inputs $\mathcal{E}_P(\mathcal{E}_r(u_q[t]))$, decrypts $\mathcal{E}_P(\mathcal{E}_r(u_q[t]))$ to $\mathcal{E}_r(u_q[t])$ and then $u_q[t]$.

P: Scales $u_q[t]$ to obtain the control input $u[t] = u_q[t]\Delta^2[t]$ and applies it to the plant.

C. Quantizer analysis and design

As before, we analyze the effect of quantization in the design of quantizers by considering the system in Fig. 2 without encryptors and decryptors.

With the quantized control input

$$u[t] = \bar{\alpha}_p(\bar{\mathbf{x}}[t]) - \bar{k}\bar{x}[t], \tag{36}$$

the quantized closed-loop system is

$$x[t+1] = ax[t] - b\bar{k}\bar{x}[t] + b(\alpha(x[t]) - \bar{\alpha}_p(\bar{\mathbf{x}}[t])). \tag{37}$$

With a Lyapunov function $V = (x[t])^2$, (37) implies that

$$\begin{aligned} &V[t+1] - V[t] \\ &= (ax[t] - b\bar{k}\bar{x}[t] + b(\alpha(x[t]) - \alpha(\bar{x}[t])))^2 - x^2[t] \\ &\le ((a-b\bar{k})^2 - 1)x^2[t] + 2(a-b\bar{k})y[t]x[t] + y^2[t], \end{aligned} \tag{38}$$

where $y[t] = b(\bar{k} + M)\Delta[t]/2$.

The expression (38) is negative outside the ball $\{x : |x| \le \Theta\Delta\}$, where $\Theta = (b(\bar{k} + M))/(1 - (a - b\bar{k}))$.

This time, consider using a modified version of the quantizer in Section III, i.e.,

$$\Delta[t] = \Delta_i := \Omega^i\Delta_0,\ t_0 = 0 \le t_i \le t < t_{i+1}, \tag{39}$$

where Ω is a scaling factor and t_i are the time instances of sensitivity updates.

Unlike Section III, the quantizer is initialized with the sensitivity Δ_0 and the saturation level q_{sat} such that satisfy

$$|x[0]| \le \Delta_0\left(q_{\text{sat}} - \frac{1}{2}\right),\ \Omega = (\Theta + \varepsilon + 1)\left(q_{\text{sat}} - \frac{1}{2}\right)^{-1} < 1, \tag{40}$$

with some $\varepsilon > 0$. Then, we have $\Theta < q_{\text{sat}} - 1/2$.

In order to avoid truncating k and c_j, make sure that q_{sat} is large enough satisfying

$$|k| \le (q_{\text{sat}} - 1/2)\Delta_0,\ |c_j| \le (q_{\text{sat}} - 1/2)\Delta_0,\ \forall j. \tag{41}$$

Also in order to guarantee $|a - b\bar{k}| < 1$, make sure that

$$\Delta_0 \le \varepsilon' + \frac{2\,(1 - |a - bk|)}{b},\ \varepsilon' > 0. \tag{42}$$

Choosing the time instances of sensitivity updates

$$t_{i+1} = \min\{t \ge t_i + 1 : \|q_{\Delta_i}(x[t])\| \le \Theta + \varepsilon + 1/2\}, \tag{43}$$

it holds that

$$|x[t_i]| \le \Delta_i\,(\Theta + \varepsilon + 1)\,,\ i = 1, 2, \cdots. \tag{44}$$

The existence of t_i is guaranteed using the similar analysis in [17] while k and c_j are not truncated.

This quantizer guarantees that

$$x[t] \in R_{i+1} := \{x : |x| \le \Delta_i\,|q_{\text{sat}} - 1/2|\} \tag{45}$$

for $t \in [t_i, t_{i+1})$ as long as k and c_j are not truncated, i.e.,

$$|k| \le (q_{\text{sat}} - 1/2)\Delta[t],\ |c_j| \le (q_{\text{sat}} - 1/2)\Delta[t],\ \forall j. \tag{46}$$

However, as $\Delta[t]$ approaches to zero, two problems occur:

- the quantized values of k and c_j will be truncated no matter how large q_{sat} is chosen, and
- the required upper bound ε_1' for the function approximation (31) approaches to zero, which possibly leads to an infinitely large p.

Therefore, asymptotic stability cannot be guaranteed. On the other hand, we may hold the sensitivity $\Delta[t]$ constant once it becomes sufficiently small to avoid the above two problems. In other words, using the quantizer in the form of

$$\Delta[t] = \begin{cases} \Delta_i := \Omega^i\Delta_0, t_0 \le t_i \le t < t_{i+1}, i = 1, \cdots, f \\ \Delta_f := \Omega^f\Delta_0, t_f \le t, \end{cases} \tag{47}$$

we can guarantee the practical stability of the system with

$$x[t] \in R_{f+1} := \{x : |x| \le \Delta_f\,|q_{\text{sat}} - 1/2|\}\,,\ t \ge t_f, \tag{48}$$

without incurring the problem.

We may also choose to use a time-invariant quantizer in the gain node to guarantee the practical stability, in which case, the region that $x[t]$ will stay depends on the sensitivity of the quantizer in the gain node.

VI. Conclusions

In this paper, the control systems combined with quantizers and encryptors/decryptors are proposed and investigated. It is shown that encrypted control systems can be constructed that achieve asymptotic stability for linear systems, and practical stability for some nonlinear systems with the aid of function approximations using Weierstrass approximation theorem. Since the combination with quantizers allows us to choose short key length, the processing time for encryption/decryption may be reduced for the sake of cipher strength.

Acknowledgment

This research was supported by the grant from Okawa Foundation for Information and Telecommunications.

The author would like to thank Prof. Kiminao Kogiso at the University of Electro-Communications for his comments on the manuscript.

References

[1] H. Sandberg, S. Amin, and K. H. Johansson, "Cyberphysical security in networked control systems: An introduction to the issue," *IEEE Control Systems Magazine*, vol. 35, no. 1, pp. 20–23, 2015.

[2] L. Zhang, H. Gao, and O. Kaynak, "Network-induced constraints in networked control systems – a survey," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 403–416, 2013.

[3] F. Bullo, J. Cortés, and S. Martínez, *Distributed Control of Robotic Networks*, ser. Applied Mathematics Series. Princeton University Press, 2009, electronically available at http://coordinationbook.info.

[4] P. Antsaklis and J. Baillieul, "Special issue on technology of networked control systems," *Proc. of the IEEE*, vol. 95, no. 1, pp. 5–8, 2007.

[5] T. Tanaka, M. Skoglund, H. Sandberg, and K. H. Johansson, "Directed information and privacy loss in cloud-based control," in *American Control Conferenc*, 2017, pp. 1666–1672.

[6] 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.

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

[8] Z. Huang, S. Mitra, and G. Dullerud, "Differentially private iterative synchronous consensus," in *ACM Workshop on Privacy in the Electronic Society*. New York, NY, USA: ACM, 2012, pp. 81–90.

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

[10] Y. Mo and R. M. Murray, "Privacy preserving average consensus," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 753–765, 2017.

[11] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, 1978.

[12] 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.

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

[14] P. Paillier, *Public-Key Cryptosystems Based on Composite Degree Residuosity Classes*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 223–238.

[15] F. Farokhi, I. Shames, and N. Batterham, "Secure and private control using semi-homomorphic encryption," *Control Engineering Practice*, vol. 67, no. Supplement C, pp. 13 – 20, 2017.

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

[17] R. W. Brockett and D. Liberzon, "Quantized feedback stabilization of linear systems," *IEEE Transactions on Automatic Control*, vol. 45, no. 7, pp. 1279–1289, 2000.

[18] F. Kerschbaum, "Secure and sustainable benchmarking in clouds," *Business & Information Systems Engineering*, vol. 3, no. 3, pp. 135–143, Jun 2011.

[19] Http://python-paillier.readthedocs.io/en/latest/index.html.

[20] W. Heemels, K. Johansson, and P. Tabuada, "An introduction to event-triggered and self-triggered control," in *IEEE Conference on Decision and Control*, 2012, pp. 3270–3285.

[21] H. K. Khalil, *Nonlinear systems*. Upper Saddle River, NJ: Prentice Hall, 2001.

[22] V. Lakshmikantham, S. Leela, and A. A. Martynyuk, *Practical Stability of Nonlinear Systems*. Teaneck, NJ: World Scientific, 1990.

[23] D. Estep, *Practical Analysis in One Variable*. New York, NY: Springer-Verlag New York, 2002.