Control Engineering Practice 145 (2024) 105874

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Encrypted distributed model predictive control of nonlinear processes ☆

Yash A. Kadakia [a], Fahim Abdullah [a], Aisha Alnajdi [b], Panagiotis D. Christofides [a,b,*]

[a] Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA, 90095-1592, USA
[b] Department of Electrical and Computer Engineering, University of California, Los Angeles, CA 90095-1592, USA

ARTICLE INFO

Keywords:
Distributed model predictive control
Encrypted control
Cybersecurity
Nonlinear systems
Process control

ABSTRACT

In this research, we present an encrypted iterative distributed model predictive controller (DMPC) to enhance the computational efficiency and cybersecurity of large-scale nonlinear processes. In this configuration, a single large process is divided into numerous smaller subsystems, each regulated by a unique Lyapunov-based MPC (LMPC) that utilizes the complete process model and exchanges control inputs with other LMPCs to address the interactions between subsystems. Further, to enhance cybersecurity, all communication links between sensors, actuators, and control input computing units are encrypted. Through a comprehensive stability analysis of the encrypted iterative DMPC, bounds are established on errors arising from encrypted communication links, disturbances, and the sample-and-hold implementation of controllers. Practical aspects such as reducing data encryption time by appropriate key length choices, sampling interval criterion, and quantization parameter selection are discussed. Simulation results of the proposed control scheme, applied to a nonlinear chemical process, showcase its effective closed-loop performance in the presence of sensor noise and process disturbances. Specifically, a non-Gaussian noise distribution is obtained from an industrial data set and added to the state measurements to justify the practical effectiveness of the proposed approach.

1. Introduction

In recent years, networks have emerged as pivotal components within manufacturing systems, replacing traditional point-to-point communications across various levels. At the field level, networks have elevated connectivity among sensors, actuators, and controllers, enabling efficient data transfer within the factory floor, while concurrently reducing wiring and minimizing potential points of failure. At the supervisory and management level, networks have facilitated automated plant-wide communication via SCADA (Supervisory Control and Data Acquisition) systems. This has, in turn, expanded data storage capacities and visibility, enabling operational trend analysis and improved decision-making for enhanced closed-loop performance, and has augmented interconnectivity of various parts of the plant.

However, these advantages come with a substantial reliance on networked communications, whether through the internet or wireless local area networks (LAN), which could be vulnerable to cyber threats. Any compromise within these systems could lead to significant consequences, such as critical service disruption, physical harm, financial loss, and potential threats to public safety. Real-world cyberattack instances underscore the need of cybersecurity measures in networked cyber–physical systems. For instance, the 2015 cyberattack on Ukraine's power grid managed by SCADA controls, led to widespread power outages (Khan, Maynard, McLaughlin, Laverty, & Sezer, 2016). Similarly, in 2021, hackers launched a DarkSide ransomware attack on Colonial Pipeline by encrypting networked communication and demanded a ransom for decryption keys. This incident forced Colonial Pipeline to halt operations, leading to extensive disruptions in fuel distribution (Tsvetanov & Slaria, 2021).

PID (proportional–integral-Derivative) controllers and PLCs (Programmable Logic Controllers) have been extensively used and continue to be utilized for controlling system states in a decentralized manner. Their decentralized operation reduces computational burden and interdependencies between different controllers. However, in systems with highly coupled process states, where the control inputs applied by one controller directly impacts the controlled states of another controller, traditional controllers might not yield adequate closed-loop performance. To overcome this constraint, complex processes have been effectively managed using model predictive controllers (MPCs). MPCs utilize a mathematical model of the process, obtained from either first-principles or data, to predict future closed-loop state evolution within a defined horizon. Subsequently, control inputs are optimized based on real-time sensor feedback, considering interactions between

☆ This paper was not presented at any IFAC meeting.

* Corresponding author at: Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA, 90095-1592, USA.
E-mail addresses: yash14@g.ucla.edu (Y.A. Kadakia), fa2@g.ucla.edu (F. Abdullah), aisha218@g.ucla.edu (A. Alnajdi), pdc@seas.ucla.edu (P.D. Christofides).

https://doi.org/10.1016/j.conengprac.2024.105874
Received 18 November 2023; Received in revised form 31 December 2023; Accepted 27 January 2024

all states and inputs. This methodology enhances control precision while minimizing utility costs.

For systems regulated by MPCs, at each sampling instance, a nonlinear optimization problem has to be solved to compute optimal control input trajectories, which can be very complex for large-scale systems involving numerous states and control inputs. To cope with this, distributed MPCs have been proposed (Liu, de la Peña, & Christofides, 2010). Networked communication has facilitated distributed control systems to be easily established within a SCADA control architecture by enhancing connectivity and data transfer between different controllers without needing elaborate wired communication. However, as mentioned earlier, this has also made the system more vulnerable to cyberattacks with the evolution of technology. Considerable research efforts have been dedicated to areas such as employing linear encrypted controllers (Darup, 2020; Darup, Redder, & Quevedo, 2018), developing machine learning-based cyberattack detectors (Al-Abassi, Karimipour, Dehghantanha, & Parizi, 2020; Dutta, Choraś, Pawlicki, & Kozik, 2020), utilizing encrypted decentralized MPCs (Kadakia, Alnajdi, Abdullah and Christofides, 2023), and creating cyberattack-resilient controllers (Paridari et al., 2017).

Addressing the aforementioned challenges, this work focuses on an encrypted iterative distributed MPC comprising a set of Lyapunov-based MPCs, utilizing encrypted networked communication for communication between sensors, actuators, and computing units responsible for calculating the control inputs. Following the formulation of the proposed control system, a thorough stability analysis is conducted to establish bounds, ensuring system stabilization within the desired stability region. Closed-loop simulations of the encrypted distributed LMPC system implemented in a nonlinear chemical process network are presented and compared with an encrypted centralized LMPC.

2. Preliminaries

2.1. Notation

The symbol $\|\cdot\|$ denotes the Euclidean norm of a vector, and $x^\top$ represents the transpose of a vector x. The sets of real numbers, integers, and natural numbers are denoted by $\mathbb{R}$, $\mathbb{Z}$, and $\mathbb{N}$, respectively. The additive groups of integers modulo M are represented by $\mathbb{Z}_M$. The symbol "\" denotes set subtraction, where $A\backslash B$ represents the set of elements in set A but not within set B. A function, $f(\cdot)$, is classified as C^1 when it is continuously differentiable in a defined domain. The least common multiple of the integers i and j is denoted by $\text{lcm}(i,j)$. The greatest common divisor that divides i and j with no remainder is denoted by $\gcd(i,j)$.

2.2. Class of systems

In this research, we consider a general category of nonlinear systems regulated by multiple unique sets of control inputs. Each distinct set of control inputs manages a particular subsystem of the process. To simplify notations, we examine two subsystems – subsystem 1 and subsystem 2 – each governed solely by u_1 and u_2, respectively. However, this same analysis can be extended to any nonlinear system controlled by N_{sys} subsystems regulated by N_{sys} unique sets of manipulated inputs. While partitioning a large-scale nonlinear process, manipulated inputs that have a strong, direct effect on certain states should be grouped together and be manipulated by the same controller. The work of Rocha, Oliveira-Lopes, and Christofides (2018) describes such methods in detail, and can be an area of future research. The overall nonlinear system is characterized by a set of ordinary differential equations (ODEs), formulated in the following manner:

$$\begin{aligned}\dot{x} &= f(x(t), u_1(t), u_2(t), w(t))\\ y &= x + v\end{aligned} \tag{1}$$

The state vector is denoted by $x \in \mathbb{R}^n$, while $y \in \mathbb{R}^n$ is the vector of state measurements that are sampled continuously. $u_1 \in \mathbb{R}^{m_1}$ and $u_2 \in \mathbb{R}^{m_2}$ represent the sets of control inputs, $w \in \mathbb{R}^w$ is the disturbance vector, and $v \in \mathbb{R}^n$ is the noise vector. The control input constraints are defined by $u_1 \in U_1 := \{u_{1_i}^{\min} \le u_{1_i} \le u_{1_i}^{\max}, i = 1, \dots, m_1\}, \subset \mathbb{R}^{m_1}$, and $u_2 \in U_2 := \{u_{2_i}^{\min} \le u_{2_i} \le u_{2_i}^{\max}, i = 1, \dots, m_2\}, \subset \mathbb{R}^{m_2}$. $u = [u_1\ u_2]^\top \in U$ is the bounded control input vector formed by concatenating u_1 and u_2. The vector function $f(\cdot)$ is locally Lipschitz with respect to its arguments. We consider $f(0,0,0,0) = 0$, such that the steady state of Eq. (1) is the origin. Without loss of generality, we set the initial time to zero ($t_0 = 0$). $S(\Delta)$ is defined as a collection of piece-wise constant functions characterized by an interval of Δ.

2.3. Stability assumptions

Accounting for interactions between the partitioned subsystems of the nonlinear process, we assume the existence of stabilizing control laws $u_1 = \Phi_1(x) \in U_1$, $u_2 = \Phi_2(x) \in U_2$ which regulate subsystems 1 and 2, respectively, such that the system of Eq. (1) with $w \equiv 0$ and $v \equiv 0$ is rendered exponentially stable, signifying the existence of a C^1 control Lyapunov function $V(x)$ that satisfies the subsequent inequalities for all $x \in \mathbb{R}^n$ within D, which is an open region around the origin:

$$c_1|x|^2 \le V(x) \le c_2|x|^2, \tag{2a}$$

$$\frac{\partial V(x)}{\partial x} f(x, \Phi_1(x), \Phi_2(x), 0) \le -c_3|x|^2, \tag{2b}$$

$$\left|\frac{\partial V(x)}{\partial x}\right| \le c_4|x|, \tag{2c}$$

where c_i, $i = \{1,2,3,4\}$ are positive constants. In the nonlinear system of Eq. (1), the closed-loop stability region can be defined as Ω_ρ, which is a level set of the control Lyapunov function V. In particular, $\Omega_\rho := \{x \in D | V(x) \le \rho\}$, where $\rho > 0$. Thus, starting from any initial condition in Ω_ρ, the control inputs $\Phi_1(x)$ and $\Phi_2(x)$ guarantee that the state trajectory of the closed-loop system remains inside Ω_ρ. Further, based on the Lipschitz property of $f(x,u_1,u_2,w)$ and the bounds on u_1, u_2, and w, the subsequent inequalities hold for all $x \in \Omega_\rho, u_1 \in U_1, u_2 \in U_2$ and $w \in W$ with positive constants M_F and L'_w:

$$|f(x,u_1,u_2,w)| \le M_F, \tag{3a}$$

$$\left|\frac{\partial V}{\partial x} f(x,u_1,u_2,w) - \frac{\partial V}{\partial x} f(x,u_1,u_2,0)\right| \le L'_w|w|. \tag{3b}$$

2.4. Paillier cryptosystem

In this study, we utilize the Paillier cryptosystem (Paillier, 1999) in order to encrypt all signals that are transmitted through the networked communication established. While we do not leverage the semi-homomorphic nature of the additive homomorphism within the Paillier cryptosystem, it is incorporated to enable the integration of conventional controllers, like PI (proportional–integral) controllers, that can calculate control inputs in an encrypted space, within the control architecture if needed (Kadakia, Suryavanshi, Alnajdi, Abdullah, & Christofides, 2024). Prior to encryption, we generate the public key (for encryption) and the private key (for decryption) and can be outlined as follows:

(1) Choose two large prime integers (p and q) randomly, such that, $\gcd((p-1)(q-1), pq) = 1$.
(2) Define, $M = pq$.
(3) Select a random integer $\hat{g} \in \mathbb{Z}_{M^2}$, where $\mathbb{Z}_{M^2}$ is the multiplicative group of integers modulo M^2.
(4) Compute $\lambda = \text{lcm}(p-1, q-1)$.
(5) Specify $\hat{L}(x) = (x-1)/M$.
(6) Verify the existence of the subsequent modular multiplicative inverse: $u = (\hat{L}(\hat{g}^\lambda \bmod M^2))^{-1} \bmod M$.

(7) If the inverse does not exist, revisit step 3 and select an alternate value of $\hat{g}$. If the inverse exists, $(M, \hat{g})$ is the public key and (λ, u) is the private key.

Upon obtaining the keys, authorized recipients receive the public key for encryption and the private key for decryption. Encryption is executed in the following manner:

$$E_M(m, r) = c = \hat{g}^m r^M \mod M^2 \tag{4}$$

where r is an integer randomly chosen from the set $\mathbb{Z}_M$, and c denotes the resulting ciphertext by encrypting m. Decryption is executed in the following manner:

$$D_M(c) = m = \hat{L}(c^\lambda \mod M^2)u \mod M \tag{5}$$

Remark 1. Traditional approaches, such as mapping floating points to a set or applying mathematical transformations to achieve data privacy, may prove inadequate in practice. When these methods are used during steady-state operation, it results in the transmission of identical values. On the other hand, during encryption, a distinct random number is generated each time data is encrypted. This feature ensures that encrypting identical numbers results in different ciphertexts, thereby significantly enhancing cybersecurity measures. While encryption enhances privacy, it also enhances cybersecurity by protecting the system against intelligent cyberattacks as discussed in the work of Kadakia et al. (2024).

2.5. Quantization

For the utilization of the Paillier cryptosystem, the data intended for encryption is required to be in the form of natural numbers within $\mathbb{Z}_M$. However, prior to encryption, the signal values exist in floating-point format. Thus, we implement quantization, to map the floating-point numbers into $\mathbb{Z}_M$ (Darup, Redder, Shames, Farokhi, & Quevedo, 2017). Employing a signed fixed-point binary representation, we establish a set, $\mathbb{Q}_{l_1,d}$, characterized by parameters l_1 and d. The parameter l_1 is defined as the total bit count (integer and fractional), and d denotes the fractional bits. The number of fractional bits represent the number of bits used to represent the fractional part of the floating point data. It is equal to the quantization parameter. The $\mathbb{Q}_{l_1,d}$ set encompasses rational numbers ranging from -2^{l_1-d-1} to $2^{l_1-d-1} - 2^{-d}$, with increments of 2^{-d}. For a number q in $\mathbb{Q}_{l_1,d}$, there exists $\beta \in \{0,1\}^{l_1}$, such that $q = -2^{l_1-d-1}\beta l_1 + \sum_{i=1}^{l_1-1} 2^{i-d-1}\beta_i$. The function $g_{l_1,d}$ maps a real-number data point a to the set $\mathbb{Q}_{l_1,d}$, and is defined by the following equation,

$$\begin{aligned} g_{l_1,d} &: \mathbb{R} \to \mathbb{Q}_{l_1,d} \\ g_{l_1,d}(a) &:= \arg\min_{q \in \mathbb{Q}_{l_1,d}} |a - q| \end{aligned} \tag{6}$$

Following this, the quantized data undergoes a transformation into a set of positive integers ($\mathbb{Z}_M$) via a bijective mapping ($f_{l_2,d}$), as detailed in (Darup et al., 2017):

$$\begin{aligned} f_{l_2,d} &: \mathbb{Q}_{l_1,d} \to \mathbb{Z}_{2^{l_2}} \\ f_{l_2,d}(q) &:= 2^d q \mod 2^{l_2} \end{aligned} \tag{7}$$

In the encryption process, integer plaintext messages from the set $Z_{2^{l_2}}$ are transformed into ciphertexts, and can then be decrypted back to set $Z_{2^{l_2}}$. To retrieve the original data point belonging to the set $\mathbb{Q}_{l_1,d}$, we define an inverse mapping denoted as $f^{-1}_{l_2,d}$ in the following manner:

$$f^{-1}_{l_2,d} : \mathbb{Z}_{2^{l_2}} \to \mathbb{Q}_{l_1,d} \tag{8}$$

$$f^{-1}_{l_2,d}(m) := \frac{1}{2^d} \begin{cases} m - 2^{l_2} & \text{if } m \geq 2^{l_2-1} \\ m & \text{otherwise} \end{cases} \tag{9}$$

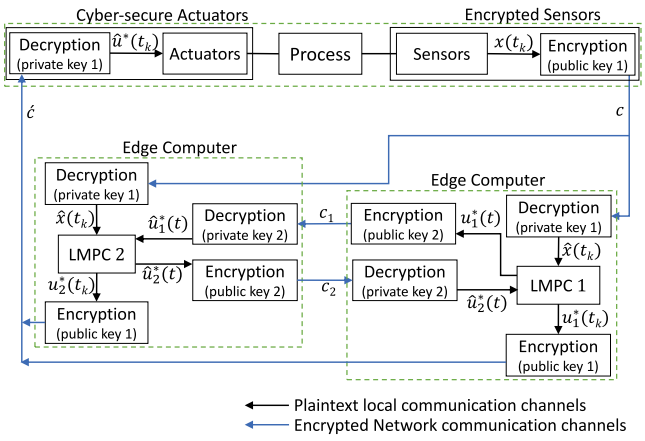

Fig. 1. Block diagram of the encrypted iterative distributed LMPC system.

3. Development of the encrypted iterative distributed LMPC

3.1. Design of the encrypted iterative distributed LMPC

Fig. 1 illustrates the control structure of the encrypted iterative distributed LMPC, where all LMPCs collaboratively optimize control actions for their respective subsystems. The sampling period represents the time between two consecutive measurements during which a constant control input is maintained by the actuators in a sample-and-hold manner. A total of two LMPCs that utilize the complete process model for computing a set of distinct control inputs has been considered to present the control strategy. A single iteration of an LMPC corresponds to an optimal control input computation by an LMPC, which may be repeated with updated input information from the other LMPC, if the termination criterion is not satisfied. The control strategy can be implemented through the following steps:

(1) At time $t = t_k$, where k is the current sampling instance, using public key 1, signals $x(t_k)$ from sensors are encrypted to ciphertext c and transmitted to the computing units of distinct control subsystems.
(2) In each unit, using private key 1, the encrypted signals are decrypted. The resulting quantized states $\hat{x}(t_k)$ initialize the LMPC model.
(3) At iteration $z = 1$, LMPC 1 computes the optimal control input trajectory $u_1^*(t)$, using the quantized states $\hat{x}(t_k)$, and assuming the stabilizing control law $u_2(t) = \Phi_2(\hat{x}(t))$ for the second subsystem, for $t \in [t_k, t_{k+N})$, where N is the prediction horizon. In parallel, LMPC 2 computes the optimal control input trajectory $u_2^*(t)$ assuming $u_1(t) = \Phi_1(\hat{x}(t))$, the stabilizing controller for the first subsystem, for $t \in [t_k, t_{k+N})$.
(4) At the end of the first iteration, LMPC 1 and LMPC 2 encrypt their computed control inputs over the prediction horizon using public key 2, to the ciphertexts c_1 and c_2, respectively. Subsequently, LMPC 1 decrypts c_2 to obtain the quantized control input of LMPC 2, $\hat{u}_2^*(t)$, and LMPC 2 decrypts c_1 to obtain the quantized control input of LMPC 1, $\hat{u}_1^*(t)$, for $t \in [t_k, t_{k+N})$.
(5) At iteration $z = 2$, both LMPCs recalculate the optimal control inputs of their subsystem using the quantized control inputs (after decryption) of the other subsystems. Subsequently, the new control input trajectories are again shared with the other LMPCs, as described previously. The aforementioned steps are reiterated till a termination condition is satisfied, which could be the number of iterations, or the difference between computed control inputs in successive iterations is less than a specified threshold value.

(6) Upon meeting this termination condition, both LMPCs encrypt their optimal control inputs for the subsequent sampling period (utilizing public key 1) and transmit the ciphertexts to the corresponding actuators of each of their respective subsystem.
(7) At the actuator, the ciphertext $\acute{c}$ undergoes decryption using private key 1 to yield $\hat{u}^*(t_k)$, the quantized input, which is applied to the process.

By encrypting all signals in a distributed setting between the sensors, controllers, and actuators, secure information exchange is established between computing units situated at various locations, eliminating the necessity of a control room.

Remark 2. In the proposed design, sensor-controller and controller-actuator communication links utilize distinct keys for encryption–decryption compared to the inter-controller communication link. However, a single pair of keys may also serve this purpose. The decision to choose a distinct set of keys aims to meet the specific cybersecurity requirements based on the cyber–physical needs of the system. For instance, in transmitting encrypted signals across the entire plant, higher bit length keys would be recommended. Conversely, when exchanging encrypted signals between various controllers or computing units, lower bit length keys might be sufficient.

Remark 3. The time and computational load required for encrypting signals increases with longer key bit lengths. A 2048-bit key results in an approximately 4096-bit ciphertext and requires about 0.066 s for encryption. On the other hand, a 1024-bit key produces a ciphertext of around 2048 bits within 0.0096 s. The most recent NIST recommendations suggest using asymmetric keys of 2048 bits, an upgrade from the previous recommendation of 1024 bits (Barker & Barker, 2019). The determination of key lengths should be guided by factors like cyber–physical vulnerability, desired cybersecurity level, and available computational resources. The time estimates were derived from encryption processes on an Intel i7-10700K 3.80 GHz computer with 64 GB of RAM. The computational complexity of encryption–decryption varies by $\mathcal{O}(\bar{k}^3)$ as mentioned in the work of Damgård, Jurik, and Nielsen (2010), where $\bar{k}$ represents the number of bits of the keys utilized. Thus, increasing the bits of the keys significantly increases the computational load for encryption–decryption.

Remark 4. In an industrial setting, the standard approach for encryption would involve employing microcontrollers within sensors and actuators to encrypt and decrypt signals, respectively. Encrypted signals are sent to the controllers via RF (radio-frequency) transmission modules. Similarly, actuators receive signals from the RF receiver module. To decrease the total computation time for encryption–decryption, large-scale systems can equip individual sensors and actuators with dedicated microcontrollers and RF modules. This setup enables parallel operations for the transmission and reception of encrypted signals.

Remark 5. A ciphertext encrypted using a 2048-bit key will be roughly 4096 bits or 512 bytes (1 byte = 8 bits). Wireless communication standards like Wi-Fi 4, Wi-Fi 5, and Wi-Fi 6 offer bandwidths ranging from hundreds of megabytes per second (MBps) to a few gigabytes per second. This bandwidth is more than adequate to transmit multiple encrypted ciphertexts at each sampling instance. For instance, a 4096-bit ciphertext would require approximately 1 microsecond for transmission through Wi-Fi with a bandwidth of 500 MBps. Therefore, the transmission of encrypted signals would not significantly burden established communication channels, while reinforcing the cybersecurity of the control system.

Remark 6. To deal with input delays, a state-predictor can be integrated. The state predictor would estimate the state values after the period corresponding to the input delay and the LMPC model would be initialized with these predicted states. This has been demonstrated in the work of Kadakia, Alnajdi et al. (2023) using an encrypted decentralized LMPC. As the LMPC is initialized with the new predicted states, the same concept can be extended to the encrypted distributed LMPC presented in this research.

3.2. Quantization errors in the control architecture

The closed-loop configuration presented in Fig. 1 introduces two error sources: one originating from state quantization in the sensor-controller link, while another stemming from control input quantization within the controller-actuator link, which are bounded as follows:

$$|x(t_k) - \hat{x}(t_k)| \leq 2^{-d-1} \tag{10a}$$

$$|u(t_k) - \hat{u}(t_k)| \leq 2^{-d-1} \tag{10b}$$

The upper bounds for the quantization error in Eq. (10) has been derived in Kadakia, Suryavanshi, Alnajdi, Abdullah and Christofides (2023). Leveraging the local Lipschitz property, the error for the stabilizing controller of the j^{th} subsystem will be bounded by the following equation, where L'_j is a positive constant, for $x \in \Omega_\rho$, the stability region:

$$|\boldsymbol{\Phi}_j(\hat{x}) - \boldsymbol{\Phi}_j(x)| \leq L'_j|\hat{x} - x| \leq L'_j 2^{-d-1} \tag{11}$$

3.3. Encrypted iterative distributed LMPC system

The optimization task for the j^{th} LMPC in the iterative distributed LMPC, during the initial iteration ($z = 1$), is formulated as:

$$\mathcal{J} = \min_{u_j \in S(\Delta)} \int_{t_k}^{t_{k+N}} L(\tilde{x}(t), \boldsymbol{\Phi}_m(\tilde{x}(t)), u_j(t))\, \mathrm{d}t,$$

$$\text{where } m = 1, 2 \text{ and } m \neq j \tag{12a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = f(\tilde{x}(t), \boldsymbol{\Phi}_m(\tilde{x}(t)), u_j(t)) \tag{12b}$$

$$u_j(t) \in U_j, \ \forall\ t \in [t_k, t_{k+N}) \tag{12c}$$

$$\tilde{x}(t_k) = \hat{x}(t_k) \tag{12d}$$

$$\dot{V}(\hat{x}(t_k), \boldsymbol{\Phi}_m(\hat{x}(t_k)), u_j(t_k)) \leq \dot{V}(\hat{x}(t_k), \boldsymbol{\Phi}_m(\hat{x}(t_k)), \boldsymbol{\Phi}_j(\hat{x}(t_k))),$$

$$\text{if } \hat{x}(t_k) \in \Omega_\rho \backslash \Omega_{\rho_{\min}} \tag{12e}$$

$$V(\tilde{x}(t)) \leq \rho_{\min}, \ \forall\ t \in [t_k, t_{k+N}),$$

$$\text{if } \hat{x}(t_k) \in \Omega_{\rho_{\min}} \tag{12f}$$

For subsequent iterations $z > 1$, following the exchange of the optimal control inputs $u_m^*(t)$ with all the other LMPCs, the optimization task for the j^{th} LMPC is:

$$\mathcal{J} = \min_{u_j \in S(\Delta)} \int_{t_k}^{t_{k+N}} L(\tilde{x}(t), \hat{u}_m(t), u_j(t))\, \mathrm{d}t,$$

$$\text{where } m = 1, 2 \text{ and } m \neq j \tag{13a}$$

$$\text{s.t.} \quad \dot{\tilde{x}}(t) = f(\tilde{x}(t), \hat{u}_m(t), u_j(t)) \tag{13b}$$

$$u_j(t) \in U_j, \ \forall\ t \in [t_k, t_{k+N}) \tag{13c}$$

$$\tilde{x}(t_k) = \hat{x}(t_k) \tag{13d}$$

$$\dot{V}(\hat{x}(t_k), \hat{u}_m(t_k), u_j(t_k)) \leq \dot{V}(\hat{x}(t_k), \boldsymbol{\Phi}_m(\hat{x}(t_k)), \boldsymbol{\Phi}_j(\hat{x}(t_k))),$$

$$\text{if } \hat{x}(t_k) \in \Omega_\rho \backslash \Omega_{\rho_{\min}} \tag{13e}$$

$$V(\tilde{x}(t)) \leq \rho_{\min}, \ \forall\ t \in [t_k, t_{k+N}),$$

$$\text{if } \hat{x}(t_k) \in \Omega_{\rho_{\min}} \tag{13f}$$

The key contrast between Eqs. (12) and (13) is that in the former, the j^{th} LMPC computes the optimal control inputs for its respective subsystem by assuming the stabilizing control laws for the remaining

subsystems, while in the latter, the LMPC uses quantized control inputs of other LMPCs (after decryption) from the previous iteration, to calculate the optimal inputs for its subsystem. $\tilde{x}$ denotes the state trajectory predicted by the LMPC model. The quantized states, denoted as $\hat{x}$, from Eqs. (12d) and (13d), initialize the LMPC model for predicting the state trajectory in accordance with Eqs. (12b) and (13b), respectively. This prediction is used to calculate the integral of the cost functions represented by Eqs. (12a) and (13a), respectively, to determine the optimized control inputs, $u_j^*(t)$, throughout the prediction horizon. However, the LMPC transmits only the first control input of the sequence which is applied to the system by the actuator within the interval $t \in [t_k, t_{k+1})$, where this process is repeated at each sampling period. Here, k is the sampling instance, while N denotes the number of sampling periods in the prediction horizon. The constraints of Eqs. (12c) and (13c) bound the control inputs, and it remains consistent across all iterations for a particular subsystem. The Lyapunov constraint of Eqs. (12e) and (13e) bounds the state $x(t_k)$ at time t_k within the region $\Omega_\rho \setminus \Omega_{\rho_{\min}}$, where $\rho_{\min}$ is a level set of V in proximity to the origin. Eqs. (12f) and (13f) ensure that the closed-loop state is bounded within $\Omega_{\rho_{\min}}$ once it enters $\Omega_{\rho_{\min}}$.

Remark 7. In the LMPC formulation presented, the LMPCs transmit only the control inputs to be implemented by the actuators over the next sampling period. To address challenges related to delayed and/or asynchronous signals, a control logic can be integrated. In instances where sensor signals are absent, the LMPC transmits the control input calculated for the subsequent sampling period during the preceding instance, ensuring continuous operation. This adaptive strategy can be selectively applied by subsystems experiencing signal reception issues within a distributed system. Moreover, a control logic can be devised to transmit the control inputs after the first iteration, if challenges arise in communicating control inputs with other controllers, switching from a distributed to a decentralized setup. Consequently, the utilization of distributed MPC introduces substantial flexibility to adapt control systems according to diverse conditions and practical requirements, all without necessitating extensive modifications.

3.4. Robustness of the encrypted distributed LMPC

In this subsection, we will conduct a comprehensive stability analysis of the nonlinear system of Eq. (1), considering bounded process disturbances. Initially, we ascertain the closed-loop stability using the encrypted stabilizing controllers $\hat{\boldsymbol{\Phi}}_1(\hat{x})$ and $\hat{\boldsymbol{\Phi}}_2(\hat{x})$, and subsequently, we extend our results to evaluate system stability under the encrypted iterative distributed LMPC defined by Eqs. (12) and (13).

Theorem 1. *We consider the system of Eq. (1) with bounded disturbances $|w| \leq w_m$, to examine the closed-loop system stability under the encrypted stabilizing controllers $\hat{\boldsymbol{\Phi}}_1(\hat{x})$ and $\hat{\boldsymbol{\Phi}}_2(\hat{x})$. The stabilizing controllers $\boldsymbol{\Phi}_1(x)$ and $\boldsymbol{\Phi}_2(x)$, without encryption, complies with the inequalities stated in Eq. (2). Also, the initial state x_0 is assumed to be within the region $\Omega_{\hat{\rho}}$ where $\hat{\rho} < \rho$. For a sufficiently large time $T > 0$, where T is defined as the time taken by $x(t)$ to enter $\Omega_{\rho_{\min}}$, the positive real numbers $L'_x, L'_{e_1}, L'_{e_2}, M_F, L'_w$, $e_1 = (L_1+1)2^{-d-1}$, and $e_2 = (L_2+1)2^{-d-1}$ can be determined, for which Δ, w, d, and $\epsilon_w > 0$ exist, such that the subsequent conditions are met:*

$$\begin{aligned} &L'_x M_F \Delta + L'_{e_1}|e_1| + L'_{e_2}|e_2| + L'_w|w| - \frac{c_3}{c_2}\rho_s \leq -\epsilon_w \\ &\rho_{\min} = \max\{V(x(t+\Delta)) | V(x(t)) \leq \rho_s\} \end{aligned} \tag{14}$$

where $\rho > \hat{\rho} > \rho_{\min} > \rho_s$. Then, $x(t)$, under the encrypted stabilizing controller, is within $\Omega_{\hat{\rho}}$ and ultimately converges to $\Omega_{\rho_{\min}}$ for $t \geq T$.

Proof. The time-derivative of the control Lyapunov function for the nonlinear system (Eq. (1)) with bounded disturbances under the stabilizing control law is:

$$\begin{aligned} \dot{V} &= \frac{\partial V(x(t))}{\partial x} f(x(t), \hat{\boldsymbol{\Phi}}_1(\hat{x}(t_k)), \hat{\boldsymbol{\Phi}}_2(\hat{x}(t_k)), w) \\ &= \frac{\partial V(x(t))}{\partial x} f(x(t), \hat{\boldsymbol{\Phi}}_1(\hat{x}(t_k)), \hat{\boldsymbol{\Phi}}_2(\hat{x}(t_k)), w) \\ &\quad - \frac{\partial V(x(t))}{\partial x} f(x(t), \hat{\boldsymbol{\Phi}}_1(\hat{x}(t_k)), \hat{\boldsymbol{\Phi}}_2(\hat{x}(t_k)), 0) \\ &\quad + \frac{\partial V(x(t))}{\partial x} f(x(t), \hat{\boldsymbol{\Phi}}_1(\hat{x}(t_k)), \hat{\boldsymbol{\Phi}}_2(\hat{x}(t_k)), 0). \end{aligned} \tag{15}$$

Based on the Lipschitz condition in Eqs. (2) and (3b), the subsequent inequality holds:

$$\dot{V} \leq \frac{\partial V(x(t))}{\partial x} f(x(t), \hat{\boldsymbol{\Phi}}_1(\hat{x}(t_k)), \hat{\boldsymbol{\Phi}}_2(\hat{x}(t_k)), 0) + L'_w|w| \tag{16}$$

Substituting the error bounds resulting due to quantization, as derived in Eq. (10),

$$\begin{aligned} \dot{V} \leq \frac{\partial V(x(t))}{\partial x} &f(x(t), \boldsymbol{\Phi}_1(\hat{x}(t_k)) \\ &+ 2^{-d-1}, \boldsymbol{\Phi}_2(\hat{x}(t_k)) + 2^{-d-1}, 0) + L'_w|w| \end{aligned} \tag{17}$$

Further, $\boldsymbol{\Phi}_j(\hat{x}(t_k)) = \boldsymbol{\Phi}_j(\hat{x}(t_k)) - \boldsymbol{\Phi}_j(x(t_k)) + \boldsymbol{\Phi}_j(x(t_k))$ for $j = \{1, 2\}$. Using the Lipschitz property, $\boldsymbol{\Phi}_j(\hat{x}(t_k)) - \boldsymbol{\Phi}_j(x(t_k)) \leq L_j|\hat{x} - x| \leq L_j 2^{-d-1}$. When we substitute this in Eq. (17), we get:

$$\begin{aligned} \dot{V} \leq \frac{\partial V(x(t))}{\partial x} &f(x(t), \boldsymbol{\Phi}_1(x(t_k)) + e_1, \boldsymbol{\Phi}_2(x(t_k)) + e_2, 0) \\ &+ L'_w|w| \end{aligned} \tag{18}$$

where $e_1 = (L_1+1)2^{-d-1}$ and $e_2 = (L_2+1)2^{-d-1}$ represent the error bounds from quantization. From the constraints stated in Eq. (2), we can re-write Eq. (18) as:

$$\begin{aligned} \dot{V} \leq \frac{\partial V(x(t))}{\partial x} &f(x(t), \boldsymbol{\Phi}_1(x(t_k)) + e_1, \boldsymbol{\Phi}_2(x(t_k)) + e_2, 0) \\ &- \frac{\partial V(x(t_k))}{\partial x} f(x(t_k), \boldsymbol{\Phi}_1(x(t_k)), \boldsymbol{\Phi}_2(x(t_k)), 0) \\ &+ \frac{\partial V(x(t_k))}{\partial x} f(x(t_k), \boldsymbol{\Phi}_1(x(t_k)), \boldsymbol{\Phi}_2(x(t_k)), 0) \\ &+ L'_w|w| \end{aligned} \tag{19}$$

From Eq. (19), we can define $g(x, e_1, e_2) = f(x, \boldsymbol{\Phi}_1(x) + e_1, \boldsymbol{\Phi}_2(x) + e_2, 0)$. In addition, the positive constants, L'_x, L'_{ξ}, and L'_q exist, such that the subsequent Lipschitz inequality holds for all $x, x' \in \Omega_{\hat{\rho}}$:

$$\begin{aligned} &\left| \frac{\partial V(x)}{\partial x} g(x, e_1, e_2) - \frac{\partial V(x')}{\partial x} g(x', 0, 0) \right| \leq \\ &L'_x|x - x'| + L'_{e_1}|e_1| + L'_{e_2}|e_2| \end{aligned} \tag{20}$$

Hence, we can re-write Eq. (19) as:

$$\begin{aligned} \dot{V} &\leq \frac{\partial V(x(t))}{\partial x} g(x(t), e_1, e_2) - \frac{\partial V(x(t_k))}{\partial x} g(x(t_k), 0, 0) \\ &\quad - c_3|x(t_k)|^2 + L'_w|w| \\ &\leq L'_x|x(t) - x(t_k)| + L'_{e_1}|e_1| + L'_{e_2}|e_2| \\ &\quad - c_3|x(t_k)|^2 + L'_w|w| \end{aligned} \tag{21}$$

From the continuity property of $x(t)$ $\forall\, t \in [t_k, t_k + \Delta)$, we have $|x(t) - x(t_k)| \leq M_F \Delta, \forall t \in [t_k, t_k + \Delta)$. Utilizing this bound and from the inequalities of Eq. (2), we can re-write Eq. (21) as follows:

$$\dot{V} \leq L'_x M_F \Delta + L'_{e_1}|e_1| + L'_{e_2}|e_2| + L'_w|w| - \frac{c_3}{c_2}\rho_s \tag{22}$$

In Eq. (22), the first term signifies the error stemming from the sample-and-hold control input implementation, the second and third terms denote quantization errors due to encryption, and the fourth term indicates the error from process disturbances. The aforementioned errors are constrained and can be effectively minimized by utilizing a lower sampling time and a higher quantization parameter for encryption. As a result, the combined sum of these is also constrained and can be rendered suitably small. Hence, there exist positive real numbers Δ, d, and ϵ_w, such that the following inequality holds for all $t \in [0, T]$:

$$L'_x M_F \Delta + L'_{e_1}|e_1| + L'_{e_2}|e_2| + L'_w|w| - \frac{c_3}{c_2}\rho_s \leq -\epsilon_w$$

implying that $\dot{V} \leq -\epsilon_w$ for any $x(t_k) \in \Omega_{\hat{\rho}} \setminus \Omega_{\rho_s}$ for all $t_k \in [0, T]$. Thus, upon satisfying the conditions of Eq. (14), under the encrypted stabilizing controller, the closed-loop system state is confined in $\Omega_{\hat{\rho}}$ and converges within $\Omega_{\rho_s} \subseteq \Omega_{\rho_{\min}}$ in time T, and stays within the desired stability region.

Now, we advance to the stability analysis of the closed-loop system employing the encrypted distributed LMPC.

Theorem 2. *We consider the system of Eq. (1) with bounded disturbances $|w| \leq w_m$, to examine the closed-loop stability under the encrypted iterative distributed LMPCs of Eqs. (12) and (13). The initial state x_0 is assumed to be within $\Omega_{\hat{\rho}}$. Utilizing the results derived in Theorem 1, and preserving our earlier assumption that $\rho > \hat{\rho} > \rho_{\min} > \rho_s$, if the ensuing conditions are met,*

$$\dot{V} \leq L'_x M_F \Delta + L'_{e_1}|e_1| + L'_{e_2}|e_2| + L'_w|w| - \frac{c_3}{c_2}\rho_s \leq -\epsilon_w$$
$$\rho_{\min} = \max\{V(x(t+\Delta)) | V(x(t)) \leq \rho_s\} \tag{23}$$

then the closed-loop state $x(t)$ remains inside $\Omega_{\hat{\rho}}$ and is ultimately bounded within $\Omega_{\rho_{\min}}$ for $t \geq T$, by implementing the encrypted iterative distributed LMPCs of Eqs. (12) and (13).

Proof. First, we establish the feasibility of the optimization problem associated with each LMPC in the encrypted distributed LMPC system, for all the states bounded within $\Omega_{\hat{\rho}}$. Subsequently, with the optimized control inputs from the encrypted distributed LMPC, we will demonstrate that the closed-loop state of Eq. (1) is bounded and converges to the stability region $\Omega_{\hat{\rho}}$, thereby extending the findings presented from Theorem 1. If $x(t_k) \in \Omega_{\hat{\rho}} \setminus \Omega_{\rho_{\min}}$, the input trajectories, $u_j(t)$, where $j = \{1, 2\}$ for $t \in [t_k, t_{k+1})$ are feasible solutions of the optimization problem of each LMPC, as these trajectories satisfy the constraints of Eqs. (12c) and (13c), as well as the Lyapunov constraints of Eqs. (12e) and (13e). Additionally, if $x(t_k) \in \Omega_{\rho_{\min}}$, the control inputs $u_j(t)$, $j = \{1, 2\}$ meet the constraints imposed in Eqs. (12c) and (13c), as well as the Lyapunov constraints of Eqs. (12f) and (13f); hence, the predicted states by the LMPC model are bounded within $\Omega_{\rho_{\min}}$. Thus, for all $x_0 \in \Omega_{\hat{\rho}}$, the LMPC optimization problems of Eqs. (12) and (13) can be solved recursively for all iterations with feasible solutions as $x(t) \in \Omega_{\hat{\rho}}$ for all times.

Next, we establish that for any $x_0 \in \Omega_{\hat{\rho}}$, the state of the closed-loop system remains bounded within $\Omega_{\hat{\rho}}$ for all times, and given a sufficiently large time $T > 0$, it converges to a small neighborhood $\Omega_{\rho_s} \subseteq \Omega_{\rho_{\min}}$ and remains there. Under the encrypted iterative distributed LMPC system, the time derivative of the control Lyapunov function is:

$$\dot{V} = \frac{\partial V(x(t))}{\partial x} f(x(t), \hat{u}_1(t_k), \hat{u}_2(t_k), w) \tag{24}$$

From the Lyapunov constraint of Eqs. (12e) and (13e), the following inequality holds:

$$\begin{aligned}\dot{V} &= \frac{\partial V(x(t))}{\partial x} f(x(t), \hat{u}_1(t_k), \hat{u}_2(t_k), w) \\ &\leq \frac{\partial V(x(t))}{\partial x} f(x(t), \hat{\Phi}_1(\hat{x}(t_k)), \hat{\Phi}_2(\hat{x}(t_k)), w)\end{aligned} \tag{25}$$

However, extending the results of Theorem 1, the time-derivative of the control Lyapunov function under the encrypted iterative distributed LMPC can be bounded as follows:

$$\begin{aligned}&\frac{\partial V(x(t))}{\partial x} f(x(t), \hat{u}_1(t_k), \hat{u}_2(t_k), w) \leq L'_x M_F \Delta \\ &+ L'_{e_1}|e_1| + L'_{e_2}|e_2| + L'_w|w| - \frac{c_3}{c_2}\rho_s \leq -\epsilon_w\end{aligned} \tag{26}$$

Hence, for the selected time T, there exist positive real numbers d, Δ, and ϵ_w, such that the subsequent inequality holds $\forall\, t \in [0, T]$,

$$\dot{V} \leq L'_x M_F \Delta + L'_{e_1}|e_1| + L'_{e_2}|e_2| + L'_w|w| - \frac{c_3}{c_2}\rho_s \leq -\epsilon_w$$

which implies that $\dot{V} \leq -\epsilon_w$ for any $x(t_k) \in \Omega_{\hat{\rho}} \setminus \Omega_{\rho_s}$ for all $t_k \in [0, T]$. This confirms that when the conditions of Eq. (23) are satisfied, the closed-loop system state remains consistently bounded within $\Omega_{\hat{\rho}}$. Furthermore, it converges to $\Omega_{\rho_s} \subseteq \Omega_{\rho_{\min}}$ within time T and stays there. With this, the proof for the stability of the system under the encrypted distributed LMPC is concluded.

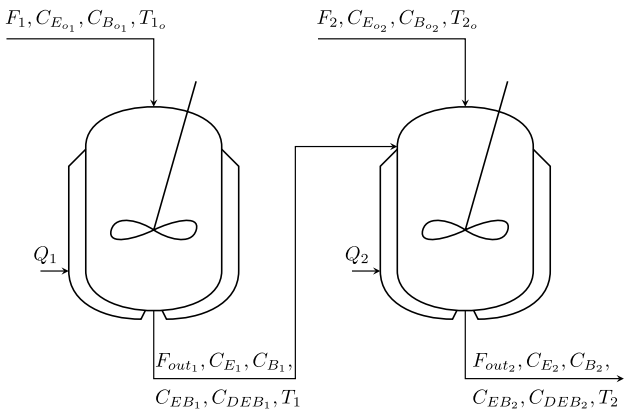

Fig. 2. Process schematic of the two CSTR network.

4. Application to a nonlinear chemical process network operating at an unstable steady state

In this section, we demonstrate the application of the proposed encrypted iterative distributed LMPC system to a nonlinear chemical process that is to be operated at an unstable steady state.

4.1. Process description and model development

The process considered involves the production of ethylbenzene (EB) through the reaction of ethylene (E) and benzene (B) in two separate non-isothermal continuous stirred tank reactors (CSTRs), connected in series, as illustrated in Fig. 2. The principal reaction, referred to as "primary", is a second-order, irreversible, and exothermic reaction, accompanied by two additional side reactions. The chemical reactions can be described as follows:

$$\mathrm{C_2H_4} + \mathrm{C_6H_6} \rightarrow \mathrm{C_8H_{10}} \quad \text{(primary)} \tag{27a}$$
$$\mathrm{C_2H_4} + \mathrm{C_8H_{10}} \rightarrow \mathrm{C_{10}H_{14}} \tag{27b}$$
$$\mathrm{C_6H_6} + \mathrm{C_{10}H_{14}} \rightarrow 2\mathrm{C_8H_{10}} \tag{27c}$$

Comprehensive information on the first-principles-based dynamic model, including equations, model parameter values, and steady-state values are provided in Kadakia, Suryavanshi et al. (2023). The state variables are the concentration of ethylene, benzene, ethylbenzene, diethylbenzene, and the reactor temperature for each CSTR in deviation terms, that is: $x^\top = [C_{E_1} - C_{E_{1s}},\ C_{B_1} - C_{B_{1s}},\ C_{EB_1} - C_{EB_{1s}},\ C_{DEB_1} - C_{DEB_{1s}},\ T_1 - T_{1s},\ C_{E_2} - C_{E_{2s}},\ C_{B_2} - C_{B_{2s}},\ C_{EB_2} - C_{EB_{2s}},\ C_{DEB_2} - C_{DEB_{2s}},\ T_2 - T_{2s}]$. The subscript "$s$" denotes the steady-state value. We create two distributed LMPCs to control the overall process. LMPC 1 optimizes the control inputs $u_1 = [C_{E_{o1}} - C_{E_{o1s}}, C_{B_{o1}} - C_{B_{o1s}}, Q_1 - Q_{1s}]^\top$. These inputs are bounded by the closed sets $[-3, 3]$ kmol/m^3, $[-3, 3]$ kmol/m^3, and $[-10^4, 2\times10^3]$ kW, respectively. LMPC 2 optimizes the control inputs $u_2 = [C_{E_{o2}} - C_{E_{o2s}}, C_{B_{o2}} - C_{B_{o2s}}, Q_2 - Q_{2s}]^\top$. These manipulated inputs are bounded by the closed sets, $[-2.5, 2.5]$ kmol/m^3, $[-2.5, 2.5]$ kmol/m^3, and $[-1.5 \times 10^4, 5 \times 10^3]$ kW, respectively. The primary goal is to manage both CSTRs at their unstable equilibrium point by utilizing the encrypted iterative distributed LMPC system. This involves the use of quantized states and control inputs for the purposes of computation and actuation.

4.2. Encrypting the distributed control system

Prior to integrating encryption and decryption into a process, the parameters d, l_1, and l_2 are chosen, considering the extreme feasible states and inputs. This involves deriving the integer bit count $l_1 - d$. In the $\mathbb{Q}_{l_1,d}$ set, the upper limit is $2^{l_1-d-1} - 2^{-d}$, and the lower limit is -2^{l_1-d-1}. Within the set, rational numbers are separated by a resolution of 2^{-d}. The quantization parameter d, representing the fractional bit count, is determined by the desired precision level and the range of state and input values. l_2 is chosen to exceed l_1. For the case discussed in this section, $l_1 - d = 16$, and subsequently, l_1 and d are fixed. Next, $d = 8$ is chosen for simulations. Accordingly l_1 is 24, and l_2 is 30. Encryption (Paillier cryptosystem) is implemented using Python's "phe" module, PythonPaillier (CSIRO's Data61, 2013). To solve the multi-constrained, non-convex optimization task of the LMPCs, the IPOPT software (Wächter & Biegler, 2006) in Python is utilized.

The termination criterion for the distributed LMPCs was set to 2 iterations. Thus, control inputs are exchanged only once with the other LMPC, at the end of the first iteration. For the computation of the control cost of the distributed LMPCs, the integration step is set to $h_c = 10^{-2} \times \Delta$. We assume a control Lyapunov function of the form $V = x^\top P x$, where P is a positive definite matrix chosen as diag;[200 200 400 1000 2.5 250 250 200 1000 0.5], through extensive simulations. Autocorrelated noise, represented as $w_k = 0.75 \times w_{k-1} + \xi_k$, was introduced to the inlet flow rates, F_1 and F_2, but the liquid level remains constant in both CSTRs at all times. Here, $k = 1, 2, \ldots$ denotes discrete time steps of $10^{-2} \times \Delta$, ξ_k is a randomly generated normally distributed variable with zero mean, and a standard deviation of 5% of the inlet flow rates. The prediction horizon of both LMPC is set to two sampling periods. The stability region is set as $\rho = 1800$, while $\rho_{\min} = 2$ represents the smaller region within which the closed-loop system state is desired to be bounded. The distributed LMPC cost function is defined as $L(x,u) = x^\top Q x + u^\top R u$, where Q = diag;[1000 1000 1500 5 8 1000 1000 3000 5 110] and R = diag;[2.1 1.95 1.5×10^{-5} 10 10 0.5×10^{-4}]. As the undesired byproduct, di-ethylbenzene, is present in minimal quantities in both CSTRs, its trajectories are not illustrated. Non-Gaussian measurement noise obtained from the noise distribution in Luo (2023) extracted from industrial data, is added to all the measured states. As this noise is normalized, it was scaled by 2% of the operating steady-state value for each concentration state, and no scaling was applied for the temperature states.

It must be ensured that the sampling time (Δ) exceeds the combined time needed for encryption–decryption of the states and control inputs, along with the time required by the LMPCs to compute the control inputs at each sampling instance for the given quantization parameter. In mathematical terms,

$$\Delta > \max(\text{encryption–decryption time}) + \max(\text{Control input computation time}) \tag{28}$$

The control inputs are applied in a sample-and-hold manner throughout the sampling period. As long as the time required for computing control inputs and encryption–decryption is shorter than the sampling period, no lag in the control variables would occur. As explained in Remarks 3 and 4, the time needed to encrypt–decrypt states and inputs depends on the bit lengths of the keys, number of microcontrollers, and RF modules used, and hence can be decided accordingly. For simulation purposes, 1024-bit length keys were used for encrypted communication between controllers, and 2048-bit length keys were utilized for all other encrypted communications. Considering the above criteria, assuming all encryption–decryption operations to be performed in series, although it can be done in parallel, the sampling time Δ was selected as 30 s in this example. Based on the constraint of Eq. (28), the encrypted distributed LMPC can only be implemented in systems that allow us to use sufficiently large sampling times that also stabilize the system as per the constraint of Eq. (26). Eqs. (12e) and (13e) are Lyapunov constraints that ensure that the time-derivative of the control Lyapunov function is more negative under the encrypted distributed LMPC than the stabilizing controller for the control input applied over the next sampling period. The future control input computed by the LMPC beyond the next sampling period may not yield a more negative time-derivative of the control Lyapunov function. Hence, we have utilized the stabilizing controller for the other subsystems in the first iteration. Moreover, as the system is operated at an unstable equilibrium, stability is critical. Alternatively, the neighboring LMPCs can utilize the future control inputs when the system is operated at a stable equilibrium.

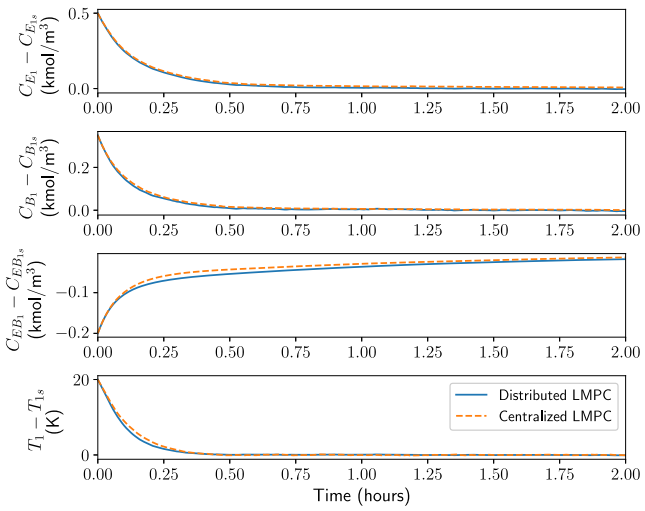

Fig. 3. State trajectories of CSTR 1 under the encrypted iterative distributed LMPC (blue solid line) and encrypted centralized LMPC (orange dashed line).

4.3. Simulation results of the encrypted distributed LMPC system

Figs. 3, 4, and 5 depict the results of the encrypted iterative distributed LMPC against the encrypted centralized LMPC. The normalized sum of the cost function for the encrypted distributed and centralized LMPCs was 0.9795 and 1, respectively. Also, the average computational time needed to compute the optimal control inputs by the distributed LMPC system and the centralized LMPC was 7.33 s and 13.14 s, respectively. Thus, not only did the distributed LMPC provide better closed-loop performance, but it also reduced the average computational time significantly compared to the centralized LMPC. This is evident with the fewer oscillations observed in the control input trajectories of the encrypted distributed LMPC in Fig. 5. No significant difference was observed in the closed-loop state trajectories in both cases, as visible in Figs. 3 and 4. Nonetheless, in both cases, the system successfully converges within $\Omega_{\rho_{\min}}$ in approximately 1.5 h of process time. We note that the time of convergence to the steady state for the desired product ethylbenzene is longer in the second CSTR as it starts with a low initial concentration; this time may be reduced by modifying the second reactor design to adjust the residence time to speed up the second reactor dynamics.

Remark 8. For the encrypted distributed LMPC investigated in this research, encryption–decryption of data as depicted in Fig. 1 leads to errors due to quantization. Kadakia, Suryavanshi et al. (2023) emphasized the potential for these errors to surpass plant/model mismatch errors in cases where distinct models are utilized in the controlled process and the LMPC. To mitigate the error caused by quantization, a higher quantization parameter d was recommended. Adopting $d = 8$ resulted in nearly indistinguishable closed-loop results with encryption when compared to those without encryption. Therefore, a quantization parameter of $d = 8$ was uniformly applied in all simulations conducted in this study.

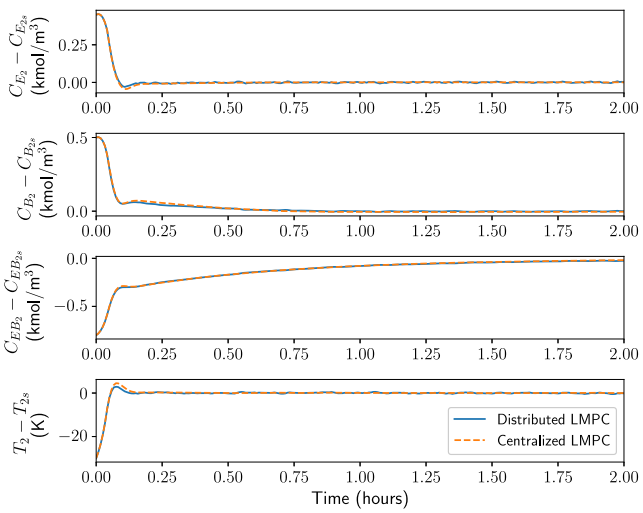

Fig. 4. State trajectories of CSTR 2 under the encrypted iterative distributed LMPC (blue solid line) and encrypted centralized LMPC (orange dashed line).

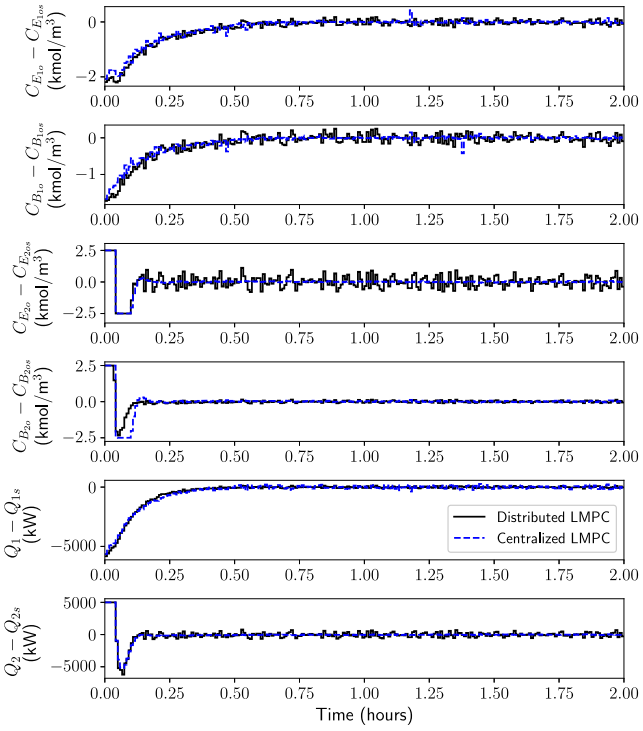

Fig. 5. Control input trajectories under the encrypted iterative distributed LMPC (black solid line) and encrypted centralized LMPC (blue dashed line).

5. Conclusion

In this research, we formulated an encrypted iterative distributed LMPC system employing encrypted signals for data transmission between sensors, controllers, and actuators. Following a comprehensive stability analysis, we determined bounds for errors from quantization, process disturbances, and the sample-and-hold implementation of the controller. With these bounds, the system could be stabilized within the desired stability region. Selection of encryption–decryption key lengths, quantization parameters, sampling time criterion, and potential methods to decrease the encryption–decryption time were discussed to facilitate practical implementation. Closed-loop simulations were performed, comparing the proposed control scheme against the encrypted centralized LMPC. Non-Gaussian sensor noise obtained from an industrial data set and process disturbances were used to demonstrate the industrial relevance and suitability of the proposed approach. The results favor the use of the encrypted distributed LMPC system, which not only improves closed-loop performance but also significantly reduces the computational time needed to calculate the control input, positioning the encrypted iterative distributed LMPC as an effective solution for improving closed-loop performance, decreasing computational time, and enhancing cybersecurity in large-scale nonlinear systems.

CRediT authorship contribution statement

Yash A. Kadakia: Investigation, Methodology, Software, Writing – original draft. **Fahim Abdullah:** Investigation, Methodology, Writing – original draft. **Aisha Alnajdi:** Investigation, Methodology, Writing – original draft. **Panagiotis D. Christofides:** Funding acquisition, Investigation, Methodology, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

Financial support from the National Science Foundation, USA, CBET-2227241, is gratefully acknowledged.

References

Al-Abassi, Abdulrahman, Karimipour, Hadis, Dehghantanha, Ali, & Parizi, Reza M (2020). An ensemble deep learning-based cyber-attack detection in industrial control system. *IEEE Access*, *8*, 83965–83973.

Barker, Elaine, & Barker, William (2019). *Recommendation for key management, part 2: best practices for key management organization*. National Institute of Standards and Technology.

CSIRO's Data61 (2013). Python paillier library. https://github.com/data61/python-paillier.

Damgård, Ivan, Jurik, Mads, & Nielsen, Jesper Buus (2010). A generalization of paillier's public-key system with applications to electronic voting. *International Journal of Information Security*, *9*, 371–385.

Darup, Moritz Schulze (2020). Encrypted MPC based on ADMM real-time iterations. *IFAC-PapersOnLine*, *53*, 3508–3514.

Darup, Moritz Schulze, Redder, Adrian, & Quevedo, Daniel E. (2018). Encrypted cloud-based MPC for linear systems with input constraints. *IFAC-PapersOnLine*, *51*, 535–542.

Darup, Moritz Schulze, Redder, Adrian, Shames, Iman, Farokhi, Farhad, & Quevedo, Daniel (2017). Towards encrypted MPC for linear constrained systems. *IEEE Control Systems Letters*, *2*, 195–200.

Dutta, Vibekananda, Choraś, Michał, Pawlicki, Marek, & Kozik, Rafał (2020). A deep learning ensemble for network anomaly and cyber-attack detection. *Sensors*, *20*, *4583*.

Kadakia, Yash A, Alnajdi, Aisha, Abdullah, Fahim, & Christofides, Panagiotis D (2023). Encrypted decentralized model predictive control of nonlinear processes with delays. *Chemical Engineering Research and Design*, *200*, 312–324.

Kadakia, Yash A, Suryavanshi, Atharva, Alnajdi, Aisha, Abdullah, Fahim, & Christofides, Panagiotis D (2023). Encrypted model predictive control of a nonlinear chemical process network. *Processes*, *11*(8), 2501.

Kadakia, Yash A, Suryavanshi, Atharva, Alnajdi, Aisha, Abdullah, Fahim, & Christofides, Panagiotis D (2024). Integrating machine learning detection and encrypted control for enhanced cybersecurity of nonlinear processes. *Computers & Chemical Engineering*, *180*, Article 108498.

Khan, Rafiullah, Maynard, Peter, McLaughlin, Kieran, Laverty, David, & Sezer, Sakir (2016). Threat analysis of BlackEnergy malware for synchrophasor based real-time control and monitoring in smart grid. In *Proceedings of the 4th international symposium for ICS & SCADA cyber security research* (pp. 1–11). Belfast, United Kingdom.

Liu, Jinfeng, de la Peña, David Muñoz, & Christofides, Panagiotis D (2010). Distributed model predictive control of nonlinear systems subject to asynchronous and delayed measurements. *Automatica*, *46*(1), 52–61.

Luo, Junwei (2023). *Machine learning modeling for process control and electrochemical reactor operation* (Ph.D. thesis), University of California, Los Angeles.

Paillier, Pascal (1999). Public-key cryptosystems based on composite degree residuosity classes. In *Proceedings of the international conference on the theory and applications of cryptographic techniques* (pp. 223–238). Berlin, Heidelberg: Springer.

Paridari, Kaveh, O'Mahony, Niamh, Mady, Alie El-Din, Chabukswar, Rohan, Boubekeur, Menouer, & Sandberg, Henrik (2017). A framework for attack-resilient industrial control systems: Attack detection and controller reconfiguration. *Proceedings of the IEEE*, *106*(1), 113–128.

Rocha, Rosiane R, Oliveira-Lopes, Luís Cláudio, & Christofides, Panagiotis D (2018). Partitioning for distributed model predictive control of nonlinear processes. *Chemical Engineering Research and Design*, *139*, 116–135.

Tsvetanov, Tsvetan, & Slaria, Srishti (2021). The effect of the colonial pipeline shutdown on gasoline prices. *Economics Letters*, *209*, Article 110122.

Wächter, Andreas, & Biegler, Lorenz T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming*, *106*, 25–57.