SCIENCE CHINA
Information Sciences

• LETTER •

December 2018, Vol. 61 129202:1–129202:3
https://doi.org/10.1007/s11432-018-9507-7

Modeling and reachability of probabilistic finite automata based on semi-tensor product of matrices

Zhipeng ZHANG[1,2], Zengqiang CHEN[1,2*] & Zhongxin LIU[1,2]

[1]College of Computer and Control Engineering, Nankai University, Tianjin 300350, China;
[2]Tianjin Key Laboratory of Intelligent Robotics, Nankai University, Tianjin 300350, China

Received 24 May 2018/Accepted 15 June 2018/Published online 7 November 2018

Citation Zhang Z P, Chen Z Q, Liu Z X. Modeling and reachability of probabilistic finite automata based on semi-tensor product of matrices. Sci China Inf Sci, 2018, 61(12): 129202, https://doi.org/10.1007/s11432-018-9507-7

Dear editor,

As mathematical models, finite automata have been a powerful synthesis tool for modeling and analyzing discrete event systems where states and events are a finite logical or discrete set. In deterministic finite automata, the successor can be uniquely determined by the predecessor and input event. In other words, it exhibits a deterministic behavior. As a practical and theoretical motivation, probabilistic finite automata (PFAs) have been shown to exhibit a stochastic behavior in many problems such as determining the reliability of sequential circuits. The research on PFAs almost always concentrates on the language-based approach: [1] considered the supervisory control of probabilistic discrete event systems modeled by PFAs, and [2] considered the state estimation of PFAs, which was a natural generalization of the detectability of finite automata.

More recently, a matrix-based approach to mix-valued logical networks based on the semi-tensor product (STP) of matrices was proposed by Cheng et al. [3]. It provided a nice systematic approach to analysis and design problems involving logical networks, including Boolean networks [3], probabilistic Boolean networks [4], game theory and finite automata. Under the framework of the STP of matrices, Ref. [5] equivalently gave the bilinear form of finite automata and provided a reachability analysis of them; Ref. [6] reported the detection and stabilization of the limit cycle for deterministic finite automata via state feedback control; and Ref. [7] examined the static output feedback stabilization of deterministic finite automata.

The contributions of this study include two aspects. First, we propose a matrix-based approach for PFAs under the framework of a STP of matrices, which generalizes the approach in [3]. Second, because the reachability analysis of PFAs is an interesting and important topic in problems involving blocking detection, and controllability, with the help of the new expression, we develop a systemic analysis of the reachability of PFAs, and provide a sufficient and necessary condition for the reachability.

Some notations are adopted as follows: $\mathbb{R}^n$ is the set of all vectors of dimension n. $\mathcal{M}_{m\times n}$ denotes the set of $m\times n$ matrices. I_n denotes the $n\times n$ dimensional identity matrix, and δ_n^k is the k-th column of identity matrix I_n. $M_{(i,j)}$ is the (i,j) element of matrix M. $\Delta_n := \{\delta_n^1, \delta_n^2, \ldots, \delta_n^n\}$. $|s|$ is the cardinality of string s. $\mathrm{Col}(M)$ represents the column set of matrix M, $\mathrm{Col}_i(M)$ represents the i-th column of matrix M, and $\mathrm{Col}_{i\Sigma}(M)$ represents the sum of the entries in the i-th column of matrix M. M^{T} is the transpose of matrix M. $\mathrm{E}[x]$ denotes the expected value of x.

Semi-tensor product of matrices. We first provide some of the necessary knowledge of the STP of matrices used in this study.

Definition 1 ([3]). Let $A \in \mathcal{M}_{m\times n}$, $B \in \mathcal{M}_{p\times q}$.

* Corresponding author (email: chenzq@nankai.edu.cn)

Then, the STP of A and B is defined as

$$A \ltimes B = (A \otimes I_{t/n})(B \otimes I_{t/p}), \quad (1)$$

where t denotes the least common multiple of n and p, i.e., $t = \mathrm{lcm}(n, p)$; $\otimes$ is the Kronecker product.

When n equals p, the STP coincides with the conventional matrix product. The symbol $\ltimes$ is omitted for convenience except for the special instructions in this study. In addition, $A_1 \ltimes A_2 \ltimes \cdots \ltimes A_n$ can be abbreviated as $\ltimes_{i=1}^{n} A_i$.

For $x \in \mathbb{R}_m$, $y \in \mathbb{R}_n$, $y \ltimes x = W_{[m,n]} \ltimes x \ltimes y$, where the matrix $W_{[m,n]}$ is the swap matrix, and can be defined as follows:

$$\begin{aligned} W_{[m,n]} = \delta_{mn}[&1, m+1, 2m+1, \ldots, (n-1)m+1, \\ &2, m+2, 2m+2, \ldots, (n-1)m+2, \ldots, \\ &m, 2m, 3m, \ldots, nm]. \end{aligned}$$

System model. PFAs[1)] can be captured by a six-tuple $\mathcal{A} = (X, \Sigma, \Gamma, f, P, x^0)$, where $X = \{x_1, x_2, \ldots, x_n\}$ represents a finite set of states with the initial state x^0; $\Sigma = \{\sigma_1, \sigma_2, \ldots, \sigma_m\}$ is a finite set of events called an alphabet; $\Gamma(x)$ is a set of feasible events defined for all $x \in X$ with $\Gamma(x) \in \Sigma$; $f : X \times \Sigma \to 2^X$ represents the transition function, which in general is a partial function on its domain; $P : X \times \Sigma \times X \to [0, 1]$ is a state transition probability function from state x_i to the state x_j with the occurrence of event σ_k, defined for all $x_i, x_j \in X$, $\sigma_k \in \Sigma$ such that $\sum_{\sigma_k \in \Gamma(x_i)} \sum_{x_j \in X} P(x_i, \sigma_k, x_j) = 1$ and $\sum_{\sigma_k \notin \Gamma(x_i)} \sum_{x_j \in X} P(x_i, \sigma_k, x_j) = 0$, and we refer to P_{ij}^k as such a probability distribution $P(x_i, \sigma_k, x_j)$ for brevity. Meanwhile, Σ^* denotes the finite set of all finite strings s on the alphabet Σ. Obviously, the transition function f and state transition probability function P can be extended over $s \in \Sigma^*$ by $f(x_i, s\sigma) = f(f(x_i, s), \sigma)$ and $P_{il}^{s\sigma} = \sum_{x_j \in X} P_{ij}^s P_{jl}^\sigma$, respectively. More details on PFAs can be found in [2, 8].

In the framework of the STP of matrices, it is possible to identify $x_i \sim \delta_n^i$ ($i \in [1, n]$) and $e_k \sim \delta_m^k$ ($k \in [1, m]$), where δ_n^i and δ_m^k are the vector forms of state x_i and input e_k, respectively.

With the help of the matrix-based approach reported in [5], a new transition probabilistic structure matrix of $\mathcal{A}$ can be defined as follows:

$$\mathcal{F} := [\mathcal{F}_1, \mathcal{F}_2, \ldots, \mathcal{F}_m] \in M_{n \times mn}, \quad (2)$$

where $\mathcal{F}_k \in \mathbb{R}^{n \times n}$ is a transition probabilistic structure matrix associated with event σ_k, which is defined as follows:

$$\mathcal{F}_{k(i,j)} = \begin{cases} P_{ij}^k, & \delta_n^j \in f(\delta_n^i, \delta_m^k), \\ 0, & \text{otherwise}. \end{cases} \quad (3)$$

Remark 1. It is quite natural to consider whether the state outputs of $\mathcal{A}$ involve stochastic behavior. The output probabilistic structure matrix can also be defined in a way similar to the definition of the transition probabilistic structure matrix.

Proposition 1. Given PFAs $\mathcal{A}$ and finite string $s \in \Sigma^*$, the dynamics of the PFAs can be described in an algebraic form as follows:

$$\mathrm{E}[x(t+1)] = \mathcal{F} u(t) \mathrm{E}[x(t)], \quad (4)$$

where $\mathrm{E}[x(t)]$ denotes the expected value of the state that is reached in t steps from $x(0)$, $u(t)$ is the input vector and $\mathcal{F}$ denotes the transition probabilistic structure matrix.

The proof is omitted here because it is very similar to that of Theorem 1 in [5]. In fact, $\mathcal{F}$ is the weighted adjacency matrix of the event σ-labeled sub-graph with respect to $\mathcal{A}$, and $\sum_{k=1}^m \mathcal{F}_k$ is the transpose of the probability transition matrix defined in [8].

Remark 2. Obviously, it is easy to verify that if $\Gamma(x_i) \neq \emptyset$, $\mathrm{Col}_{i\Sigma}(\sum_1^m \mathcal{F}_k) = 1$ is satisfied for each i. Otherwise, $\mathrm{Col}_{i\Sigma}(\sum_1^m \mathcal{F}_k) = 0$.

Remark 3. The probability transition matrix was introduced in [8], but the structural information contained in the event-driven description was lost. In other words, the probability transition matrix in [8] could not tell us the transition probabilistic distribution of the event that caused the transition. Under the framework of the STP of matrices, the stochastic behavior of the PFAs can be precisely captured in the transition probabilistic structure matrix defined in this study.

Reachability analysis of PFAs. The reachability analysis of PFAs is fundamental and important research in many control problems. Here, the reachability definition of the PFAs is introduced as follows.

Definition 2. Considering the PFAs in (4), (i) state $x^d = \delta_n^q \in X$ is said to be reachable from $x^0 = \delta_n^p$ with a finite input string $s \in \Sigma^*$ if $P_{pq}^s > 0$; (ii) state $x^d = \delta_n^q \in X$ is said to be reachable from $x^0 = \delta_n^p$ with a probability of one with a finite input string $s \in \Sigma^*$ if $P_{pq}^s = 1$.

With the help of the new expression, we have the following main result on the reachability of the PFAs.

1) PFAs can be rewritten as PFA for a given or specific probabilistic finite automaton.

Proposition 2. Considering the PFAs in (4), (i) state $x^d = \delta_n^q \in X$ is reachable from $x^0 = \delta_n^p$ by a finite input string $s \in \Sigma^*$, if and only if there exists r such that

$$\mathrm{Col}_r((\delta_n^q)^{\mathrm{T}}(\mathcal{F}W_{[n,m]})^{|s|}\delta_n^p) > 0; \tag{5}$$

(ii) state $x^d = \delta_n^q \in X$ is reachable from $x^0 = \delta_n^p$ with a probability of one by a finite input string $s \in \Sigma^*$, if and only if

$$\delta_n^q \in \mathrm{Col}((\mathcal{F}W_{[n,m]})^{|s|}\delta_n^p), \tag{6}$$

where $W_{[n,m]}$ denotes the swap matrix defined above.

Proof. According to the new expression (4) and through a direct mathematical induction, we can obtain

$$\begin{aligned}
\mathrm{E}[x(1)] &= \mathcal{F}W_{[n,m]}\mathrm{E}[x(0)]u(1)\\
\mathrm{E}[x(2)] &= \mathcal{F}W_{[n,m]}\mathrm{E}[x(1)]u(2)\\
&= (\mathcal{F}W_{[n,m]})^2\mathrm{E}[x(0)]u(1)u(2)\\
&\vdots\\
\mathrm{E}[x(|s|)] &= \mathcal{F}W_{[n,m]}\mathrm{E}[x(|s|-1)]u(|s|)\\
&= (\mathcal{F}W_{[n,m]})^{|s|}\mathrm{E}[x(0)] \ltimes_{k=1}^{|s|} u(k)\\
&= (\mathcal{F}W_{[n,m]})^{|s|}\delta_n^p \ltimes_{k=1}^{|s|} u(k).
\end{aligned}$$

(i) Based on the Definition 2, state $x^d = \delta_n^q \in X$ is reachable from $x^0 = \delta_n^p$ if and only if there exists r such that $\mathrm{Col}_r\{(\delta_n^q)^{\mathrm{T}}(\mathcal{F}W_{[n,m]})^{|s|}\delta_n^p\} > 0$.

(ii) State $x^d = \delta_n^q \in X$ is reachable from $x^0 = \delta_n^p$ with a probability of one if and only if its overall expected value of state $x(|s|)$ is δ_n^q, that is, $\delta_n^q \in \mathrm{Col}\{(\mathcal{F}W_{[n,m]})^{|s|}\delta_n^p\}$.

Example. The following example from [2] is illustrated in Figure 1 to validate the proposed result. The sets of states and events are rewritten in vector form, and the initial state and target state are $x^0 = \delta_6^1$ and $x^d = \delta_6^3$, respectively.

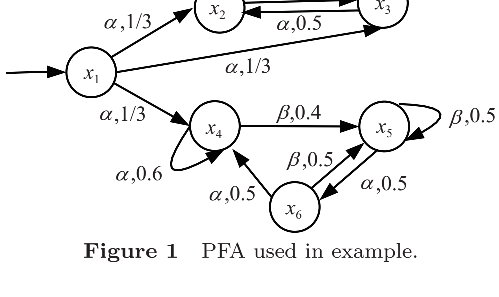

Figure 1 PFA used in example.

The algebraic form of the considered PFA can be described as formula (4), and the transition probabilistic structure matrix can be defined in the following matrix form:

$$\begin{bmatrix}
0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1/3 & 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1/3 & 0 & 0 & 0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0\\
1/3 & 0 & 0 & 0.6 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.4 & 0.5 & 0.5\\
0 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}.$$

When $|s| = 3$, using the properties of the STP of matrices, we can obtain matrix $(\mathcal{F}W_{[n,m]})^3\delta_6^1$, and it is easy to verify that (i) state $x^d = \delta_6^3$ is reachable from $x^0 = \delta_6^1$ with a finite input string of length 3; (ii) but no string of length 3 exists such that $x^d = \delta_6^3$ is reachable from $x^0 = \delta_6^1$ with a probability of one.

Conclusion. We proposed a matrix-based modeling approach to PFAs under the framework of an STP of matrices. Meanwhile, with the new expression, we provided a sufficient and necessary condition for the reachability of PFAs, including the reachability and reachability with a probability of one. Finally, a simple example was used to validate the proposed result.

In the subsequent research on this topic, we will concentrate on the study of the controllability and stabilization of PFAs based on the results obtained here.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 61573199, 61573200).

References

1 Pantelic V, Postma S M, Lawford M. Probabilistic supervisory control of probabilistic discrete event systems. IEEE Trans Autom Control, 2009, 54: 2013–2018

2 Keroglou C, Hadjicostis C N. Verification of detectability in probabilistic finite automata. Automatica, 2017, 86: 192–198

3 Cheng D Z, Qi H S. A linear representation of dynamics of Boolean networks. IEEE Trans Autom Control, 2010, 55: 2251–2258

4 Zhao Y, Cheng D Z. On controllability and stabilizability of probabilistic Boolean control networks. Sci China Inf Sci, 2014, 57: 012202

5 Xu X R, Hong Y G. Matrix expression and reachability analysis of finite automata. J Control Theory Appl, 2012, 10: 210–215

6 Han X G, Chen Z Q, Liu Z X, et al. The detection and stabilisation of limit cycle for deterministic finite automata. Int J Control, 2018, 91: 874–886

7 Zhang Z P, Chen Z Q, Han X G, et al. On the static output feedback stabilization of deterministic finite automata based upon the approach of semi-tensor product of matrix. Kybernetika, 2018, 1: 41–60

8 Cassandras C, Lafortune S. Introduction to Discrete Event System. 2nd ed. New York: Springer Science and Business Media, 2008