AIMS Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 10(11): 25175–25192.
DOI: 10.3934/math.20251114
Received: 21 August 2025
Revised: 30 September 2025
Accepted: 21 October 2025
Published: 03 November 2025

Research article

Controller design for cyber-physical systems with inputs via logical networks

Guodong Zhao and Haitao Li*

School of Mathematics and Statistics, Shandong Normal University, Jinan 250000, China

* **Correspondence:** Email: lihaitao_sdu@163.com.

Abstract: This paper investigated the robustness of cyber-physical systems (CPSs) under controller design. First, a formal definition of robustness for CPSs with inputs was established. Second, a systematic framework was proposed, featuring two novel algorithms that transform both infinite and finite systems into equivalent logical dynamical networks with inputs. Subsequently, the robustness of these systems was rigorously analyzed under the influence of the designed controllers. Finally, a numerical example was provided to illustrate the controller design process that ensures robustness for the considered CPSs.

Keywords: logical network; cyber-physical system; switched system; robustness; semi-tensor product of matrices; input-output dynamical stability
Mathematics Subject Classification: 37N35, 93D99

1. Introduction

Since CPSs were first proposed, they have given rise to great mass research fervor. So far, the research of CPSs has been applied to many emerging fields, such as smart grids [1, 2], smart vehicles [3, 4], cloud manufacturing [5], medical monitors [6], transportation logistics [7], transnational e-commerce [8], and so on.

CPSs are multidimensional complex systems. A CPS consists of two parts with interactive relations: one is the finite system representing the cyber part, and the other is the infinite system representing the physical part. In this paper, the input of the CPS is divided into two parts: one is the control input and the other is the disturbance input. Varieties of properties of CPSs have important research value, such as adaptability, security, functionality, and so on. This paper studies its robustness. It describes the ability of the system to function normally in the presence of disturbance. The definitions of robustness are numerous, such as input-to-state stable [9], input-to-state dynamical stablility [10] and so on. Robustness deliberated in this paper is the classical input-output dynamical stablility [11].

In recent years, a novel matrix product called the semi-tensor product (STP) of matrices has been proposed [12] for the representation and analysis of Boolean networks (BNs) and Boolean control networks (BCNs). This methodology has enabled breakthroughs in observability, controllability, stability, and stabilization of BNs/BCNs [13–15].

The definition of robustness proposed by Rungger and Tabuada in [11] is generalized to analyze the robustness of the CPS. It is another version of input-to-state stability [16], which is the well-known notion of robustness for control systems. See more details in [17, 18]. On the premise of this work, we apply the STP method to investigate set stabilization in logical control networks to design a state-feedback controller of the logical network with inputs. Furthermore, the method of controller design for the given CPS is given.

Furthermore, based on the aforementioned results, an interesting idea naturally comes up. Logical networks not only behave like finite systems but also have more mathematical tools, the STP, for example, over finite systems to investigate. This paper aims at studying the concept of controller design for CPSs. With the help of the STP, we try to "convert" the given CPS to a logical control network. Thus, we investigate the controller design problem of the logical network rather than the given CPS.

The main results are illustrated as follows: (1) Two algorithms respectively convert finite systems and infinite systems with cost functions into logical control networks with cost functions. (2) The controller design for logical control networks is transformed into the controller design of finite and infinite systems. (3) A sufficient and necessary condition reveals the preservation of the robustness between logical network with inputs with cost functions and the original system with inputs with cost functions.

This paper is arranged as follows: Section 2 presents the preliminaries used in this article. Section 3 gives the main results of the problem we investigated. Section 4 shows an example to illustrate the effectiveness of the results obtained in Section 5.

Notations: $\mathbb{R}_{n\times t}$ denotes the sets of $n \times t$ real matrices. $Col_j(L)$ is denoted as the jth column of matrix L. $\mathcal{K}$ is a set of functions $\mu : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ with $\mu(0) = 0$, which are continuous and strictly increasing. $\mathcal{KLD}$ is denoted as the set of functions $\nu : \mathbb{R}_{\geq 0} \times \mathbb{N} \to \mathbb{R}_{\geq 0}$ with $\nu(0, t) = 0$ for all $t \in \mathbb{N}$, which are strictly increasing with respect to the first parameter and strictly decreasing to zero with respect to the second parameter and satisfy $\nu(r, t + s) = \nu(\nu(r, s), t)$ and $\nu(r, t) = r$ for all $r \in \mathbb{R}_{\geq 0}$ and $s, t \in \mathbb{N}$. $\Delta_k := \{\delta_k^i | i = 1, 2, \cdots, k\}$, where δ_k^i defines the ith column of the $k-$dimensional identity matrix I_k. $\mathcal{L}_{m\times n}$ denotes the set of $m\times n$ logical matrix, if their elements are denoted by $\delta_m[i_1\ i_2\ \cdots\ i_n]$. $(L)_{m,n}$ denotes the elements of L in the mth row and nth column. For $a, b \in \mathbb{R}$ with $a \leq b$, we define the open, half-open, and closed intervals in $\mathbb{R}$ as (a, b), $[a, b)$ and $(a, b]$, $[a, b]$, respectively. For $a, b \in \mathbb{Z}$ with $a \leq b$, we use $(a; b)$, $[a; b)$ and $(a; b]$, $[a; b]$ to denote the corresponding intervals in $\mathbb{Z}$. $\otimes$ is the Kronecker product.

2. Preliminaries

In the beginning, the basic definition for the STP of two matrices is presented. Then, the definition is followed by some useful mathematic tools in the theory of the STP.

Definition 2.1. *The semi-tensor product of two matrices is defined as* $X \ltimes Y = (X \otimes I_{\omega/t})(Y \otimes I_{\omega/p})$, *where* $X \in \mathbb{R}_{n\times t}$, $Y \in \mathbb{R}_{p\times q}$, $\omega = lcm(t, p)$. $lcm(t, p)$ *in the above definition is the lowest common multiple of* t

and p.

The STP is the extension of the general matrices product. We will omit $\ltimes$ without confusion in the sequel.

Lemma 2.2. *[12] We are given a pseudo-logical (or logical) function $g : \Delta_{\Pi_{i=1}^n k_i} \to \mathbb{R}$ (or $g : \Delta_{\Pi_{i=1}^n k_i} \to \Delta_m$). There exists a unique structural matrix $V_g \in \mathbb{R}_{1\times\Pi_{i=1}^n k_i}$ (or $V_g \in \mathcal{L}_{m\times\Pi_{i=1}^n k_i}$) of g satisfying*

$$g(x_1, x_2, \cdots, x_n) = V_g \ltimes_{i=1}^n x_i.$$

This paper will investigate the logical control network as follow:

$$X(t+1) = G(U^c(t), U^d(t), X(t)), \tag{2.1}$$

where $U^c \in \Delta_{k_1}$ is the control input, $U^d \in \Delta_{k_2}$ is the disturbance input, $X \in \Delta_{k_3}$ is the state, and $G : \Delta_{k_1k_2k_3} \to \Delta_{k_3}$ is a logical mapping. Applying Lemma 2.2, logical control network (2.1) can be transformed into the equivalent algebraic form in the following:

$$X(t+1) = L_G U^c(t) U^d(t) X(t),$$

where $L_G \in \mathcal{L}_{k_3\times k_1k_2k_3}$ denotes the state transition matrix.

3. Main results

In this section, we first present the underlying systems for the given CPS. Based on that, the corresponding definition of robust stabilizability (input-output dynamical stabilizablility) is also provided. Then, the next subsection attempts to figure out that how to design a controller for a finite system with inputs to make it robust. After that, the controller to make an infinite system with inputs robust is designed.

3.1. Definitions of a system with inputs and its robustness

First, we describe an underlying system with inputs for the given CPS. Unlike the existed results [15], the inputs include the control input and disturbance input.

Definition 3.1. *(Underlying System) A quadruple $S = (X_0, U, X, f)$ called a system with inputs consists of*

(i) *initial state variable set $X_0 \subseteq X$;*

(ii) *input variable set $U = (U^c, U^d)$ composed of control input set U^c and disturbance input set U^d;*

(iii) *state variable set X;*

(iv) *transition mapping $f : U \times X \rightrightarrows X$.*

Furthermore, system S is a finite system when both X and U are finite sets, or it is an infinite system. Throughout the remainder of this article, all systems have their own inputs $U = (U^c, U^d)$, unless otherwise stated.

The following example testifies the rationality for Definition 3.1.

Example 3.2. *Consider a CPS $S = (S_c, S_p)$.*

(1) The cyber part of S is system $S_c = (X_c^0, U_c, X_c, f_c)$, where $X_c^0 = X_c = \{a_1, a_2\}$, $U_c = (U_c^c, U_c^d) = (\{\top, \perp\}, \{1, 0\})$, and f_c is shown in Figure 1 when $u_c^d = 0$ and Figure 2 for $u_c^d = 1$.

(2) The physical part of S is system $S_p = (X_p^0, U_p, X_p, f_p)$, where $X_p^0 = X_p = [-2, 2]$, $X_p = [-5, 5]$, $U_p = (U_p^c, U_p^d) = ([-1, 3], [0, 2])$, and

$$f_p = \begin{cases} 0.4x_p + u_p^c + u_p^d, & x_c = a_1, \\ 0.3x_p + u_p^c + u_p^d, & x_c = a_2, \end{cases}$$

where $u_p^c \in U_p^c$, $u_p^d \in U_p^d$, $x_p \in X_p$, and $x_c \in X_c$.

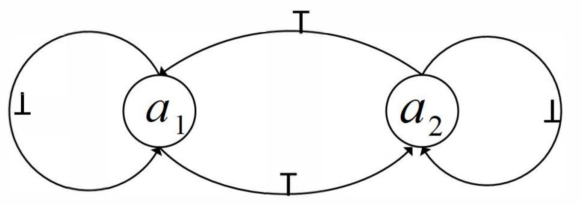

Figure 1. $u_c^d = 0$.

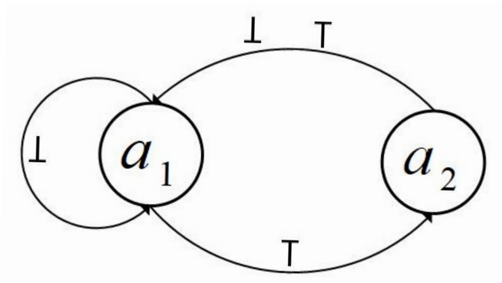

Figure 2. $u_c^d = 1$.

It is obvious that system S_c, which is the cyber part of S, is a finite system and system S_p, which is the physical part of S, is an infinite system. The whole CPS $S = (S_c, S_p)$ is considered as an infinite system.

Next, we render input and output cost functions to system S as follows:

Definition 3.3. *Triple (S, I, O), which is called a system with cost functions, is made up of:*

(i) *a system $S = (X_0, U, X, f)$, where $U = (U^c, U^d)$ holds;*

(ii) *input cost function $I : U^d \times X \to \mathbb{R}_{\geq 0}$;*

(iii) *output cost function $O : U^d \times X \to \mathbb{R}_{\geq 0}$.*

Before introducing the definition of robust stabilizability for system (S, I, O), we must confirm that input cost function I satisfies the following assumption. The following assumption depicts an obvious fact that there exists a state $u_0^d \in U^d$ denoting the situation that the given disturbance disappears.

Assumption 3.4. *Assume that there exists a state $u_0^d \in U^d$ such that, for $\forall x \in X$,*

$$I(u_0^d, x) = 0 \tag{3.1}$$

holds, where input cost function I is defined in Definition 3.3.

Then, extending the definition of robustness for the general system in reference [11], we get the definition of robust stabilizability for a system with inputs.

Definition 3.5. *(Robust stabilizability) We regard (S, I, O) as a system with inputs with cost functions, and φ as a $\mathcal{K}$ function, β as a $\mathcal{KLD}$ function, $\lambda \geq 0$ as a real number. S is said to be (φ,β,λ)–practically input-output dynamical stabilizable ($(\varphi,\beta,\lambda) - pIODS$), if there exists a state feedback controller $u^c(t) = g(x(t))$, such that, for $\forall$ trajectory G of S, the following inequality holds:*

$$O(u^d(t), x(t)) \leq \max_{\hat{t}\in[0;t]} \beta(\varphi(I(u^d(\hat{t}), x(\hat{t}))), t - \hat{t}) + \lambda, \tag{3.2}$$

where $\forall t \in \mathbb{N}$. Furthermore, state feedback controller $u^c(t) = g(x(t))$ is called a robust controller. (S, I, O) is called robust stabilizable.

If we combine Assumption 3.4 and Definition 3.5, we have an interesting property of robust stabilizable system (S, I, O).

Proposition 3.6. *Consider (S, I, O) as a robust stabilizable system. If (S, I, O) satisfies Assumption 3.4, then there exists a set $\emptyset \neq \Omega \subset X$ such that, for $\forall x \in \Omega$, inequality*

$$O(u_0^d, x(t)) \leq \lambda \tag{3.3}$$

holds.

Proof. By Definition 3.5, robust stabilizable system (S, I, O) implies that there exists a state feedback controller $u^c(t) = g(x(t))$, such that the inequality

$$O(u^d(t), x(t)) \leq \max_{\hat{t}\in[0;t]} \beta(\varphi(I(u^d(\hat{t}), x(\hat{t}))), t - \hat{t}) + \lambda \tag{3.4}$$

holds, where φ is a $\mathcal{K}$ function, and β is a $\mathcal{KLD}$ function, $\lambda \geq 0$ is a real number.

Assume that we let $u^d(t) = u_0^d$, where $t \geq T > 0$. It means that disturbance u^d disappears after time T. We choose new initial states $x(T)$, $u^d(T) = u_0^d$, and $u^c(T) = g(x(T))$. Then, (3.4) can be rewritten as

$$O(u_0^d, x(t)) \leq \max_{\hat{t}\in[T;t]} \beta(\varphi(I(u_0^d, x(\hat{t}))), t - \hat{t}) + \lambda. \tag{3.5}$$

Furthermore, we define a set $\Omega = \{x(t) \mid$ system trajectory $((g(x(t)), u^d(t)), x(t))$ from initial states $x(T)$, $u^d(T) = u_0^d$ and $u^c(T) = g(x(T))$, $t \geq T\}$. Then, (3.5) can be rewritten as

$$O(u_0^d, x(t)) \leq \max_{\hat{t}\in[T;t]} \beta(\varphi(I(u_0^d, x(\hat{t}))), t - \hat{t}) + \lambda = \lambda,$$

where $x(t) \in \Omega$.

The conclusion follows. □

3.2. Robust controller design

This subsection focuses on the robust controller design for the given finite systems. First, the given finite systems are converted into logical control networks by Algorithm 3.7. Then, the method of the robust controller design is constructed for the obtained logical control networks via STP. Finally, the constructed controllers are proved to be also effective for the original finite systems.

First, consider finite system $S = (X_0, U, X, f)$ with cost functions I and O, where $U = (U^c, U^d)$, $X = \{x_1, x_2, \ldots, x_{k_3}\}$, $U^c = \{u_1^c, u_2^c, \ldots, u_{k_1}^c\}$, and $U^d = \{u_1^d, u_2^d, \ldots, u_{k_2}^d\}$.

The following algorithm constructs a logical control network with cost function $(\bar{S}, \bar{I}, \bar{O})$ corresponding to (S, I, O), where $\bar{S}$ is a logical control network, and $\bar{I}$ and $\bar{O}$ are two pseudo-logical functions.

Algorithm 3.7. *(Exact simulation algorithm)*

Step 1 *Define $\bar{U}^c := \Delta_{k_1}$, $\bar{U}^d := \Delta_{k_2}$, and $\bar{X} := \Delta_{k_3}$;*

Step 2 *Define relation $R_{U^c} = \{(u_i^c, \delta_{k_1}^i) \mid i = 1, 2, \cdots, k_1\}$, $R_{U^d} = \{(u_i^d, \delta_{k_2}^i) \mid i = 1, 2, \cdots, k_2\}$, $R_U := R_{U^c} \times R_{U^d}$, and $R_X = \{(x_i, \delta_{k_3}^i) | i = 1, 2, \cdots, k_3\}$. Then, let $R := R_U \times R_X$ and we get $\bar{X}_0$ via R_X and X_0;*

Step 3 *Define $L_{\bar{f}} \in \mathcal{L}_{k_3 \times k_1 k_2 k_3}$, and let $Col_j(L_{\bar{f}}) := \delta_{k_3}^{i_j}$, for $j = 1, 2, \cdots, k_1 k_2 k_3$, where i_j satisfies both $x_{i_j} = f(u_r^c, u_s^d, x_t)$ and $(r-1)k_1 + (s-1)k_2 + t = j$;*

Step 4 *By Lemma 2.2, we get $M_{\bar{I}} \in \mathbb{R}_{1 \times k_2 k_3}$ and $M_{\bar{O}} \in \mathbb{R}_{1 \times k_2 k_3}$, which satisfy $I(u_i^d, x_j) = M_{\bar{I}} \delta_{k_2}^i \delta_{k_3}^j$ and $O(u_i^d, x_j) = M_{\bar{O}} \delta_{k_2}^i \delta_{k_3}^j$, where $i = 1, 2, \cdots, k_2$ and $j = 1, 2, \cdots, k_3$ hold.*

Step 5 *Define the logical control network with cost functions $(\bar{S}, \bar{I}, \bar{O})$ as*

$$\bar{x}(t+1) = L_{\bar{f}} \bar{u}^c(t) \bar{u}^d(t) \bar{x}(t), \tag{3.6}$$

$$\bar{I}(\bar{u}^d(t), \bar{x}(t)) = M_{\bar{I}} \bar{u}^d(t) \bar{x}(t),$$

$$\bar{O}(\bar{u}^d(t), \bar{x}(t)) = M_{\bar{O}} \bar{u}^d(t) \bar{x}(t). \tag{3.7}$$

Remark 3.8. *The computational complexity of Algorithm 3.7 is $O(k_1 k_2 k_3)$.*

Via Algorithm 3.7, it is obvious that (S, I, O) is equivalent to $(\bar{S}, \bar{I}, \bar{O})$. In the following, an exact input-output simulation relation (EIOSR) is defined to describe this equivalence relation.

Before giving the relation, we explain the symbol $U^c(x)$ as follow:

$$U^c(x) = \{u^c \in U^c \mid \forall u^d \in U^d : f((u^c, u^d), x) \neq \emptyset\}.$$

Definition 3.9. *View (S, I, O) and $(\bar{S}, \bar{I}, \bar{O})$ as two systems. Define $R = (U^c, U^d) \times (\bar{U}^c, \bar{U}^d) \times X \times \bar{X}$ as an equivalence relation. R is said to be an exact input-output simulation relation from (S, I, O) to $(\bar{S}, \bar{I}, \bar{O})$, if*

(i) $\forall x \in X, \exists \bar{x} \in \bar{X} : (x, \bar{x}) \in R_X$ *and* $\forall (u^c, u^d) \in (U^c(x), U^d), \exists (\bar{u}^c, \bar{u}^d) \in (\bar{U}^c(x), \bar{U}^d)$*:*

  (a) $((u^c, u^d), (\bar{u}^c, \bar{u}^d), x, \bar{x}) \in R$,

  (b) $\forall x' \in f((u^c, u^d), x), \exists \bar{x}' \in \bar{f}((\bar{u}^c, \bar{u}^d), \bar{x})$ *such that* $(x', \bar{x}') \in R_X$*;*

(ii) $\forall \bar{x} \in \bar{X}, \exists x \in X : (x, \bar{x}) \in R_X$ *and* $\forall (\bar{u}^c, \bar{u}^d) \in (\bar{U}^c(x), \bar{U}^d), \exists (u^c, u^d) \in (U^c(x), U^d)$*:*

(c) $((u^c, u^d), (\bar{u}^c, \bar{u}^d), x, \bar{x}) \in R$,

(d) $\forall \bar{x}' \in \bar{f}((\bar{u}^c, \bar{u}^d), \bar{x}), \exists x' \in f((u^c, u^d), x)$ *such that* $(x', \bar{x}') \in R_X$;

where R_X *means the projection of* R *on* $X \times \bar{X}$,

(iii) $\bar{I}(\bar{u}^d, \bar{x}) = I(u^d, x)$;

(iv) $\bar{O}(\bar{u}^d, \bar{x}) = O(u^d, x)$.

The following content provides the method to get a controller for a logical control network with cost functions $(\bar{S}, \bar{I}, \bar{O})$ via the theory of a reachable set in reference [19].

First of all, we look for set Ω in Proposition 3.6, which is a set where the system is stabilized. Assume disturbance $\delta_{k_2}^q$ satisfies $(u_0^d, \delta_{k_2}^q) \in R_{U^d}$. Let $\bar{u}^d(t) = \delta_{k_2}^q$ in equation (3.7). Then we can get $\bar{O}(\delta_{k_2}^q, \bar{x}(t)) = M_{\bar{O}} \delta_{k_2}^q \bar{x}(t)$. Check $Col_i(M_{\bar{O}}), i = 1, 2, \ldots, k_3$. Choose i satisfying $Col_i(M_{\bar{O}}) \leq \lambda$ and record it into set $\{i_1, i_2, \ldots, i_l\}$. Define $\bar{\Omega} = \{\bar{x}_{i_1}, \bar{x}_{i_2}, \ldots, \bar{x}_{i_l}\}$. Via an EIOSR, we can get $\Omega = \{x_{i_1}, x_{i_2}, \ldots, x_{i_l}\}$ satisfying $(x_{i_n}, \bar{x}_{i_n}) \in R_X, n = 1, 2, \ldots, l$.

Definition 3.10. *Examine logical control network (3.6).* $x_a \in \Delta_{k_3}$ *is said to be one-step reachable from* $x_b \in \Delta_{k_3}$ *under a certain disturbance, if, for certain disturbance* $\bar{u}^d \in \Delta_{k_2}$*, there exists a control* $\bar{u}^c \in \Delta_{k_1}$ *satisfying* $x_a = L_{\bar{f}} \ltimes \bar{u}^c \ltimes \bar{u}^d \ltimes x_b$.

Next, we partition $L_{\bar{f}}$ as follow:

$$L_{\bar{f}} = [L_{\bar{f}}^1 \; L_{\bar{f}}^2 \; \cdots \; L_{\bar{f}}^{k_1}],$$

where $L_{\bar{f}}^i \in \mathcal{L}_{k_3 \times k_2 k_3}, i = 1, 2, \ldots, k_1$.

For each $i \in 1, 2, \ldots, k_1$, let us divide $L_{\bar{f}}^i$ as follow:

$$L_{\bar{f}}^i = [L_{\bar{f}}^{i_1} \; L_{\bar{f}}^{i_2} \; \cdots \; L_{\bar{f}}^{i_{k_2}}],$$

where $L_{\bar{f}}^{i_j} \in \mathcal{L}_{k_3 \times k_3}, j = 1, 2, \ldots, k_2$.

For any $h \in \mathbb{Z}^+$, we inductively define a series of reachable sets under disturbance $\delta_{k_2}^q$ satisfying $(u_0^d, \delta_{k_2}^q) \in R_{U^d}$ as follows:

$$\begin{aligned} S_1(\bar{\Omega}) = \{\delta_{k_3}^\alpha :\ & \text{for certain disturbance } \delta_{k_2}^q, \text{ there exists an integer} \\ & nq \in \{1, 2, \ldots, k_1\} \text{ such that} \\ & \bigvee_{n=1}^{k_1} \bigvee_{s=1}^{l} (L_{\bar{f}}^{nq})_{i_s, \alpha} = 1\}, \end{aligned} \tag{3.8}$$

$$\begin{aligned} S_h(\bar{\Omega}) = \{\delta_{k_3}^\alpha :\ & \text{for certain disturbance } \delta_{k_2}^q, \text{ there exists an integer} \\ & nq \in \{1, 2, \ldots, k_1\} \text{ such that} \\ & \bigvee_{n=1}^{k_1} \bigvee_{\delta_{k_3}^{\alpha'} \in S_{h-1}(\bar{\Omega})} (L_{\bar{f}}^{nq})_{\alpha', \alpha} = 1\}. \end{aligned} \tag{3.9}$$

Theorem 3.11. *There exists a controller causing $(\bar{S}, \bar{I}, \bar{O})$ to transfer into $\bar{\Omega}$ under the disturbance $\delta_{k_2}^q$ if and only if there exists a positive integer $\eta \leq k_3$ such that $\bar{\Omega} \subseteq S_1(\bar{\Omega})$ and $S_\eta(\bar{\Omega}) = \Delta_{k_3}$ are satisfied, where S_1 and S_η are defined in (3.8) and (3.9), respectively.*

Proof. Necessity: Suppose controller $\bar{K} = \delta_{k_3}[n_1 \ \ n_2 \ \ \cdots \ \ n_{k_3}]$ makes $\bar{S}$ transfer into $\bar{\Omega}$. It is obvious that $x(1; x(0), \bar{K}x(0), \delta_{k_2}^q) \in \bar{\Omega}$ when $x(0) \in \bar{\Omega}$. Thus, we have $\bar{\Omega} \subseteq S_1(\bar{\Omega})$. Via the construction of reachable sets, we can get $S_1(\bar{\Omega}), S_2(\bar{\Omega}), \ldots, S_\eta(\bar{\Omega})$ which implies $S_{h-1}(\bar{\Omega}) \subseteq S_h(\bar{\Omega}), 2 \leq h \leq \eta$. For any arbitrary initial state, it will drive to $\bar{\Omega}$ under the influence of controller $\bar{K}$. As a result, there exists a positive integer $\eta \leq k_3$ such that $S_\eta(\bar{\Omega}) = \Delta_{k_3}$ is satisfied.

Sufficiency: Due to $\bar{\Omega} \subseteq S_1(\bar{\Omega})$, we see that $S_{h-1}(\bar{\Omega}) \subseteq S_h(\bar{\Omega}) \subseteq \Delta_{k_3}$ for any positive integer $h \leq \eta$, where $S_0(\bar{\Omega}) = \emptyset$. Defining $\hat{S}_h(\bar{\Omega}) = S_h(\bar{\Omega}) \backslash S_{h-1}(\bar{\Omega})$, where $h = 1, 2, \ldots, \eta$, we have $\hat{S}_{h_1}(\bar{\Omega}) \cap \hat{S}_{h_2}(\bar{\Omega}) = \emptyset$, where $h_1, h_2 \in \{1, 2, \ldots, \eta\}, h_1 \neq h_2$, and we have $\bigcup_{h=1}^{\eta} \hat{S}_h(\bar{\Omega}) = \Delta_{k_3}$. Thus, for any positive integer $\kappa \leq k_3$, there exists a unique positive integer $h_\kappa \leq \eta$ such that $\delta_{k_3}^\kappa \in \hat{S}_{h_\kappa}(\bar{\Omega})$. Moreover, via the construction of reachable sets, we can obtain that

(1) when $h_\kappa = 1$, for given disturbance $\delta_{k_2}^q$, there exists an integer $n_\kappa \in \{1, 2, \ldots, k_2\}$ such that $\bigvee_{s=1}^{l} (L_{\bar{f}}^{n_\kappa q})_{i_s, \kappa} = 1$,

(2) when $2 \leq h_\kappa \leq \eta$, for given disturbance $\delta_{k_2}^q$, there exists an integer $n_\kappa \in \{1, 2, \ldots, k_2\}$ such that $\bigvee_{\delta_{k_3}^{\alpha'} \in S_{h-1}(\bar{\Omega})} (L_{\bar{f}}^{n_\kappa q})_{\alpha', \kappa} = 1$.

Define $\bar{K} = \delta_{k_1}[n_1 \ \ n_2 \ \ \cdots \ \ n_{k_3}]$. Then under disturbance $\delta_{k_2}^q$, for any initial state $\bar{x}(0) = \delta_{k_3}^\kappa$, there exists h_κ such that $\bar{x}(h_\kappa; \delta_{k_3}^\kappa, \bar{u}^c, \delta_{k_2}^q) \in \bar{\Omega}$. Since initial state $\bar{x}(0)$ is arbitrary, we have $\bar{x}(t; \bar{x}(0), \bar{u}^c, \delta_{k_2}^q) \in \bar{\Omega}$ when $t \geq \eta$.

As a result, there exists a controller $\bar{K}$ causing $(\bar{S}, \bar{I}, \bar{O})$ to transfer into $\bar{\Omega}$.

The conclusion follows. □

It can be seen from the proof of sufficiency in Theorem 3.11 that controller $\bar{K}$ of logical control network $\bar{S}$ is constructed.

The following propositions give controller K of finite system S by controller $\bar{K}$ of logical control network $\bar{S}$ and the EIOSR.

Proposition 3.12. *Assume there exists an EIOSR between (S, I, O) and $(\bar{S}, \bar{I}, \bar{O})$. Given controller $\bar{K}$ of $(\bar{S}, \bar{I}, \bar{O})$, then controller K of (S, I, O) is obtained.*

Proof. Give an arbitrary state-input sequence of $(\bar{S}, \bar{I}, \bar{O})$ through controller $\bar{K}$ as follow:

state: $\hat{x}(0), \hat{x}(1), \cdots\cdots;$

input: $(\hat{u}^c(0), \delta_{k_2}^q), (\hat{u}^c(1), \delta_{k_2}^q), \cdots\cdots.$

Via an EIOSR, we can get a corresponding state-input sequence of (S, I, O) as follow:

state: $x(0), x(1), \cdots\cdots;$

input: $(u^c(0), u_0^d), (u^c(1), u_0^d), \cdots\cdots,$

where $((\hat{u}^c(t), \delta_{k_2}^q), (u^c(t), u_0^d), \hat{x}(t), x(t)) \in R$. So, we can get controller K via the state-input sequence of (S, I, O).

The conclusion follows. □

Proposition 3.13. *Assume there exists an EIOSR between* (S, I, O) *and* $(\bar{S}, \bar{I}, \bar{O})$. *Given controller* K *of* (S, I, O), *then controller* $\bar{K}$ *of* $(\bar{S}, \bar{I}, \bar{O})$ *is obtained.*

Proof. It is obvious, so that we omit it.

Theorem 3.14. *Assume there exists an EIOSR between* (S, I, O) *and* $(\bar{S}, \bar{I}, \bar{O})$. *Controller* K *renders* (S, I, O) *to be* $(\varphi, \beta, \lambda) - pIODS$ *if and only if controller* $\bar{K}$ *renders* $(\bar{S}, \bar{I}, \bar{O})$ *to be* $(\varphi, \beta, \lambda) - pIODS$.

Proof. Necessity: Since there exists an EIOSR between (S, I, O) and $(\bar{S}, \bar{I}, \bar{O})$, we can obtain controller $\bar{K}$ corresponding to controller K via Proposition 3.13. Due to controller K rendering (S, I, O) to be $(\varphi, \beta, \lambda) - pIODS$, we have

$$O((u^c(t), u^d(t)), x(t)) \leq \max_{\hat{t}\in[0,t]} \beta(\varphi(I((u^c(\hat{t}), u^d(\hat{t})), x(\hat{t}))), t - \hat{t}) + \lambda, \forall t \in \mathbb{N}.$$

Via an EIOSR, we get

$$\bar{O}((\bar{u}^c(t), \bar{u}^d(t)), \bar{x}(t)) \leq \max_{\hat{t}\in[0,t]} \beta(\varphi(\bar{I}((\bar{u}^c(\hat{t}), \bar{u}^d(\hat{t})), \bar{x}(\hat{t}))), t - \hat{t}) + \lambda, \forall t \in \mathbb{N},$$

where $((u^c(t), u^d(t)), (\bar{u}^c(t), \bar{u}^d(t)), x(t), \bar{x}(t)), ((u^c(\hat{t}), u^d(\hat{t})), (\bar{u}^c(\hat{t}), \bar{u}^d(\hat{t})), x(\hat{t}), \bar{x}(\hat{t})) \in R$.

Therefore, there exists controller $\bar{K}$ rendering $(\bar{S}, \bar{I}, \bar{O})$ to be $(\varphi, \beta, \lambda) - pIODS$.

Sufficiency: Defining a large enough positive integer M, it satisfies $S_M(\bar{\Omega}) = \Delta_{k_3}$ and

$$\max_{\hat{t}\in[0;t]} \beta(\varphi(\bar{I}((\bar{u}^c(\hat{t}), \bar{u}^d(\hat{t})), \bar{x}(\hat{t}))), M - \hat{t}) \to 0. \tag{3.10}$$

Case I: If $t < M + \eta$, we define

$$\bar{O}_{max} = \max\{\bar{O}((\bar{u}^c(0), \bar{u}^d(0)), \bar{x}(0)), \ldots, \bar{O}(\bar{u}^c(t), \bar{u}^d(t)), \bar{x}(t))\},$$

$$\bar{I}_{max} = \max\{\bar{I}((\bar{u}^c(0), \bar{u}^d(0)), \bar{x}(0)), \ldots, \bar{I}(\bar{u}^c(t), \bar{u}^d(t)), \bar{x}(t))\}.$$

Since controller $\bar{K}$ renders $(\bar{S}, \bar{I}, \bar{O})$ to be $(\varphi, \beta, \lambda) - pIODS$, we have

$$\bar{O}_{max} \leq \max_{\hat{t}\in[0;t]} \beta(\varphi(\bar{I}_{max}), t - \hat{t}) + \lambda.$$

Via an EIOSR, we get

$$O_{max} = \max\{O((u^c(0), u^d(0)), x(0)), \ldots, O(u^c(t), u^d(t)), x(t))\},$$

$$I_{max} = \max\{I((u^c(0), u^d(0)), x(0)), \ldots, I(u^c(t), u^d(t)), x(t))\},$$

and controller K of (S, I, O). Under the influence of K and the EIOSR, we have

$$O_{max} \leq \max_{\hat{t}\in[0;t]} \beta(\varphi(I_{max}), t - \hat{t}) + \lambda.$$

Case II: If $t \geq M + \eta$, we get $\max_{\hat{t}\in[0;t]} \beta(\varphi(\bar{I}((\bar{u}^c(\hat{t}), \bar{u}^d(\hat{t})), \bar{x}(\hat{t}))), M - \hat{t}) \to 0$. This can be interpreted that $\bar{I}$ has no influence on output cost function $\bar{O}$. Via Assumption 3.4, it can be seen that logical network $(\bar{S}, \bar{I}, \bar{O})$ is in a zero-disturbance state $\bar{u}_0^d$. Via Theorem 3.11, logical network $(\bar{S}, \bar{I}, \bar{O})$ has entered into $\bar{\Omega}$ under the influence of controller $\bar{K}$. Through Proposition 3.6, we see that

$$\bar{O}((\bar{u}^c(0), \bar{u}^d(0)), \bar{x}(0)) \leq \lambda \leq 0 + \lambda.$$

We can get $O \leq \lambda \leq 0 + \lambda$ by the EIOSR and controller K by Proposition 3.12.

In summary, under the influence of controller K, (S, I, O) satisfies

$$O((u^c(t), u^d(t)), x(t)) \leq \max_{\hat{t}\in[0,t]} \beta(\varphi(I((u^c(\hat{t}), u^d(\hat{t})), x(\hat{t}))), t - \hat{t}) + \lambda, \forall t \in \mathbb{N}.$$

That is, controller K renders (S, I, O) to be $(\varphi, \beta, \lambda) - pIODS$.
The conclusion follows. □

3.3. Controller design for infinite systems with inputs

Subject to the method, which is proposed to analyze the robustness of infinite systems in [15], we also analyze the given infinite systems with inputs via the corresponding finite systems. Furthermore, in the part of controller design, it is an obvious idea that whether the controller of the corresponding finite systems is still available for the given infinite systems.

This subsection will prove the above ideas and present the precondition for this hypothesis. First, we present a relation between infinite systems with cost functions and finite systems with cost functions based on Propositions 3.12 and 3.13. The relation will help us to investigate the controller design.

Definition 3.15. *Define (S, I, O) and $(\hat{S}, \hat{I}, \hat{O})$ as an infinite system with inputs and a finite system with inputs, respectively. Let $R(\varepsilon) \subset (U^c, U^d) \times (\hat{U}^c, \hat{U}^d) \times X \times \hat{X}$ be parameterized by $\varepsilon \in [0, \infty)$. $R(\varepsilon)$ is said to be an approximate input-output simulation relation ($AIOSR$) between (S, I, O) and $(\hat{S}, \hat{I}, \hat{O})$, if $R(\varepsilon) \subset R(\varepsilon')$ holds for all $\varepsilon \leq \varepsilon'$ and for every $\varepsilon \in [0, \infty)$, we have*

(i) $\forall x \in X, \exists \hat{x} \in \hat{X} : (x, \hat{x}) \in R_X(\varepsilon)$ *and* $\forall (u^c, u^d) \in (U^c(x), u^d), \exists (\hat{u}^c, \hat{u}^d) \in (\hat{U}^c(\hat{x}), \hat{u}^d)$*:*

- *(a)* $(x, \hat{x}, (u^c, u^d), (\hat{u}^c, \hat{u}^d)) \in R(\varepsilon)$,
- *(b)* $\forall x' \in f((u^c, u^d), x), \exists \hat{x}' \in \hat{f}((\hat{u}^c, \hat{u}^d), \hat{x}) : (x', \hat{x}') \in R_X(\varepsilon)$;

(ii) $\forall \hat{x} \in \hat{X}, \exists x \in X : (x, \hat{x}) \in R_X(\varepsilon)$ *and* $\forall (\hat{u}^c, \hat{u}^d) \in (\hat{U}^c(\hat{x}), \hat{u}^d), \exists (u^c, u^d) \in (U^c(x), u^d)$*:*

- *(c)* $(x, \hat{x}, (u^c, u^d), (\hat{u}^c, \hat{u}^d)) \in R(\varepsilon)$,
- *(d)* $\forall \hat{x}' \in \hat{f}((\hat{u}^c, \hat{u}^d), \hat{x}), \exists x' \in f((u^c, u^d), x) : (x', \hat{x}') \in R_X(\varepsilon)$;

(iii) $I((u^c, u^d), x) = \hat{I}((\hat{u}^c, \hat{u}^d), \hat{x})$;

(iv) $O((u^c, u^d), x) = \hat{O}((\hat{u}^c, \hat{u}^d), \hat{x})$.

Following this relation, we propose an algorithm to obtain a corresponding finite system from the given infinite system.

We are given an infinite system with inputs $S = (X_0, (U^c, U^d), X, f)$, where $X_0 \subseteq X = [a_1, b_1] \cup \cdots \cup [a_n, b_n]$, $U^c = [c_1, d_1] \cup \cdots \cup [c_m, d_m]$, $U^d = [e_1, f_1] \cup \cdots \cup [e_p, f_p]$.

Algorithm 3.16. *(Approximate input-output simulation algorithm)*

Step 1 *Define* n_1 *as the largest positive integer satisfying* $a_1 + 2n_1\varepsilon \leq b_1$*;*

Step 2 *Similar to Step 1, we can obtain* $n_2, \ldots, n_n, m_1, \ldots, m_m, p_1, \cdots, p_p$*;*

Step 3 *Define*

$$\begin{aligned}\hat{X}_0 \subseteq \hat{X} &= \{a_1, \ldots, a_1 + 2n_1\varepsilon, a_2, \ldots, a_2 + 2n_2\varepsilon, \ldots, a_n, \ldots, a_n + 2n_n\varepsilon\},\\ \hat{U}^c &= \{c_1, \ldots, c_1 + 2m_1\varepsilon, c_2, \ldots, c_2 + 2m_2\varepsilon, \ldots, c_m, \ldots, c_m + 2m_m\varepsilon\},\\ \hat{U}^d &= \{e_1, \ldots, e_1 + 2p_1\varepsilon, e_2, \ldots, e_2 + 2p_2\varepsilon, \ldots, e_p, \ldots, e_p + 2p_p\varepsilon\};\end{aligned}$$

Step 4 *For* $\forall(\hat{u}^c, \hat{u}^d) \in (\hat{U}^c, \hat{U}^d), \forall \hat{x} \in \hat{X}$*, we look for* $[a_i + (2n_i - 1)\varepsilon, a_i + (2n_i + 1)\varepsilon)$ *to satisfy* $f((\hat{u}^c, \hat{u}^d), \hat{x}) \in [a_i + (2n_i - 1)\varepsilon, a_i + (2n_i + 1)\varepsilon)$*, and then we define* $\hat{x}' = a_i + 2n_i\varepsilon$*. Define* $\hat{f} : (\hat{U}^c, \hat{U}^d) \times \hat{X} \rightrightarrows \hat{X}$*;*

Step 5 *Define* $\hat{I}((\hat{u}^c, \hat{u}^d), \hat{x}) = I((u^c, u^d), x)$*, where* $((u^c, u^d), (\hat{u}^c, \hat{u}^d), x, \hat{x}) \in R(\varepsilon)$*;*

Step 6 *Define* $\hat{O}((\hat{u}^c, \hat{u}^d), \hat{x}) = O((u^c, u^d), x)$*, where* $((u^c, u^d), (\hat{u}^c, \hat{u}^d), x, \hat{x}) \in R(\varepsilon)$*.*

Remark 3.17. *The computational complexity of Algorithm 3.16 is* $O(k_1k_2k_3)$*.* Next, to make this part readable, the following example will illustrate our idea.

Example 3.18. *We continue with the physical part with inputs of the CPS with inputs in Example 3.2.*

We set the approximate simulation relation $R(\varepsilon)$ *between the physical part with inputs* S_p *and the corresponding finite system with inputs* $\hat{S}_p$ *parameterized by* $\varepsilon = 0.5$*. Through Algorithm 3.16, we can get finite system with inputs*

$$\hat{S}_p = (\hat{X}_p^0, (\hat{U}_p^c, \hat{U}_p^d), \hat{X}_p, \hat{f}),$$

where

$$\hat{X}_p^0 = \{-2, -1, 0, 1, 2\},$$

$$\hat{X}_p = \{-5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5\},$$

$$(\hat{U}_p^c, \hat{U}_p^d) = (\{-1, 0, 1, 2, 3\}, \{0, 1, 2\}),$$

and $\hat{f}$ *is as shown in the following tables:*

Table 1. $\hat{f}$ when $x_c = a_1, \hat{u}^d = 0$.

$(\hat{u}_p^c, \hat{x}_p)$	-5	-4	-3	-2	-1	0	1	2	3	4	5
-1	-3	-3	-2	-2	-1	-1	-1	0	0	1	1
0	-2	-2	-1	-1	0	0	0	1	1	2	2
1	-1	-1	0	0	1	1	1	2	2	3	3
2	0	0	1	1	2	2	2	3	3	4	4
3	1	1	2	2	3	3	3	4	4	5	5

Table 2. $\hat{f}$ when $x_c = a_2, \hat{u}^d = 0$.

$(\hat{u}^c_p, \hat{x}_p)$	-5	-4	-3	-2	-1	0	1	2	3	4	5
-1	-2	-2	-2	-2	-1	-1	-1	0	0	0	0
0	-1	-1	-1	-1	0	0	0	1	1	1	1
1	0	0	0	0	1	1	1	2	2	2	2
2	1	1	1	1	2	2	2	3	3	3	3
3	2	2	2	2	3	3	3	4	4	4	4

Since the value of the input cost function is zero when t is large enough, which is converted to the zero-disturbance state $\hat{u}^d_p = 0$, we just give transition $\hat{f}$ when the finite system with inputs corresponding to the physical part with inputs is in zero-disturbance state $\hat{u}^d_p = 0$. It is convenient for us to design the controller in Section 4.

Next, we will investigate whether the controller of the corresponding finite system with inputs is still available for the given finite system with inputs.

Before considering the controller, objective state set Ω of the infinite system with inputs is first given, which may be one interval or something else, and the objective state set $\hat{\Omega}$ of the finite system with inputs can be obtained via Algorithm 3.16.

In the following, we will demonstrate the main work which verifies the hypothesis proposed in the start of this subsection.

Theorem 3.19. *Assume there exists an AIOSR between (S, I, O) and $(\hat{S}, \hat{I}, \hat{O})$. There exists a controller $u^c = Kx$ rendering (S, I, O) to be pIODS, if and only if there exists a controller $\hat{u}^c = \hat{K}\hat{x}$ rendering $(\hat{S}, \hat{I}, \hat{O})$ to be pIODS.*

Proof. Necessity: Since there exists a controller $u^c = Kx$ rendering the infinite system with cost functions (S, I, O) to be $pIODS$, we have

$$O((u^c(t), u^d(t)), x(t)) \le \max_{t' \in [0;t]} \beta(\kappa(I((u^c(t'), u^d(t')), x(t'))), t - t') + \lambda$$

established and there exists T such that the infinite system with inputs enters into an objective state set Ω when $t \ge T$.

We are given an arbitrary state-input sequence of the infinite system with inputs as follows:

state: $x(0), x(1), \cdots, x(T), x(T+1), \cdots;$

input: $(u^c(0), u^d(0)), (u^c(1), u^d(1)), \cdots, (u^c(T), u^d(T)), (u^c(T+1), u^d(T+1)), \cdots,$

where $u^c(0), u^c(1), \cdots, u^c(T), u^c(T+1), \cdots$ is given by $u^c = Kx$.

On account of an $AIOSR$ between the infinite system with cost functions (S, I, O) and the finite system with cost functions $(\hat{S}, \hat{I}, \hat{O})$, we can obtain a state-input sequence of the controlled finite system. Via Definition 3.15, for $x(0) \in X$, we can find $\hat{x}(0) \in \hat{X}$ to satisfy $(x(0), \hat{x}(0)) \in R_X(\varepsilon)$. For $(u^c(0), u^d(0)) \in (U^c, U^d)$, we can also find $(\hat{u}^c(0), \hat{u}^d(0)) \in (\hat{U}^c, \hat{U}^d)$ to satisfy $(x(0), \hat{x}(0), (u^c(0), u^d(0)), (\hat{u}^c(0), \hat{u}^d(0))) \in R(\varepsilon)$. For $x(1) \in f((u^c(0), u^d(0)), x(0))$, there exists $\hat{x}(1) \in \hat{f}((\hat{u}^c(0), \hat{u}^d(0)), \hat{x}(0))$ such that $(x(1), \hat{x}(1)) \in R_X(\varepsilon)$. Repeating this process, we can get the state-input sequence of the finite system with inputs as follows:

state: $\hat{x}(0), \hat{x}(1), \cdots, \hat{x}(T), \hat{x}(T+1), \cdots;$

input: $(\hat{u}^c(0), \hat{u}^d(0)), (\hat{u}^c(1), \hat{u}^d(1)), \cdots, (\hat{u}^c(T), \hat{u}^d(T)), (\hat{u}^c(T+1), \hat{u}^d(T+1)), \cdots.$

From the mapping relation between the state-input sequence of the finite system with inputs, we can get its controller $\hat{u}^c = \hat{K}\hat{x}$. For the infinite system with inputs, we have $x(T), x(T+1), \cdots \in \Omega$. Since $(x(T), \hat{x}(T)), (x(T+1), \hat{x}(T+1)), \cdots \in R(\varepsilon)$, we can get $\hat{x}(T), \hat{x}(T+1), \cdots \in \hat{\Omega}$.

It can be seen from Definition 3.15 that

$$\begin{aligned} \hat{O}((\hat{u}^c(t), \hat{u}^d(t)), \hat{x}(t)) &= O((u^c(t), u^d(t)), x(t)) \\ &\leq \max_{t' \in [0;t]} \beta(\kappa(I((u^c(t'), u^d(t')), x(t'))), t - t') + \lambda \\ &= \max_{t' \in [0;t]} \beta(\kappa(\hat{I}((\hat{u}^c(t'), \hat{u}^d(t')), \hat{x}(t'))), t - t') + \lambda. \end{aligned}$$

So there exists a controller $\hat{u}^c = \hat{K}\hat{x}$ rendering the finite system with cost functions $(\hat{S}, \hat{I}, \hat{O})$ to be *pIODS*.

Sufficiency: Since there exists a controller $\hat{u}^c = \hat{K}\hat{x}$ rendering the finite system with cost functions $(\hat{S}, \hat{I}, \hat{O})$ to be *pIODS*, we can give an arbitrary state-input sequence of the finite system with inputs as follows:

state: $\hat{x}(0), \hat{x}(1), \cdots, \hat{x}(T), \hat{x}(T+1), \cdots;$

input: $(\hat{u}^c(0), \hat{u}^d(0)), (\hat{u}^c(1), \hat{u}^d(1)), \cdots, (\hat{u}^c(T), \hat{u}^d(T)), (\hat{u}^c(T+1), \hat{u}^d(T+1)), \cdots.$

We consider that T is large enough that $\hat{I}((\hat{u}^c(t), \hat{u}^d(t)), \hat{x}(t)) = 0$ when $t \geq T$. That is equal to $\hat{u}^d(t) = 0$ when $t \geq T$. We have that $\hat{x}(t) \in \hat{\Omega}$ when $t \geq T$. Then the above input sequence is equal to the following:

input: $(\hat{u}^c(0), \hat{u}^d(0)), (\hat{u}^c(1), \hat{u}^d(1)), \cdots, (\hat{u}^c(T), 0), (\hat{u}^c(T+1), 0), \cdots.$

For $\forall x(0)$ of the infinite system with inputs satisfying $(x(0), \hat{x}(0)) \in R_X(\varepsilon)$, let $x(0)$ run with the above input sequence, that is, it runs with controller $\hat{u}^c = \hat{K}\hat{x}$. Owing to Definition 3.15, we can get

$$(x(0), \hat{x}(0), (\hat{u}^c(0), \hat{u}^d(0)), (\hat{u}^c(0), \hat{u}^d(0))) \in R(\varepsilon), x(1) \in f((\hat{u}^c(0), \hat{u}^d(0)), x(0)),$$

$$\hat{x}(1) \in \hat{f}((\hat{u}^c(0), \hat{u}^d(0)), \hat{x}(0)), \textit{ and } (x(1), \hat{x}(1)) \in R_X(\varepsilon).$$

In turn, we can obtain the state-input sequence of the infinite system with inputs as follows:

state: $x(0), x(1), \cdots, x(T), x(T+1), \cdots;$

input: $(\hat{u}^c(0), \hat{u}^d(0)), (\hat{u}^c(1), \hat{u}^d(1)), \cdots, (\hat{u}^c(T), 0), (\hat{u}^c(T+1), 0), \cdots.$

From the mapping relation between the above state-input sequence of the infinite system with inputs, we can get its controller $u^c = Kx$. Due to $\hat{x}(t) \in \hat{\Omega}, t \in \{T, T+1, \cdots\}$ and $(x(t), \hat{x}(t)) \in R_X(\varepsilon), t \in \{0, 1, \cdots\}$, we have $x(t) \in \Omega, t \in \{T, T+1, \cdots\}$.

Via the relation of input and output cost functions between these two systems in Definition 3.15, we can get the input-output dynamical stable equation of the infinite system with inputs as follows:

$$O((u^c(t), u^d(t)), x(t)) = \hat{O}((\hat{u}^c(t), \hat{u}^d(t)), \hat{x}(t))$$

$$\leq \max_{t'\in[0;t]} \beta(\kappa(\hat{I}((\hat{u}^c(t'), \hat{u}^d(t')), \hat{x}(t'))), t-t') + \lambda$$
$$= \max_{t'\in[0;t]} \beta(\kappa(I((u^c(t'), u^d(t')), x(t'))), t-t') + \lambda.$$

So there exists a controller $u^c = Kx$ rendering the finite system with cost functions (S, I, O) to be $pIODS$.

The conclusion follows. □

4. Illustrative example

This is a continuation of Examples 3.2 and 3.18.

We are given the objective state set $\Omega = \Omega_c \cup \Omega_p = \{a_2\} \cup (0.5, 1.5]$ of the CPS with inputs $S = (S_c, S_p)$ in Example 3.2. We assume its zero-disturbance state is $u^d_{c0} = 0, u^d_{p0} = 0$.

Define input and output cost functions I and O of the CPS with inputs $S = (S_c, S_p)$ as follows:

$$I((u^c_c, u^d_c), (u^c_p, u^d_p), x_c, x_p) = I^d_c(u^d_c) + |u^d_p|,$$
$$O((u^c_c, u^d_c), (u^c_p, u^d_p), x_c, x_p) = \begin{cases} \rho, & ((u^c_c, u^d_c), (u^c_p, u^d_p), x_c, x_p) \in \Lambda, \\ arbitrary\ \ value, & else, \end{cases}$$

where $\Lambda = ((U^c_c, 0), (U^c_p, 0), \Omega_c, \Omega_p), I^d_c(0) = 0, I^d_c(1) = 100.$

Then, according to the following steps, the controller of the given CPS with inputs $S = (S_c, S_p)$ is designed.

Step 1 : Applying Algorithm 3.16, the cyber part with inputs S_c is converted to the corresponding logical network with inputs $\bar{S}_c$ when it is in zero-disturbance state $u^d_{c0} = 0$. Define $\bar{X}_c := \Delta_2, a_1 \sim \delta^1_2, a_2 \sim \delta^2_2, \bar{U}^c_c := \Delta_2, \top \sim \delta^1_2, \perp \sim \delta^2_2, \bar{U}^d_c := \Delta_2, 1 \sim \delta^1_2, 0 \sim \delta^2_2$. Then $\bar{S}_c$ is constructed as follow:

$$\bar{x}_c(t+1) = \bar{L}_c \bar{u}^c_c(t) \bar{x}_c(t),$$

where

$$\bar{L}_c = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}.$$

Step 2 : It can be seen from the proof of Theorem 3.11 that controller $\bar{u}^c_c(t) = \bar{K}_c \bar{x}_c(t)$ of the logical network with inputs $\bar{S}_c$ is obtained, which can render $\bar{S}_c$ to eventually transform into $\bar{\Omega}_c$.

$$\bar{K}_c = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

Step 3 : Via Proposition 3.12, controller K_c of the cyber part with inputs $\bar{S}_c$ is achieved.

Step 4 : Based on Example 3.18, the finite system with inputs $\hat{S}_p = (\hat{X}^0_p, (\hat{U}^c_p, \hat{U}^d_p), \hat{X}_p, \hat{f}_p)$ corresponding to the physical physical part of the cyber-physical with inputs S_p is given.

Via an AIOSR, we know that

$$\begin{aligned} \hat{x}'_p \in \hat{f}^1_p((\hat{u}^c_p, \hat{u}^d_p), x_p) &\Longleftrightarrow |\hat{x}'_p - 0.4\hat{x}_p - \hat{u}^c_p - \hat{u}^d_p| \leq 0.5, \\ \hat{x}'_p \in \hat{f}^2_p((\hat{u}^c_p, \hat{u}^d_p), x_p) &\Longleftrightarrow |\hat{x}'_p - 0.3\hat{x}_p - \hat{u}^c_p - \hat{u}^d_p| \leq 0.5. \end{aligned} \tag{4.1}$$

Equation group (4.1) gives the relation between successor $\hat{x}_p'$ of the finite system with inputs and successor x_p' of the infinite system with inputs.

Step 5 : Similar to **Step1**, we can convert the finite system with inputs $\hat{S}_p$ to the corresponding logical network with inputs $\bar{S}_p$ when it is in a zero-disturbance state. Define $\bar{X}_p := \Delta_{11}$, $\bar{U}_p^c := \Delta_5$. Then, let $\bar{R}_{X_p} = \{(-5+2\times 0.5(i-1), \delta_{11}^i)|i = 1, 2, \cdots, 11\}$ and $\bar{R}_{U_p^c} = \{(-1+2\times 0.5(i-1), \delta_5^i)|i = 1, 2, \cdots, 5\}$.

Then, logical network with inputs $\bar{S}_p$ is as follow:

When $\bar{x}_c(t) = \delta_2^1$,

$$\bar{x}_p(t+1) = \bar{L}_p^1 \bar{u}_p^c(t)\bar{x}_p(t),$$

where

$$\begin{aligned}\bar{L}_p^1 = \delta_{11}[&3\ 3\ 4\ 4\ 5\ 5\ 5\ 6\ \ 6\ \ 7\ \ 7\\ &4\ 4\ 5\ 5\ 6\ 6\ 6\ 7\ \ 7\ \ 8\ \ 8\\ &5\ 5\ 6\ 6\ 7\ 7\ 7\ 8\ \ 8\ \ 9\ \ 9\\ &6\ 6\ 7\ 7\ 8\ 8\ 8\ 9\ \ 9\ \ 10\ \ 10\\ &7\ 7\ 8\ 8\ 9\ 9\ 9\ 10\ \ 10\ \ 11\ \ 11].\end{aligned}$$

When $\bar{x}_c(t) = \delta_2^2$,

$$\bar{x}_p(t+1) = \bar{L}_p^2 \bar{u}_p^c(t)\bar{x}_p(t),$$

where

$$\begin{aligned}\bar{L}_p^2 = \delta_{11}[&4\ 4\ 4\ 4\ 5\ 5\ 5\ 6\ \ 6\ \ 6\ \ 6\\ &5\ 5\ 5\ 5\ 6\ 6\ 6\ 7\ \ 7\ \ 7\ \ 7\\ &6\ 6\ 6\ 6\ 7\ 7\ 7\ 8\ \ 8\ \ 8\ \ 8\\ &7\ 7\ 7\ 7\ 8\ 8\ 8\ 9\ \ 9\ \ 9\ \ 9\\ &8\ 8\ 8\ 8\ 9\ 9\ 9\ 10\ \ 10\ \ 10\ \ 10].\end{aligned}$$

Step 6 : Based on the above logical network with inputs, the controller is designed as follow:

When $\bar{x}_c(t) = \delta_2^1$, the controller is $\bar{u}_{p1}^c(t) = \bar{K}_{p1}\bar{x}_p(t)$, where

$$\bar{K}_{p1} = \delta_5[4\ 4\ 3\ 3\ 2\ 2\ 2\ 1\ 1\ 1\ 1].$$

When $\bar{x}_c(t) = \delta_2^2$, the controller is $\bar{u}_{p2}^c(t) = \bar{K}_{p2}\bar{x}_p(t)$, where

$$\bar{K}_{p2} = \delta_5[3\ 3\ 3\ 3\ 2\ 2\ 2\ 1\ 1\ 1\ 1].$$

Step 7 : Via Proposition 3.12, we can get controller K_c of the cyber part of the CPS with inputs corresponding to $\bar{K}_c$ and controllers $\hat{K}_{p1}$ and $\hat{K}_{p2}$ of the physical part of CPS with inputs corresponding $\bar{K}_{p1}$ and $\bar{K}_{p2}$. Via Theorem 3.19, controllers K_{p1} and K_{p2} are obtained.

5. Conclusions

This paper has studied the robustness of CPSs with inputs under controller design and presented a number of new results. First, the models of CPSs with inputs and their definitions of the robustness are presented. Second, two algorithms have converted the given infinite systems and finite systems into logical networks with inputs. Third, the robustness under the influence of the controller between them has been analyzed. Eventually, an illustrative example has been given to show the process of controller design rendering CPSs with inputs robust.

Actually, in the future work, we can consider how to reduce the high computational complexity for large-scale CPS. However, it should be pointed out that there are still some trick problems in reducing the high computational complexity for state-flipped control. This will be investigated in our future work. This manuscript focused on the modeling and controller design for CPSs. Additional, many scholars consider the problem of information security issues of CPSs. See more details in [20, 21]. This will also be considered in our future work.

Author contributions

Guodong Zhao: Conceptualization, Methodology, Validation, Formal analysis, Investigation, Writing-original draft preparation, Writing-review and editing, Visualization, Haitao Li: Conceptualization, Software, Validation, Formal analysis, Resources, Data curation, Supervision, Project administration, Funding acquisition. All authors have read and agreed to the published version of the manuscript.

Use of Generative-AI tools declaration

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

The work was supported by the National Natural Science Foundation of China under grants 62273216 and 62473239, and the Major Basic Research Project of Natural Science Foundation of Shandong Province under grant ZR2024ZD41.

Conflict of interest

Professor Haitao Li is an editorial board member for AIMS Mathematics and was not involved in the editorial review and/or the decision to publish this article.

The authors declare that The authors have no financial or personal relationships with other people or organizations that could have inappropriately influenced our work. There are no professional or other personal interests of any nature in any product, service, and/or company that could be construed as influencing the opinions presented in, or the review of, the manuscript entitled "Controller design for cyber-physical systems with inputs via logical networks with inputs".

References

1. X. Gao, M. Peng, C. Tse, Cascading failure analysis of cyber–physical power systems considering routing strategy, *IEEE Trans. Circuits Syst. II Express Briefs*, **70** (2023), 136–140. https://doi.org/10.1109/TCSII.2021.3071920
2. M. Cintuglu, O. Mohammed, K. Akkaya, A. Uluagac, A survey on smart grid cyber-physical system testbeds, *IEEE Commun. Surv. Tutor.*, **19** (2017), 446–464. https://doi.org/10.1109/COMST.2016.2627399
3. M. Xing, Y. Wang, Q. Pang, G. Zhuang, Dynamic-memory event-based asynchronous attack detection filtering for a class of nonlinear cyber-physical systems, *IEEE Trans. Cybern.*, **53** (2023), 653–667. https://doi.org/10.1109/TCYB.2021.3127231
4. Y. Feng, B. Hu, H. Hao, Y. Gao, Z. Li, J. Tan, Design of distributed cyber–physical systems for connected and automated vehicles with implementing methodologies, *IEEE Trans. Ind. Informat.*, **14** (2018), 4200–4211. https://doi.org/10.1109/TII.2018.2805910
5. T. Nguyen, S. Zeadally, A. Vuduthala, Cyber-physical cloud manufacturing systems with digital twins, *IEEE Internet Comput.*, **26** (2022), 15–21. https://doi.org/10.1109/MIC.2021.3058921
6. A. Tyagi, S. Aswathy, G. Aghila, N. Sreenath, AARIN: Affordable, accurate, reliable and innovative mechanism to protect a medical cyber-physical system using blockchain technology, *Int. J. Intell. Netw.*, **2** (2021), 175–183. https://doi.org/10.1016/j.ijin.2021.09.007
7. Y. Tsang, T. Yang, Z. Chen, C. Wu, K. Tan, How is extended reality bridging human and cyber-physical systems in the IoT-empowered logistics and supply chain management? *Int. Things*, **20** (2022), 100623. https://doi.org/10.1016/j.iot.2022.100623
8. C. Lee, C. Ng, C. Chung, K. Keung, Cloud-based cyber-physical logistics system with nested MAX-MIN ant algorithm for e-commerce logistics, *Expert Syst. Appl.*, **211** (2023), 118643. https://doi.org/10.1016/j.eswa.2022.118643
9. E. Sontag, *Input-to-State Stability*, London: Springer, 2015.
10. L. Grün, Input-to-state dynamical stability and its Lyapunov function characterization, *IEEE Trans. Autom. Control*, **47** (2002), 1499–1504. https://doi.org/10.1109/TAC.2002.802761
11. R. Matthias, T. Paulo, A notion of robustness for cyber-physical systems, *IEEE Trans. Autom. Control*, **61** (2016), 2108–2123. https://doi.org/10.1109/TAC.2015.2492438
12. D. Cheng, Y. Dong, Semi-tensor product of matrices and its some applications to physics, *Methods Appl. Anal.*, **10** (2003), 565–588.
13. H. Li, Y. Li, W. Li, Y. Liu, Stability analysis of block logical dynamical systems and its application in logical networks with time delay, *IEEE Trans. Autom. Control*, **69** (2024), 7211–7215. https://doi.org/10.1109/TAC.2024.3394549
14. H. Li, Z. Liu, W. Li, Nonsingularity of Grain-like cascade feedback shift registers subject to fault attacks, *Sci. China Inf. Sci.*, **67** (2024), 192203. https://doi.org/10.1007/s11432-023-4044-8
15. G. Zhao, H. Li, T. Hou, Input–output dynamical stability analysis for cyber-physical systems via logical networks, *IET Control Theory Appl.*, **14** (2020), 2566–2572. https://doi.org/10.1049/iet-cta.2020.0197

16. E. D. Sontag, *Input-to-state stability*, London: Springer, 2015.

17. F. Huang, X. Chang, Tracking control of interval Type-2 fuzzy systems with improved adaptive event-triggered mechanisms and deception attacks under privacy protection, *IEEE Trans. Cybern.*, **46** (2025), 1695–1707. https://doi.org/10.1002/oca.3285

18. S. Wang, X. Chang, Quantized fuzzy H_∞ control for networked systems with a markovian sensor distortion under hybrid attacks, *Int. J. Comput. Math.*, **102** (2025), 1465–1484. https://doi.org/10.1080/00207160.2025.2505024

19. R. Li, M. Yang, T. Chu, State feedback stabilization for Boolean control networks, *IEEE Trans. Autom. Control*, **58** (2013), 1853–1857. https://doi.org/10.1109/TAC.2013.2238092

20. J. Wang, D. Wang, H. Yan, H. Shen, Composite antidisturbance H_∞ control for hidden Markov jump systems with multi-sensor against replay attacks, *IEEE Trans. Autom. Control*, **69** (2024), 1760–1766. https://doi.org/10.1109/TAC.2023.3326861

21. H. Shen, Y. Wang, J. Wu, J. H. Park, J. Wang, Secure control for Markov jump cyber–physical systems subject to malicious attacks: A resilient hybrid learning scheme, *IEEE Trans. Cybern.*, **54** (2024), 54. https://doi.org/10.1109/TCYB.2024.3448407