Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

# Semi-tensor product approach to controllability, reachability, and stabilizability of extended finite state machines*

Chao DONG[†‡1], Yongyi YAN[1], Huiqin LI[1], Jumei YUE[2]

[1]College of Information Engineering, Henan University of Science and Technology, Luoyang 471000, China

[2]College of Agricultural Equipment Engineering, Henan University of Science and Technology, Luoyang 471000, China

[†]E-mail: dongchao@stu.haust.edu.cn

Received Aug. 26, 2023; Revision accepted Oct. 4, 2023; Crosschecked

**Abstract:** This paper uses the semi-tensor product (STP) of matrices and adopts algebraic methods to study the controllability, reachability, and stabilizability of extended finite state machines (EFSMs). Firstly, we construct the bilinear dynamic system model of the EFSM, laying the foundation for further research. Secondly, combined with this bilinear dynamic system model, we propose theorems for the controllability, reachability, and stabilizability of the bilinear dynamic system model of the EFSM. Finally, we design an algorithm to determine the controllability and stabilizability of the EFSM. The correctness of the main results was verified through examples.

**Key words:** Semi-tensor product; STP approach; STP method; Matrix approach; Algebraic method; Finite-valued systems
https://doi.org/10.1631/FITEE.2300578 **CLC number:**

unedited

## 1 Introduction

In finite state machines (FSMs), the system's controllability, reachability, and stabilizability are of great significance and often reflect the essential characteristics of the system. Controllability relates to whether the state variables in the system can be controlled. Reachability concerns whether the specified state in the system can be reached in a limited number of state changes. Stabilizability relates to the stable state of the entire system. By studying and discussing these properties, we can achieve optimal control strategies for FSM.

As the research on FSMs deepens, many scholars have been exploring the controllability, reachability, and stabilizability of FSMs while studying other problems. This has laid the foundation for further solving research objectives. Farah (2010) used reachability graphs to represent the execution states of protocols and to verify consistency. A new verification technology based on backtracking for complex FSM modeling to detect all deadlock errors is proposed. This provides a new method for detecting FSM system deadlocks. Yao (2017) conducted a stabilizability analysis by combining the method of Multiple Lyapunov Functions with pattern-dependent average dwell time techniques, and derived and proved corresponding sufficient conditions. Based on this, a state feedback controller is designed to achieve stabilizability for hybrid systems. This provides a new approach for studying multiple systems. Similarly, many other scholars have made outstanding contributions to the controllability (Henry et al., 2023; Dridi et al., 2022; Gluck et al., 2023), reachability (Kocher et al., 2021; Pilch et al., 2021; Adak et al., 2021), and stabilizability (Han et al., 2018; Boukerrou et al., 2022; Han et al., 2018; Zhang et al.,

‡ Corresponding author
* Project supported by the National Natural Science Foundation of China (No. U1804150, 62073124)
ORCID: Chao DONG, https://orcid.org/0000-0002-1420-6320;
Yongyi YAN, https://orcid.org/0000-0002-7181-5894;
Huiqin LI, https://orcid.org/0000-0003-3035-4128;
Jumei YUE, https://orcid.org/ 0000-0003-1474-0088
© Zhejiang University Press 2023

2020) of FSMs, further demonstrating their importance. However, we found that the studies mentioned above all used traditional formal language methods to introduce the concepts of controllability, reachability, and stabilizability for normal FSMs. Additionally, these methods do not apply to EFSMs. Therefore, we have identified a new mathematical tool, the STP, and have used algebraic methods to study the controllability, reachability, and stabilizability of EFSMs.

The advantage of the STP is that it automatically finds the hierarchy of data and effectively processes multi-dimensional arrays while generalizing matrix multiplication to two matrices of any dimension and preserving the main properties of ordinary matrix multiplication. This method has wide applications in solving complex mathematical and engineering problems, such as FSMs (Dou et al., 2019; Yan et al., 2023; Yan et al., 2022), information systems (Li et al., 2018; Wang et al., 2019; Wang et al., 2019), Boolean networks (Kong et al., 2023; Li et al., 2023; Sun et al., 2023), engineering electronics (Lv et al., 2021; Wu et al., 2016), fuzzy control systems (Chang et al., 2022; Wang et al., 2019; Wang et al., 2022), and more.

Most of the research results on the FSM using traditional formal language methods have been applied to the normal FSM. Moreover, the results of these studies do not apply to the EFSM. Therefore, we have first proposed introducing the concepts of controllability, reachability, and stabilizability from control theory into the bilinear dynamic system model of the EFSM under the framework of STP. These concepts enable us to study the EFSM like those for studying control theory systems. Firstly, we constructed the bilinear dynamic system model of the EFSM and achieved the purpose of studying the EFSM using algebraic methods. Secondly, we introduced the concepts of controllability, reachability, and stabilizability from control theory into the bilinear dynamic system model of the EFSM based on the STP framework, obtained the concept of controllability, reachability, and stabilizability for the bilinear dynamic system model of the EFSM, and established a foundation for analyzing controllability, reachability, and stabilizability from a mathematical computation perspective. Finally, we designed an algorithm based on mathematical computation to determine the controllability and stabilizability of the bilinear dynamic system model of the EFSM. Compared with traditional methods, our algorithm does not require complex logical symbolic operations and can easily find the needed matrices according to definitions. It only requires algebraic calculations to determine the controllability and stabilizability of the dynamic system model of the EFSM.

The following five sections are organized according to the research roadmap. Section 2 briefly introduces this paper's research object, tools, and related definitions. In Section 3, we construct a bilinear dynamic system model of EFSMs. In Section 4, we introduce some concepts from control theory into the bilinear dynamic system model of EFSMs and design an algorithm. In Section 5, an illustrative example supports the results of this paper, and in Section 6, a brief conclusion is provided.

## 2 Preliminaries

Drawing inspiration from previous research in the field of EFSMs (Goorden et al., 2020; Foster et al., 2018; Lin et al., 2020), we present the main objective of this paper.

**Definition 1** An EFSM is a seven-tuple: $A=(L,D,\Sigma,A_s,L_0,D_0,L_m)$, where $L=(l_1,l_2,\cdots,l_n)$ is a finite set of states; $D=D_1\times\cdots\times D_P$ is a domain of $p$ one-dimensional data variables; $\Sigma$ is a finite set of events; $L_0\subseteq L$ is a set of initial states; . $D_0=D_0^1\times\cdots\times D_0^P$ . is a set of initial data values; $L_m\subseteq L$ is a set of marked states; and $A_s$ is a finite set of transitions, each transition $\alpha\in A_s$ being a five-tuple $\alpha=(o_\alpha,t_\alpha,\sigma_\alpha,g_\alpha,f_\alpha)$, where:

$o_\alpha\in L$ is the origin state of $\alpha$;

$t_\alpha\in L$ is the terminal state of $\alpha$;

$\sigma_\alpha\in\Sigma$ is the transition label;

$g_\alpha\subseteq D$ is the enabling guard of $e$;

$f_\alpha:D\to D$ is the data update function.

If the current state is $o_\alpha$ and $g_\alpha$ is judged to be true, $\alpha=(o_\alpha,t_\alpha,\sigma_\alpha,g_\alpha,f_\alpha)$ will transition. At this point, the state $t_\alpha$ is reachable, and the data update function $f_\alpha$ updates the current value.

unedited

**Definition 2** (Zahn et al., 2022) A state $l_i \in L$ is said to be controllable to $l_j \in L$, if for $L_0 = l_i$ there is an input string $e = e_1 e_2 \cdots e_t \in \Sigma$, so that $l_i(t+1) = l_j$, where $l_i$ and $l_j$ are different states in $L$, $l_i(t+1)$ is the state at time $t+1$, $e$ is called an input sequence, and $l_i \xrightarrow{e_1} \cdots \xrightarrow{e_t} l_j$ is a state transition path.

Then, if the state $l_i \in L$ is controllable to any state $l_j \in L$, the state $l_i \in L$ is said to be controllable. And in the same way, if the state $l_j \in L$ is reachable to any state $l_i \in L$, the state $l_j \in L$ is said to be reachable.

Next, we give the concepts of the controllability, reachability, and stabilizability of nonempty sets of states.

Given two nonempty sets $L_1 \subseteq L$ and $L_2 \subseteq L$, we assume that $L_1 \bigcup L_2 = L$, $L_1 \bigcap L_2 = \varnothing$, where $l_\alpha \in L_1$ and $l_\beta \in L_2$.

Then, the nonempty set of states $L_1 \subseteq L$ is said to be controllable if, for any state $l_\beta \in L_2$, there exists a state $l_\alpha \in L_1$ such that $l_\alpha$ is controllable to $l_\beta$. And in the same way, the nonempty set of states $L_2 \subseteq L$ is said to be reachable if, for any state $l_\alpha \in L_1$, there exists a state $l_\beta \in L_2$ such that $l_\beta$ is reachable from $l_\alpha$.

The nonempty set of states $L_1 \subseteq L$ is said to be one-step returnable if, for any state $l_{\alpha_1} \in L_1$, there exists an input sequence $e \in \Sigma$ and a state $l_{\alpha_2} \in L_1$ such that $l_{\alpha_1} \in L_1$ and $l_{\alpha_2} \in L_1$ are reachable from each other, where $l_{\alpha_1}$ and $l_{\alpha_2}$ represent different states in $L_1$.

The nonempty set of states $L_1 \subseteq L$ is said to be stabilizable if $L_1$ is controllable and one-step returnable.

**Definition 3** (Cheng et al., 2010) For $\boldsymbol{M} \in \boldsymbol{R}^{m\times n}$ and $\boldsymbol{N} \in \boldsymbol{R}^{p\times q}$, their STP is expressed as $\boldsymbol{M} \hat{a} \boldsymbol{N}$, defined as:

$$\boldsymbol{M} \hat{a} \boldsymbol{N} := (\boldsymbol{M} \otimes \boldsymbol{I}_{s/n})(\boldsymbol{N} \otimes \boldsymbol{I}_{s/p}), \tag{1}$$

where $s$ is the least common multiple of $n$ and $p$, $\boldsymbol{R}^{m\times n}$ is the set of $m \times n$ real matrices, and $\otimes$ is the Kronecker product.

**Remark 1** The STP of a matrix is an extension of traditional matrix multiplication. When $n$ is equal to $p$, it becomes a traditional matrix multiplication. In this paper, all matrix products are STPs, and we omit the sign $\ltimes$ unless otherwise specified. Although STP breaks through the dimension limitation of traditional matrix multiplication, it still retains the basic properties of the traditional matrix.

**Definition 4** (Yan et al., 2015) (Swap Matrix) A swap matrix $\boldsymbol{W}_{[m,n]}$, an $mn \times mn$ matrix, is defined as follows. Its rows and columns are labeled by double index $(i, j)$, the rows are arranged by the ordered multi-index $Id[j, i; n, m]$, and the columns are arranged by the ordered multi-index $Id[i, j; m, n]$. Then, the element at the position $[(I, J), (i, j)]$ is

$$\boldsymbol{W}_{((I,J),(i,j))} = \delta_{i,j}^{I.J} = \begin{cases} 1, & I = i \text{ and } J = j \\ 0, & \text{otherwise} \end{cases}. \tag{2}$$

Then, for any $\boldsymbol{X} \in \boldsymbol{R}^m$ and $\boldsymbol{Y} \in \boldsymbol{R}^n$, we have

$$\begin{cases} \boldsymbol{W}_{[m,n]} \text{ 忖 } \boldsymbol{X} & \boldsymbol{Y} = \boldsymbol{Y} \text{ ? } \boldsymbol{X} \\ \boldsymbol{W}_{[n,m]} \text{ 忖 } \boldsymbol{Y} & \boldsymbol{X} = \boldsymbol{X} \text{ ? } \boldsymbol{Y} \end{cases}. \tag{3}$$

**Notations** $\boldsymbol{\delta}_n^l$ is the $l-$th column of identity matrix $\boldsymbol{I}_n$ and $\boldsymbol{\Delta}_n := \{\boldsymbol{\delta}_n^l \,|\, l = 1, 2, \cdots, n\}$. $col_i(\boldsymbol{M})$ is the $i-$th column of matrix $\boldsymbol{M}$, and $col(\boldsymbol{M})$ is the set of all the columns of matrix $\boldsymbol{M}$. $\boldsymbol{M}_{(n,m)}$ stands for the element in the $n-$th row and the $m-$th column of the matrix $\boldsymbol{M}$.

## 3 Bilinear Dynamic System Model of EFSMs

In this section, we will construct a bilinear dynamical system model of the EFSMs and compare it with the normal FSM.

Given an EFSM, $A = (L, D, \Sigma, A_s, L_0, D_0, L_m)$, where $L = \{l_1, l_2, \cdots, l_n\}$, $\Sigma = \{e_1, e_2, \cdots, e_m\}$, and $D = \{d_1, d_2, \cdots, d_p\}$ identify $l_i (1 \le i \le n)$ with $\delta_n^i (1 \le i \le n)$, expressed as $l_i \square \boldsymbol{\delta}_n^i$, for simplicity, $\boldsymbol{\delta}_n^i$ is the vector form of $l_i$. Then, $L$ can be denoted as $\boldsymbol{\Delta}_n$, that is, $L = \{\boldsymbol{\delta}_n^1, \boldsymbol{\delta}_n^2, \cdots, \boldsymbol{\delta}_n^n\}$. Similarly, for $\Sigma$,

unedited

identify $e_j$ with $\boldsymbol{\delta}_m^j (1 \le j \le m)$ (denoted as $e_j \square \boldsymbol{\delta}_m^j$), $\boldsymbol{\delta}_m^j$ is the vector form of $e_j$, and use $\boldsymbol{\Delta}_m$ to denote $\Sigma$, i.e., $\Sigma = \{\boldsymbol{\delta}_m^1, \boldsymbol{\delta}_m^2, \cdots, \boldsymbol{\delta}_m^m\}$. We do the same with $d_z (1 \le z \le p)$, that is, identify $d_z$ with $\boldsymbol{\delta}_p^z$, denoted as $d_z \square \boldsymbol{\delta}_p^z$, so $\boldsymbol{\delta}_p^z$ is the vector form of $d_z$; then $D$ can be denoted as $\Delta_p$, i.e., $D = \{\boldsymbol{\delta}_p^1, \boldsymbol{\delta}_p^2, \cdots, \boldsymbol{\delta}_p^z\}$.

In the EFSM, a transition $\alpha = (o_\alpha, t_\alpha, \sigma_\alpha, g_\alpha, f_\alpha)$ is enabled if the current state is $o_\alpha$ and the guard condition $g_\alpha$ is judged to be true. When the transition $\alpha$ is executed, the state $t_\alpha$ is reached, and the variables are updated by applying $f_\alpha$. Then, the essence of $g_\alpha$ is the judgment of the input sequence. In our dynamic system model, when the judgment is false, the system output is 0; otherwise, the system output is normal. Identify $g_\alpha$ with $\boldsymbol{\delta}_2^k$ ($k=1$ or $k=2$), denoted as $g_a \square \boldsymbol{\delta}_2^k$ ($k=1$ or $k=2$), that is, $\boldsymbol{\delta}_2^k$ ($k=1$ or $k=2$) is the vector form of $g_\alpha$, i.e., $g_\alpha = \left\{\boldsymbol{\delta}_2^1, \boldsymbol{\delta}_2^2\right\}$.

**Definition 5** $\boldsymbol{F} = [\boldsymbol{F}_1 \ldots \boldsymbol{F}_m] \in \boldsymbol{R}^{n \times 2nm}$ is called the state transition structure matrix (STSM) of an EFSM $A = (L, D, \Sigma, A_S, L_0, D_0, L_m)$, which is defined as:

$$\boldsymbol{F}_{j(s,t)} := \begin{cases} \text{if } \boldsymbol{\delta}_2^k = \boldsymbol{\delta}_2^1 \begin{cases} 1 & \text{if } \boldsymbol{\delta}_n^s \in f(\boldsymbol{\delta}_n^t, \boldsymbol{\delta}_m^j) \\ 0 & \text{otherwise} \end{cases} \\ \text{if } \boldsymbol{\delta}_2^k = \boldsymbol{\delta}_2^2 \quad 0 \end{cases}, \tag{4}$$

where $f$ indicates that the current state can be transferred when receiving a new event.

**Remark 2** We obtain the STSM $\boldsymbol{F}$, which can describe all dynamic behaviors of an EFSM. Furthermore, we can utilize this property to compute the new state of the original state after receiving an event.

**Definition 6** $\boldsymbol{H} = [\boldsymbol{H}_1 \ldots \boldsymbol{H}_m] \in \boldsymbol{R}^{p \times 2pm}$ is called the state output structure matrix (SOSM) of an EFSM $A = (L, D, \Sigma, A_S, L_0, D_0, L_m)$, which is defined as:

$$\boldsymbol{H}_{i(s,t)} := \begin{cases} \text{if } \boldsymbol{\delta}_2^k = \boldsymbol{\delta}_2^1 \begin{cases} 1 & \text{if } \boldsymbol{\delta}_p^s \in f_\alpha(\boldsymbol{\delta}_p^t, \boldsymbol{\delta}_m^j) \\ 0 & \text{otherwise} \end{cases} \\ \text{if } \boldsymbol{\delta}_2^k = \boldsymbol{\delta}_2^2 \quad 0 \end{cases}. \tag{5}$$

The following is the STSM and SOSM of normal FSMs (Yan et al., 2015), compared with the STSM and SOSM of EFSMs.

A normal FSM is a seven-tuple $A = (X, E, Y, f_1, g_1, x^0, X^m)$, in which $X$, $E$, and $Y$ are finite sets of states, input symbols, and outputs, respectively, $x^0 \in X$ is the initial state, $X^m \in X$ is the set of accepted states, and $f_1$ and $g_1$ are transition and output functions, which are defined as

$$f_1 : X \times E \to X,$$
$$g_1 : X \times E \to Y.$$

For an FSM $A = (X, E, Y, f, g, x^0, X^m)$, where $X = \{x_1, x_2, \cdots, x_z\}$, $E = \{e_1, e_2, \cdots, e_c\}$, and $Y = \{y_1, y_2, \cdots, y_v\}$ identify $x_f (1 \le f \le z)$ with $\boldsymbol{\delta}_z^f (1 \le f \le z)$, expressed as $x_f \square \boldsymbol{\delta}_z^f$, and $\boldsymbol{\delta}_z^f (1 \le f \le z)$ is the vector form of $x_f (1 \le f \le z)$. Then, $X$ can be denoted as $\boldsymbol{\Delta}_z$, that is, $X = \{\boldsymbol{\delta}_z^1, \boldsymbol{\delta}_z^2, \cdots, \boldsymbol{\delta}_z^z\}$. Similarly, for $E$, identify $e_g$ with $\boldsymbol{\delta}_c^g (1 \le g \le c)$ ( denoted as $e_g \square \boldsymbol{\delta}_c^g$ ), $\boldsymbol{\delta}_c^g (1 \le g \le c)$ is the vector form of $e_g$, and we use $\boldsymbol{\Delta}_c$ to denote $E$, i.e., $E = \{\boldsymbol{\delta}_c^1, \boldsymbol{\delta}_c^2, \cdots, \boldsymbol{\delta}_c^c\}$. We do the same with $y_h (1 \le h \le v)$, that is, identify $y_h$ with $\boldsymbol{\delta}_v^h (1 \le h \le v)$, denoted as $y_h \square \boldsymbol{\delta}_v^h$; $\boldsymbol{\delta}_v^h$ is the vector form of $y_h$, then $Y$ can be denoted as $\boldsymbol{\Delta}_v$, i.e., $Y = \{\boldsymbol{\delta}_v^1, \boldsymbol{\delta}_v^2, \cdots, \boldsymbol{\delta}_v^v\}$.

**Definition 7** (Yan et al., 2015) $\boldsymbol{F}' = [\boldsymbol{F}_1 \ldots \boldsymbol{F}_m]$ is called the STSM of normal FSM $A = (X, E, Y, f, g, x^0, X^m)$, which is defined as:

$$\boldsymbol{F}'_{i(s,f)} = \begin{cases} 1 & \boldsymbol{\delta}_z^s \in f_1(\boldsymbol{\delta}_z^f, \boldsymbol{\delta}_c^i) \\ 0 & \text{otherwise} \end{cases}. \tag{6}$$

**Definition 8** (Yan et al., 2015) $\boldsymbol{H}' = [\boldsymbol{H}_1 \ldots \boldsymbol{H}_m]$ is called the SOSM of normal FSM $A = (X, E, Y, f, g, x^0, X^m)$, which is defined as:

$$\boldsymbol{H}'_{i(h,t)} = \begin{cases} 1 & \boldsymbol{\delta}_v^h \in g_1(\boldsymbol{\delta}_z^f, \boldsymbol{\delta}_c^i) \\ 0 & \text{otherwise} \end{cases}. \tag{7}$$

**Remark 3** In the framework of the STP, $\boldsymbol{F}'$ is uniquely determined by $e_i$ in normal FSMs, and $\boldsymbol{F}$ is uniquely determined by $e_j$ and $g_a$ in EFSMs, which is the essential difference. At the same time, $\boldsymbol{F}'$ and $\boldsymbol{F}$ represent the dynamic logical evolution

unedited

of all states of normal FSMs and EFSMs, which means that the relevant research and properties of normal FSMs do not apply to EFSMs. Similarly, $\boldsymbol{H}'$ and $\boldsymbol{H}$ are also the case.

**Theorem 1** Let $\boldsymbol{F}=[\boldsymbol{F}_1\ldots\boldsymbol{F}_m]$ be the STSM of an EFSM $A=(L,D,\Sigma,A_s,L_0,D_0,L_m)$, $L=\{l_1,l_2,\cdots,l_n\}$, $\Sigma=\{e_1,e_2,\cdots,e_m\}$, $g_\alpha=\left\{\boldsymbol{\delta}_2^1,\boldsymbol{\delta}_2^2\right\}$. Then, after the EFSM reads input $e_j$ at any $l_i$, the new state is

$$l_{i+1}=\boldsymbol{F}\,\text{寸}\boldsymbol{\delta}_m^j\quad\boldsymbol{\delta}_n^i\,?\,\boldsymbol{\delta}_2^k, \tag{8}$$

where $\boldsymbol{\delta}_m^j$ is the vector form of the input $e_j$, $\boldsymbol{\delta}_n^i$ is the vector form of state $l_i(1\le i\le n)$, and $\boldsymbol{\delta}_2^k$ is the judgment of the input sequence of the dynamic system in the EFSM.

**Proof** The transition function of an EFSM $A$ is $f:L_0\times\Sigma\to L_m$, which can be expressed as

$$f(\boldsymbol{\delta}_n^i,\boldsymbol{\delta}_m^j)=\boldsymbol{F}\times\boldsymbol{\delta}_m^j\times\boldsymbol{\delta}_n^i, \tag{9}$$

where $\boldsymbol{\delta}_n^i$ and $\boldsymbol{\delta}_m^j$ are vector forms of state $l_i(1\le i\le n)$ and event $e_j(1\le j\le m)$, respectively.

The EFSM $A$ reads input $e_j(1\le j\le m)$ at the state $l_i$ and moves to the state $l_{i+1}$ after being judged to be "true", i.e., $f(l_i,e_j)=l_{i+1}$.

According to Definition 5, the column judged "true" by $col_i(\boldsymbol{F}_j)$ contains one element equal to 1; all the other elements are 0.

Now, we can calculate $\boldsymbol{F}\times\boldsymbol{\delta}_m^j\times\boldsymbol{\delta}_n^i$, but we should be careful. $\boldsymbol{F}\times\boldsymbol{\delta}_m^j\times\boldsymbol{\delta}_n^i$ contains the vectors under two different decisions between state $l_i$ and state $l_{i+1}$ after reading input $e_j(1\le j\le m)$, i.e., state $l_i$ through state $l_{i+1}$ are judged "true" and "false". To make clear the state after reading the input, the true state after reading the input can be obtained by multiplying the judgment vector after $\boldsymbol{F}\times\boldsymbol{\delta}_m^j\times\boldsymbol{\delta}_n^i$, i.e., $\boldsymbol{F}\times\boldsymbol{\delta}_m^j\times\boldsymbol{\delta}_n^i$ â $\boldsymbol{\delta}_2^k$.

Then, after the EFSM reads input $e_j(1\le j\le m)$ at any state $l_i$, the new state is $l_{i+1}=\boldsymbol{F}\,\text{寸}\boldsymbol{\delta}_m^j\quad\boldsymbol{\delta}_n^i\,?\,\boldsymbol{\delta}_2^k$.

**Remark 4** We have obtained a method for receiving events in any initial state and obtaining the new state, which provides a solid mathematical foundation for determining the controllability, reachability, and stabilizability of the EFSM. For details, we can refer to Examples 1 and 2.

**Theorem 2** Let $\boldsymbol{H}=[\boldsymbol{H}_1\ldots\boldsymbol{H}_m]$ be the SOSM of an EFSM $A=(L,D,\Sigma,A_s,L_0,D_0,L_m)$, $L=\{l_1,l_2,\cdots,l_n\}$, $\Sigma=\{e_1,e_2,\cdots,e_m\}$, $g_\alpha=\left\{\boldsymbol{\delta}_2^1,\ \boldsymbol{\delta}_2^2\right\}$. Then, after the EFSM reads the input $e_j(1\le j\le m)$ at any state value $d_i$, the new state value is

$$d_{i+1}=\boldsymbol{H}\,\text{寸}\boldsymbol{\delta}_m^j\quad\boldsymbol{\delta}_p^z\,?\,\boldsymbol{\delta}_2^k, \tag{10}$$

where $\boldsymbol{\delta}_m^j$ is the vector form of the input sequence $e_1,e_2,\cdots,e_m$, $\boldsymbol{\delta}_p^z$ is the vector form of the data variable $d_z(1\le z\le p)$, and $\boldsymbol{\delta}_2^k$ is the judgment of transition matrix of the dynamic system in EFSM.

**Proof** The proof is omitted since it is very similar to that of Theorem 1.

## 4 Controllability, Reachability, and Stabilizability of EFSMs

In this section, we study the controllability, reachability, and stabilizability of the EFSM by using the bilinear dynamic system model of the EFSM in Section 3.

**Theorem 3** The state $l_i\,\square\,\boldsymbol{\delta}_n^i\in L$ is controllable to the state $l_j\,\square\,\boldsymbol{\delta}_n^j\in L$, if and only if there exists a finite input sequence $e_j\,\square\,\boldsymbol{\delta}_m^j\in\Sigma$ such that $g_a$ is judged to be true.

**Proof** (Necessity) If state $l_i\,\square\,\boldsymbol{\delta}_n^i\in L$ is controllable to state $l_j\,\square\,\boldsymbol{\delta}_n^j\in L$, it also means that there exists a finite sequence of inputs $e_j\,\square\,\boldsymbol{\delta}_m^j\in\Sigma$ such that the EFSM operates properly, then $g_a$ must be judged true at this point.

(Sufficiency) We assume that there is a finite input sequence $e_j\,\square\,\boldsymbol{\delta}_m^j\in\Sigma$ that makes $g_a$ judged to be true, then the EFSM operates properly at this time. Then, we can obtain that $l_i\,\square\,\boldsymbol{\delta}_n^i\in L$ to $l_j\,\square\,\boldsymbol{\delta}_n^j\in L$ is controllable.

Next, we will provide an example to illustrate Theorem 3.

**Example 1** Suppose there is an EFSM $A_1$ as

unedited

shown in Fig. 1, where $IN$ represents the input of the EFSM, $\varnothing$ represents that the output is 0 when $g_\alpha$ is judged to be false, $L=\{l_1,l_2,l_3\}$, $\Sigma=\{e_1,e_2\}$.

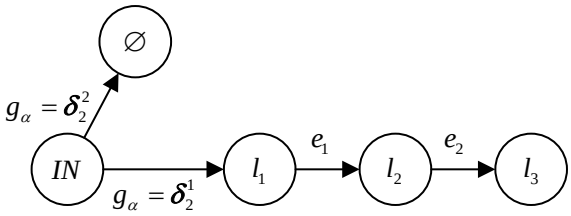


**Fig. 1 The state transition diagram of $A_1$**

It can be seen from Fig. 1 that state $l_3$ in $L$ can be obtained by receiving events $e_1$ and $e_2$ from states $l_1$ and $l_2$, with a state transition path of $l_1 \xrightarrow{e_1} l_2 \xrightarrow{e_2} l_3$. At this time, state $l_1$ is controllable to state $l_3$. We can also think of the state transition path as $l_2 \xrightarrow{e_2} l_3$, so that state $l_2$ is controllable to state $l_3$.

Under the STP framework, the STSM of the EFSM in Fig. 1 is $\boldsymbol{F}=\delta_3[2\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 3\ 0\ 0\ 0]$.

Then, the state transition path $l_1 \xrightarrow{e_1} l_2 \xrightarrow{e_2} l_3$ is $\boldsymbol{F} \ltimes \boldsymbol{\delta}_2^1 \quad \boldsymbol{\delta}_3^1 ? \boldsymbol{\delta}_2^1=\boldsymbol{\delta}_3^2$ and $\boldsymbol{F} \ltimes \boldsymbol{\delta}_2^2 \quad \boldsymbol{\delta}_3^2 ? \boldsymbol{\delta}_2^1=\boldsymbol{\delta}_3^3$, and the state transition path $l_2 \xrightarrow{e_2} l_3$ is $\boldsymbol{F} \ltimes \boldsymbol{\delta}_3^2 \quad \boldsymbol{\delta}_2^2 ? \boldsymbol{\delta}_2^1=\boldsymbol{\delta}_3^3$. According to Theorem 3, we can know that state $l_1$ is controllable to state $l_3$, and state $l_2$ is also controllable to state $l_3$.

According to the contents and proof of Theorem 3, we can make the following corollary.

**Corollary 1** The state $l_j \square \boldsymbol{\delta}_n^j \in L$ is reachable from the state $l_i \square \boldsymbol{\delta}_n^i \in L$ if and only if there exists a finite input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$ such that $g_a$ is judged to be true.

The proof of Corollary 1 is similar to the proof of Theorem 3 and is thus omitted. Moreover, Corollary 1 can be more easily understood by referring to the description of Theorem 3 in example 1.

Then, we give the concepts of the controllability, reachability, and stabilizability of nonempty sets of states.

**Theorem 4** $L_1 \subseteq L$ is controllable if and only if there exists an input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$ and $g_a$ is judged to be true such that all $l_\alpha \square \boldsymbol{\delta}_n^\alpha$ in $L_1$ are controllable.

**Proof** (Necessity) Suppose that $L_1 \subseteq L$ is controllable. We can know for any state $l_\beta \square \boldsymbol{\delta}_n^\beta \in L_2$, and there exists a state $l_\alpha \square \boldsymbol{\delta}_n^\alpha \in L_1$ such that $l_\alpha \square \boldsymbol{\delta}_n^\alpha$ is controllable to $l_\beta \square \boldsymbol{\delta}_n^\beta$. Based on the conclusion of Theorem 3, there exists an input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$, and $g_a$ is judged to be true.

(Sufficiency) Since there exists an input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$, and $g_a$ is judged to be true such that all $l_\alpha \square \boldsymbol{\delta}_n^\alpha$ in $L_1$ are controllable, then we can get that $L_1 \subseteq L$ is controllable.

Next, we will provide an example to illustrate Theorem 4.

**Example 2** Suppose there is an EFSM $A_2$ as shown in Fig. 2, where $IN$ represents the input of the EFSM, $\varnothing$ represents that the output is 0 when $g_\alpha$ is judged to be false, $L=\{l_1,l_2,l_3,l_4,l_5\}$, $L_1=\{l_1,l_2,l_3\}$, $L_2=\{l_4,l_5\}$, $\Sigma=\{e_1,e_2\}$.

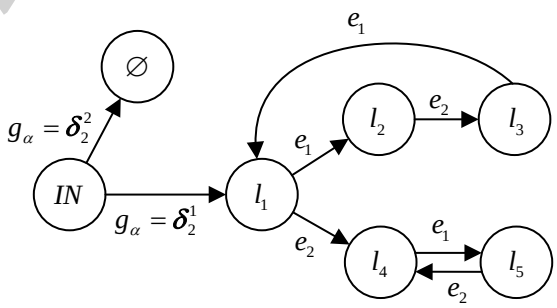


**Fig. 2 The state transition diagram of $A_2$**

As shown in Fig. 2, there is a state transition path $l_1 \xrightarrow{e_1} l_2 \xrightarrow{e_2} l_3 \xrightarrow{e_1} l_1$ in $L_1$, indicating that $L_1$ is controllable. Similarly, there is a state transition path $l_4 \xrightarrow{e_1} l_5 \xrightarrow{e_2} l_4$ in $L_2$, indicating that $L_2$ is controllable.

Under the STP framework, the STSM of the EFSM in Fig. 2 is $\boldsymbol{F}=\delta_5[2\ 0\ 0\ 0\ 1\ 0\ 5\ 0\ 0\ 0\ 4\ 0\ 3\ 0\ 0\ 0\ 0\ 0\ 4\ 0]$.

Then, the state transition path $l_1 \xrightarrow{e_1} l_2 \xrightarrow{e_2} l_3 \xrightarrow{e_1} l_1$ is $\boldsymbol{F} \ltimes \boldsymbol{\delta}_2^1 \quad \boldsymbol{\delta}_5^1 ? \boldsymbol{\delta}_2^1=\boldsymbol{\delta}_5^2$, $\boldsymbol{F} \ltimes \boldsymbol{\delta}_2^2 \quad \boldsymbol{\delta}_5^2 ? \boldsymbol{\delta}_2^1=\boldsymbol{\delta}_5^3$, and $\boldsymbol{F} \ltimes \boldsymbol{\delta}_2^1 \quad \boldsymbol{\delta}_5^3 ? \boldsymbol{\delta}_2^1=\boldsymbol{\delta}_5^1$,

unedited

and the state transition path $l_4 \xrightarrow{e_1} l_5 \xrightarrow{e_2} l_4$ is $\boldsymbol{F} \ltimes \boldsymbol{\delta}_2^1 \quad \boldsymbol{\delta}_5^4 ? \boldsymbol{\delta}_2^1 = \boldsymbol{\delta}_5^5$ and $\boldsymbol{F} \ltimes \boldsymbol{\delta}_2^2 \quad \boldsymbol{\delta}_5^5 ? \boldsymbol{\delta}_2^1 = \boldsymbol{\delta}_5^4$ . According to Theorem 4, we can know that if $L_1$ is controllable, $L_2$ is also controllable.

**Remark 5** Through Examples 1 and 2, it can be seen that, under the STP framework, we only need to find out the STSM of the EFSM according to Definition 5 and perform simple mathematical calculations to determine whether the corresponding state and state set are controllable. The method is simple and effective.

According to the contents and proof of Theorem 4, we can make the following corollary.

**Corollary 2** $L_2 \subseteq L$ is reachable, if and only if there exists an input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$ and $g_a$ is judged to be true such that all $l_\beta \square \boldsymbol{\delta}_n^\beta$ in $L_2$ are reachable.

The proof of Corollary 2 is similar to the proof of Theorem 4 and is thus omitted. Moreover, Corollary 2 can be more easily understood by referring to the description of Theorem 4 in Example 2.

**Theorem 5** $L_1 \subseteq L$ is one-step returnable, if and only if there exists an input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$ and $g_a$ is judged to be true such that all $l_\alpha \square \boldsymbol{\delta}_n^\alpha$ in $L_1$ are mutually reachable.

**Proof** (Necessity) Suppose that $L_1 \subseteq L$ is one-step returnable. We can know that all $l_\alpha \square \boldsymbol{\delta}_n^\alpha$ in $L_1$ are reachable to one another. Based on the content of Section 2, there exists an input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$, and $g_a$ is judged to be true.

(Sufficiency) Since there exists an input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$, and $g_a$ is judged to be true such that all $l_\alpha \square \boldsymbol{\delta}_n^\alpha$ in $L_1$ are mutually reachable, then we can get that $L_1 \subseteq L$ is one-step returnable.

Based on the content of Section 2, Theorem 4, and Theorem 5, we have the following result on the stabilizability of the EFSM.

**Theorem 6** $L_1 \subseteq L$ is stabilizable, if and only if Theorem 4 and Theorem 5 hold.

We have introduced the controllability, reachability, and stabilizability from control theory into bilinear dynamic system models of EFSMs. Next, we design an algorithm to judge the controllability, one-step returnability, and stabilizability of bilinear dynamic system models of EFSMs.

**Algorithm 1**

**Input:** The STSM $F$ of EFSM $A$.

**Output:** The controllable state set $S_C$, the one-step returnable state set $S_O$ and stabilizable state set $S$.

1. Let $i = 1$, for $l_i$, if there exists a finite input sequence $e_j \square \boldsymbol{\delta}_m^j \in \Sigma$ such that $\boldsymbol{F} \ltimes \boldsymbol{\delta}_m^j \quad \boldsymbol{\delta}_n^j ? \boldsymbol{\delta}_2^1 = \boldsymbol{\delta}_n^i$. Then, $l_i$ is controllable, else $l_i$ is not controllable.
2. Let $i = i + 1$, repeat step 1, until $i = n$, check whether $l_i$ is controllable. If YES, we put it into the controllable state set $S_C$.
3. Let $i = 1$, consider $l_i$, if there exists a set of states such that $F \ltimes \boldsymbol{\delta}_m^j \quad \boldsymbol{\delta}_n^i ? \boldsymbol{\delta}_2^1 = \boldsymbol{\delta}_n^{i+1}$, $\boldsymbol{F} \ltimes \boldsymbol{\delta}_m^j \quad \boldsymbol{\delta}_n^{i+1} \ltimes \boldsymbol{\delta}_2^1 = \boldsymbol{\delta}_n^{i+2}, \cdots, \boldsymbol{F} \quad \boldsymbol{\delta}_m^j \ltimes \boldsymbol{\delta}_n^{i+t} \quad \boldsymbol{\delta}_2^1 = \boldsymbol{\delta}_n^i$, then these states are considered mutually reachable.
4. Let $i = i + 1$, repeat step 3, until $i = n$; if $l_\alpha \square \boldsymbol{\delta}_n^\alpha$ in $L_1$ are mutually reachable, we put it into the one-step returnable state set $S_O$.
5. The stabilizable state set of EFSM is $S = S_C \cap S_O$.

**Remark 6** In Algorithm 1, we first traverse all states of the EFSM based on Theorem 3 and Theorem 4 to find the controllable state set of the EFSM, then find all reachable states based on Corollary 1, judge all mutually reachable states according to Theorem 5, find the one-step returnable state set, and finally find the stabilizable state set of the EFSM according to Theorem 6. Algorithm 1 can be applied to various models of EFSMs in different fields, enabling us to determine their controllability, one-step returnability, and stabilizability. Using algebraic computation to accurately understand the properties of the state of the EFSM under various conditions helps us to develop effective control strategies.

unedited

## 5 An Illustrative Example

In this section, we provide an illustrative example to confirm our results' correctness and effectiveness.

**Example 3** Suppose there is an EFSM $A_3$ as shown in Fig. 3, where $IN$ represents the input of the EFSM, $\varnothing$ represents the output being 0 when $g_\alpha$ is judged to be false, $L=\{l_1,l_2,l_3,l_4,l_5,l_6\}$, $L_1=(l_1,l_2,l_3)$, and $L_2=(l_4,l_5,l_6)$, $\Sigma=\{e_1,e_2\}$.

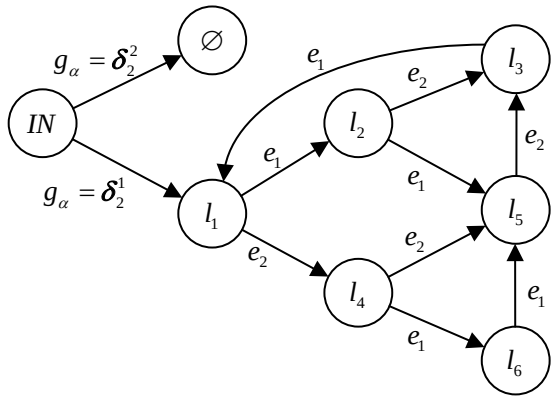


**Fig. 3 The state transition diagram of $A_3$**

In Fig. 3, when the state starts at $l_1$, reads input $e_1$, gets result $l_2$, $l_1$ receives signal $e_2$ and gets result $l_4$; $l_2$ receives signal $e_1$ and gets result $l_5$, $l_2$ receives signal $e_2$ and gets result $l_3$, $l_3$ receives signal $e_1$ and gets result $l_1$; $l_4$ receives signal $e_1$ and gets result $l_6$, $l_4$ receives signal $e_2$ and gets result $l_5$; $l_5$ receives signal $e_2$ and gets result $l_3$; $l_6$ receives signal $e_1$ and gets result $l_5$.

In the dynamic system model of the EFSM, $L=(\delta_6^1,\delta_6^2,\delta_6^3,\delta_6^4,\delta_6^5,\delta_6^6)$, $L_1=(\delta_6^1,\delta_6^2,\delta_6^3)$, and $L_2=(\delta_6^4,\delta_6^5,\delta_6^6)$, $e=(\delta_2^1,\delta_2^2)$.

Then, the state starts at $l_1 \square \delta_6^1$, reads input $e_1 \square \delta_2^1$, gets result $l_2 \square \delta_6^2$, $l_1 \square \delta_6^1$ receives signal $e_2 \square \delta_2^2$ and gets result $l_4 \square \delta_6^4$; $l_2 \square \delta_6^2$ receives signal $e_1 \square \delta_2^1$ and gets result $l_5 \square \delta_6^5$, $l_2 \square \delta_6^2$ receives signal $e_2 \square \delta_2^2$ and gets result $l_3 \square \delta_6^3$, $l_3 \square \delta_6^3$ receives signal $e_1 \square \delta_2^1$ and gets result $l_1 \square \delta_6^1$; $l_4 \square \delta_6^4$ receives signal $e_1 \square \delta_2^1$ and gets result $l_6 \square \delta_6^1$, $l_4 \square \delta_6^4$ receives signal $e_2 \square \delta_2^2$ and gets result $l_5 \square \delta_6^5$; $l_5 \square \delta_6^5$ receives signal $e_2 \square \delta_2^2$ and gets result $l_3 \square \delta_6^3$; $l_6 \square \delta_6^6$ receives signal $e_1 \square \delta_2^1$ and gets result $l_5 \square \delta_6^5$.

Next, we use Algorithm 1 to judge the controllability, one-step returnability, and stabilizability of the EFSM $A_3$.

We find the STSM $F$, i.e., $\boldsymbol{F}=\delta_6[2\ 0\ 5\ 0\ 1\ 0\ 6\ 0\ 0\ 0\ 5\ 0\ 4\ 0\ 3\ 0\ 0\ 0\ 5\ 0\ 3\ 0\ 0\ 0]$.

**Step 1** Based on Theorem 3, it determines whether the state of the EFSM $A_3$ is controllable. $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^1?\ \delta_2^1=\delta_6^2$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^1?\ \delta_2^1=\delta_6^4$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^2?\ \delta_2^1=\delta_6^3$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^2?\ \delta_2^1=\delta_6^5$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^3?\ \delta_2^1=\delta_6^1$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^4?\ \delta_2^1=\delta_6^5$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^4?\ \delta_2^1=\delta_6^6$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^5?\ \delta_2^1=\delta_6^3$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^6?\ \delta_2^1=\delta_6^5$.

As shown above, we test the controllability of all states using algebraic methods, and we can conclude that states $l_1$, $l_2$, $l_3$, $l_4$, $l_5$, and $l_6$ are controllable.

**Step 2** Based on Theorem 4, combined with Step 1, we know that nonempty sets of states $S_{C_1}=L_1$ and $S_{C_2}=L_2$ are controllable.

**Step 3** Based on Corollary 1, it determines whether the state of the EFSM $A_3$ is reachability. $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^1?\ \delta_2^1=\delta_6^2$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^1?\ \delta_2^1=\delta_6^4$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^2?\ \delta_2^1=\delta_6^3$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^2?\ \delta_2^1=\delta_6^5$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^3?\ \delta_2^1=\delta_6^1$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^4?\ \delta_2^1=\delta_6^5$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^4?\ \delta_2^1=\delta_6^6$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^5?\ \delta_2^1=\delta_6^3$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^6?\ \delta_2^1=\delta_6^5$.

As shown above, we test the reachability of all states using algebraic methods, and we can conclude that states $l_1$, $l_2$, $l_3$, $l_4$, $l_5$, and $l_6$ are reachable.

**Step 4** In an EFSM $A_3$, only $\delta_6^1\to\delta_6^2\to\delta_6^3\to\delta_6^1$ : $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^1?\ \delta_2^1=\delta_6^2$, $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^2?\ \delta_2^1=\delta_6^3$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^3?\ \delta_2^1=\delta_6^1$; $\delta_6^2\to\delta_6^3\to\delta_6^1\to\delta_6^2$ : $\boldsymbol{F}\ 寸\delta_2^2\ \ \delta_6^2?\ \delta_2^1=\delta_6^3$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^3?\ \delta_2^1=\delta_6^1$, $\boldsymbol{F}\ 寸\delta_2^1\ \ \delta_6^1?\ \delta_2^1=\delta_6^2$;

unedited

$\delta_6^3 \to \delta_6^1 \to \delta_6^2 \to \delta_6^3$ : $\boldsymbol{F}$ 忖$\delta_2^1$ $\delta_6^3$? $\delta_2^1 = \delta_6^1$ , $\boldsymbol{F}$ 忖$\delta_2^1$ $\delta_6^1$? $\delta_2^1 = \delta_6^2$ , $\boldsymbol{F}$ 忖$\delta_2^2$ $\delta_6^2$? $\delta_2^1 = \delta_6^3$ in $L_1$ are one-step returnable, that is, $S_O = L_1$ is one-step returnable.

**Step 5** In this case, we can find that the stabilizable state set of the EFSM $A_3$ is $S = S_C \bigcap S_O = L_1$.

## 6 Conclusion

In this paper, the controllability, reachability, and stabilizability of EFSMs were studied by using the STP method. Firstly, we established the bilinear dynamic system model of the EFSM under the STP framework, which enabled us to analyze the dynamic behavior of the EFSM. Based on this, we studied the controllability, reachability, and stabilizability of the EFSM and proposed the relevant theorems. Further, we designed an algorithm to determine the related properties of EFSMs. In addition, the observability of the EFSM can be analyzed based on the above method. Overall, our approach contributes to a more in-depth study of EFSMs.

### Contributors

Chao DONG and Yongyi YAN designed the research. Huiqin LI and Jumei YUE processed the data. Chao DONG drafted the manuscript. Yongyi YAN, Huiqin LI and Jumei YUE helped organize the manuscript. Chao DONG, Yongyi YAN, Huiqin LI and Jumei YUE revised and finalized the paper.

### Compliance with ethics guidelines

Chao DONG, Yongyi YAN, Huiqin LI and Jumei YUE declare that they have no conflict of interest.

### Data availability

The authors confirm that the data supporting the findings of this study are available within the paper.

### References


Adak S, Mukherjee S, Das S, 2021. Reachability problem in non-uniform cellular automata. *Inform Sciences*, 543:72-84. https://doi.org/10.1016/j.ins.2020.07.034

Boukerrou H, Millerioux G, Minier M, et al., 2022. Construction of dead-beat switched automata: application to cryptography. 10th International Conference on Systems and Control, p.23-30. https://doi.org/10.1109/ICSC57768.2022.9993895

Chang Z, Ge AD, Sun MC, 2022. Interval type-2 fuzzy logic system based on Semi-tensor product. 41st Chinese Control Conference, p.2507-2512.

Cheng DZ, Qi HS, Li ZQ, 2010. Analysis and control of Boolean networks: A Semi-tensor product approach. Springer, London, p.29-37.

Dou WH, Li HT, Alsaadi FE, 2019. Semi-tensor product approach to controllability, reachability, and stabilizability of probabilistic finite automata. *Math Probl Eng*, 1-7. http://dx.dor.org/10.1155/2019/8021750

Dridi S, El Yacoubi S, Bagnoli F, 2022. Kalman condition and new algorithm approach for regional controllability of Peripherally-linear elementary cellular automata via boundary actions. *J Cell Autom*, 16(3-4):173-195.

Foster M, Taylor RG, Brucker AD, et al., 2018. Formalising extended finite state machine transition merging. 20th International Conference on Formal Engineering Methods, p.373-387. http://dx.dor.org/10.1007/978-3-030-02450-5_22

Gluck R, 2023. Compatibility of refining and controlling plant automata with bisimulation quotients. 20th International Conference on Relational and Algebraic Methods in Computer Science, p.87-104. https://doi.org/10.1007/978-3-031-28083-2_6

Goorden M, van de Mortel-Fronczak J, Reniers M, et al., 2020. Structuring multilevel discrete-event systems with dependence structure matrices. *IEEE T Automat Contr*, 65(4):1625-1639. https://doi.org/10.1109/TAC.2019.2928119

Han XG, Chen ZQ, 2018. A matrix-based approach to verifying stability and synthesizing optimal stabilizing controllers for finite-state automata. *J Franklin I*, 355(17):8642-8663. https://doi.org/10.1016/j.jfranklin.2018.09.009

Han XG, Chen ZQ, Liu ZX, et al., 2018. The detection and stabilisation of limit cycle for deterministic finite automata. *Int J Control*, 91(4):874-886. https://doi.org/10.1080/00207179.2017.1295319

Henry L, Jeron T, Markey N, 2023. Control strategies for offline testing of timed systems. *Form Method Syst Des*, 60(2):147-194. https://doi.org/10.1007/s10703-022-00403-w

Kocher C, 2021. Reachability problems on reliable and lossy queue automata. *Theor Comput Syst*, 65(8):1211-1242. https://doi.org/10.1007/s00224-021-10031-2

Kong XS, Li HT, Gu EG, 2023. Lyapunov-based sampled-data set stabilisation of boolean control networks with time delay and state constraint. *Int J Control*. http://dx.dor.org/10.1080/00207179.2023.2191151

Li HT, Zhao GD, Meng M, et al., 2018. A survey on applications of semi-tensor product method in engineering. *Sci China Inform Sci*, 61(1). http://dx.dor.org/10.1007/s11432-017-9238-1

Lin YD, Lai YK, Bui QT, et al., 2020. ReFSM: Reverse engineering from protocol packet traces to test generation by extended finite state machines. *J Netw Comput Appl*, 171. https://doi.org/10.1016/j.jnca.2020.102819


unedited

Li WR and Li HT, 2023. Set reachability and set stability of Boolean networks with state-dependent asynchronous updating rule. *Asian J Control*, in press.
http://dx.dor.org/10.1002/asjc.3159

Lv ZY, Wu YH, Zhao Q, et al., 2021. Design and control of a novel coaxial tilt-rotor UAV. *IEEE T Ind Electron*, 69(4):3810-3821.
http://dx.dor.org/10.1109/tie.2021.3075886

Pilch C, Schupp S, Remke A, 2021. Optimizing reachability probabilities for a restricted class of stochastic hybrid automata via flowpipe-construction. 18th International Conference on Quantitative Evaluation Systems, p.435-456.
https://doi.org/10.1007/978-3-030-85172-9_23

Sun CL and Li HT, 2023. State-flipped control and Q-learning for finite horizon output tracking of Boolean control networks. *Int J Syst Sci*, 54(12):2452-2464.
http://dx.dor.org/10.1080/00207721.2023.2230196

Wang B, and Feng JE, 2019. On detectability of probabilistic Boolean networks. *Inform Sciences*, 483:383-395.
http://dx.dor.org/10.1016/j.ins.2019.01.055

Wang SL, Li HT, 2019. Column stacking approach to resolution of systems of fuzzy relational inequalities. *J Franklin I*, 356(6):3314-3332.
http://dx.dor.org/10.1016/j.jfranklin.2019.02.007

Wang XJ, Luo C, Li C, 2022. The feedback stabilization of finite-state fuzzy cognitive maps. *T I Meas Control*, 44(13):2485-2499.
http://dx.dor.org/10.1177/01423312221085785

Wang YH, Cheng DZ, Liu XY, 2019. Matrix expression of shapley values and its application to distributed resource allocation. *Sci China Inform Sci*, 62(2).
http://dx.dor.org/10.1007/s11432-018-9414-5

Wu YH, Shen TL, 2016. Policy iteration approach to control residual gas fraction in IC engines under the framework of stochastic logical dynamics. *IEEE T Contr Syst T*, 25(3):1100-1107.
http://dx.dor.org/10.1109/tcst.2016.2587247

Yao LH, and Li JM, 2017. Input–output finite-time stabilization of a class of nonlinear hybrid systems based on FSM with MDADT. *J Franklin I*, 354(9):3797-3812.
http://dx.dor.org/10.1016/j.jfranklin.2016.07.013

Yan YY, Cheng DZ, Feng JE, et al., 2023. Survey on applications of algebraic state space theory of logical systems to finite state machines. *Sci China Inform Sci*, 66(1).
http://dx.dor.org/10.1007/s11432-022-3538-4

Yan YY, Chen ZQ, Liu ZX, 2015. Semi-tensor product approach to controllability and stabilizability of finite automata. *J Syst Eng Electron*, 26(1):134-141.
http://dx.dor.org/10.1109/JSEE.2015.00018

Yan YY, Yue JM, Chen ZQ, 2022. Observed data-based model construction of finite state machines using exponential representation of LMs. *IEEE T Circuits-II*, 69(2):434-438.
http://dx.dor.org/10.1109/tcsii.2021.3087189

Zahn F, Kleinert T, Hagenmeyer V, 2022. Assessing the combination of differential flatness and deterministic automata for controllable hybrid systems. IEEE 61st Conference on Decision and Control, p.2612-2619.
https://doi.org/10.1109/CDC51059.2022.9993043

Zhang ZP, Chen ZQ, Han XG, et al., 2020. Stabilization of probabilistic finite automata based on semi-tensor product of matrices. *J Franklin I*, 357(9):5173-5186.
https://doi.org/10.1016/j.jfranklin.2020.02.028

Zoubeyr F, Tari A, Ouksel AM, 2010. Backward validation of communicating complex state machines in web services environments. *Distrib Parallel Dat*, 27(3):255-270.
http://dx.dor.org/10.1007/s10619-010-7061-8

unedited