Identification of Multi-Event Stochastic Fuzzy Discrete Event Systems

Hao Ying, *Fellow, IEEE,* and Feng Lin, *Fellow, IEEE*

***Abstract*—We recently introduced a novel category of fuzzy discrete event systems (FDES) termed stochastic fuzzy discrete event systems (SFDES), wherein multiple fuzzy automata occur randomly with different probabilities. We also developed two techniques for identifying event transition matrices in single-event SFDES employing the max-product fuzzy inference. One of them, named the Equation-Systems-Based Technique, focuses on single-event SFDES identification, where the fuzzy automaton of each FDES has only one event. Expanding on our research, this paper delves into multi-event SFDES identification, allowing each FDES to encompass a sequence of events. Upon activation of an FDES, all its events occur sequentially. Our mathematical proof first establishes the associativity of the max-product inference operation, leading to the introduction of a pivotal concept called an Equivalent Overall Event Transition Matrix for a consecutive event sequence. This concept establishes a theoretical framework for utilizing the Equation-Systems-Based Technique in a novel three-step method for identifying multi-event SFDES. The technique is employed in the first two steps to: (1) determine the number of fuzzy automata in a SFDES, and (2) calculate their occurrence frequencies. In the third step, multi-event transition matrices of the SFDES are learned by using stochastic-gradient-descent-based algorithms that we previously developed for multi-event FDES, provided the numbers of consecutive events for each fuzzy automaton within the SFDES are known. Theoretical analysis reveals the interconnections between the event transition matrices learned by the algorithms, the Equivalent Overall Event Transition Matrices derived from these matrices, and the target event transition matrices. To illustrate our findings, we present an informative example.**

***Index Terms*—fuzzy automaton, fuzzy discrete event systems, stochastic discrete event systems, system identification, supervised learning**

I. Introduction

The fuzzy discrete events systems (FDES) theory extends the discrete event systems (DES) theory that originated in the 1980s [1]. In DES, a system is composed of states, events, and state transitions that occur due to a sequence of events. This framework is useful for modeling practical systems that cannot be adequately described by differential or difference equations. These systems share a common characteristic - qualitative changes in system state are the results of occurrences of a sequence of events.

The supervisory control theory of DES [2], [3] introduced fundamental concepts such as controllability [4] and observability [5]. This led to the investigation of other important topics, including robust control, online control, decentralized control, limited-lookahead control, control under partial observation, and hierarchical control. DES theory has found practical applications in various fields such as air-traffic management, communications, smart grids, manufacturing, transportation, and computers.

The DES theory mandates binary state representation at two discrete levels, making it ill-suited for event-driven systems with ambiguous states characterized by continuous levels, as often encountered in healthcare. Describing an individual's health state typically involves subjectivity and ambiguity, lacking absolute certainty.

In order to effectively represent and process ambiguity and subjectivity in event-driven systems with continuous states, we expanded the DES theory by integrating it with fuzzy sets and fuzzy logic, resulting in the development of the FDES theory. This involved introduction of the notations of "fuzzy state" and "fuzzy event" [6]. A fuzzy automaton, similar to its crisp counterpart, mathematically modeled an FDES and was represented by

$$G = (\mathbf{Q}, \Sigma, \varphi, \mathbf{q_o}) \tag{1}$$

where $\mathbf{Q}$ was a vector representing N individual fuzzy states, $\mathbf{q_o}$ was an initial (i.e., pre-event) fuzzy state vector, Σ was a set of fuzzy events, each of which was characterized by an $N \times N$ event transition matrix, and $\varphi : \mathbf{Q} \times \Sigma \rightarrow \mathbf{Q}$ was an event transition mapping.

To represent ambiguity and subjectivity in DES, we extended the binary state to a fuzzy state by allowing membership of a state to be in the interval [0, 1], rather than 0 or 1. In addition, we allowed the elements of the event transition matrix to be in [0, 1], rather than 0 or 1, so that both the fuzzy state and fuzzy event could have partial memberships. We also generalized the transition mapping through fuzzy inference. These generalizations establish a DES and its associated automaton as a special case of an FDES with a fuzzy automaton. In [6], we also extended the parallel composition, optimal control, and observability of DES to FDES.

Other researchers have expanded the theoretical framework of FDES in various directions. These extensions include state-based control [7], state-feedback control [8], supervisory control [9], [10], [11], [12], decentralized control [13], [14], [15], online control [16], detectabilities [17], diagnosability [14], [18], [19], prognosis [20], [21], predictability [22], and opacity [23], [24]. Additionally, controllability of FDES has been studied [25]. Type-2 fuzzy sets have been used to expand the type-1-fuzzy-set-dominated FDES framework [26], and the

Hao Ying is with the Department of Electrical and Computer Engineering, Wayne State University, 5050 Anthony Wayne Dr., Detroit, MI 48202, USA. (corresponding author; e-mail: hao.ying@wayne.edu). Feng Lin is with the same department (e-mail: flin@wayne.edu). This work is supported in part by the National Science Foundation of USA under grant ECCS-2146615.

notions of generalized FDES [25] and semi-discrete events with fuzzy logic [27] have been proposed. The modeling of FDES based on a generalized linguistic variable has been explored [28]. In terms of applications, we studied decision-making using FDES and developed a new method for practical problems [29]. This method has been applied to optimal regimen selection in HIV/AIDS treatments with good retrospective clinical results [30], [31]. Non-medical applications of FDES include mobile robots [32], [33], [34] and air conditioning systems [35], [36]. A recent survey of the FDES literature is available [37].

The event transition matrix governs the state-to-state transitions within an FDES, thereby determining its crucial properties such as observability [1] and predictability [2]. Consequently, this matrix is of paramount importance. Manually crafting a matrix for a specific application can be an arduous and time-intensive endeavor, frequently necessitating the expertise of a domain specialist, such as a physician.

The challenge of manually producing an event transition matrix for an FDES motivated us to develop stochastic-gradient-descent-based online learning algorithms to learn the event transition matrix using pre- and post-event state vector pairs. We have developed four sets of learning algorithms to address different practical conditions that may arise for single-event FDES: (1) when both pre- and post-event individual states are known [39], [40], (2) when post-event individual states are available, but the corresponding pre-event individual states are not and they are linked to variables with known values [40], (3) when pre-event individual states are given, but the corresponding post-event individual states are unknown and they are linked to variables whose values are known [41], and (4) when both pre- and post-event individual states are unknown, but they are associated with variables with known values [42]. Moreover, we have developed learning algorithms to simultaneously learn multiple event transition matrices of an event sequence for multi-event FDES [43]. These learning algorithms are the only ones currently available in the literature for deterministic FDES.

As far as we are aware, most publications in the literature on FDES, except [44], [45], [46], [47], focus on the deterministic FDES, where the post-event state is computed from the pre-event state without any randomness. Modeling of the nondeterministic FDES is investigated in [44] while modeling and control of probabilistic FDES (PFDES for short) are addressed by us in [45].

We recently proposed a totally new class of probabilistic FDES and named them the stochastic FDES (SFDES for short) [46]. A SFDES consists of two or more fuzzy automata. Each of the fuzzy automata represents a sequence of fuzzy events with an occurrence probability. In the presence of a pre-event state, one does not know which of the fuzzy event sequences will occur and thus cannot know or determine the post-event state beforehand. At any moment, which event sequence will take place is random. While both the SFDES and the PFDES [45] deal with randomness in fuzzy automata, they differ fundamentally and are created for modeling different kinds of systems. A PFDES has only one fuzzy automaton while a SFDES has at least two fuzzy automata. A fuzzy automaton in a SFDES can characterize, for example, a patient or a group of similar patients. This feature offers a significant advantage over PFDES and makes SFDES more suitable for certain applications.

SFDES can be a useful modeling tool for practical problems, especially those in biomedicine. For example, SFDES can be utilize to model a disease treatment, which can be regarded as a fuzzy event or a fuzzy event sequence. It is common knowledge that treatment of many diseases, especially those involving cancer and heart, does not guarantee improvement, let alone cure. For these and many other diseases, there can be only three possible treatment outcome states (aka, post-event states) for any specific patient -"improved," "barely changed," and "worsen." Patient states like these are genuinely ambiguous (and can also be subjective), and hence are true fuzzy states. There can be overlaps between such fuzzy states as well. Disease treatment is such a complex process involving many known and unknown physiological mechanisms and factors in the body that even highly experienced specialist physicians cannot reliably predict treatment outcome state for any given patient. Which outcome state will be reached for any particular patient is fundamentally uncontrollable and thus appears to be at least somewhat random from the standpoint of the physician and patient.

For a population of patients suffering a same disease, an identical treatment can lead to different treatment outcomes for different patients. A treatment may be modeled by multiple fuzzy automata of a SFDES, one for each treatment outcome. Each fuzzy automaton represents one of poorly-understood physiological processes in patient body that leads to a treatment outcome state (e.g., "serious condition" or "stable condition"). Pre- and post-treatment state pairs for each patient in the population may be attained using his/her medical and personal information. The state pairs can then be used to potentially find out: (1) number of fuzzy automata in the SFDES, (2) their event transition matrices, and (3) their occurrence probabilities. Successful completion of the model identification can result in a clinically useful SFDES model.

In [46], we not only proposed the notion of a SFDES, but developed a technique called the Prerequired-Pre-Event-State-Based Technique to identify single-event SFDES. SFDES can be categorized into single-event type and multi-event type. The latter type allows two or more consecutive events (i.e., an event sequence) to occur for any fuzzy automaton in a SFDES while the former type only permits one event for each automaton. Regardless of the types, events are required to be independent one another. From an application standpoint, single-event SFDES can be adequate for modeling real-world problems (e.g., disease treatment).

For a single-event SFDES with N individual states, the technique first creates the following N pre-event state vectors: $\mathbf{\Theta_1} = [\mathbf{1}\ \mathbf{0}\ \ldots\ \mathbf{0}], \mathbf{\Theta_2} = [\mathbf{0}\ \mathbf{1}\ \ldots\ \mathbf{0}], \ldots, \mathbf{\Theta_N} = [\mathbf{0}\ \mathbf{0}\ \ldots\ \mathbf{1}]$, each of which is $1 \times N$. It then feeds them one at a time to the SFDES being identified to obtain the post-event state vectors. Completing feeding of all the N vectors is regarded as one round of feeding. The technique performs rounds of feeding. The post-event state vector corresponding to $\mathbf{\Theta_h}$ is actually the h-th row of one of the event transition matrices of the

SFDES, although we do not know which matrix owns the row in the vector-feeding stage. After the feeding stage, the technique has a mechanism to correctly assign all these rows to the matrices as well as to estimate occurrence probabilities of the fuzzy automata. The technique is suitable for single-event SFDES using either the max-product or max-min fuzzy inference method,. Importantly, the technique is easy to use as it achieves the identification goals without any adjustable parameter or hyperparameter. Two necessary and sufficient conditions and two sufficient conditions are established for the identification of single-event SFDES with different settings.

This Prerequired-Pre-Event-State-Based Technique is developed for application scenario where the above-mentioned particular N pre-event state vectors can be used. There are applications for which the technique is proper. Nevertheless, the technique may not be suitable for other applications because the particular N pre-event state vectors $\mathbf{\Theta_h}$ that it requires are not permissible. This motivated us to develop another identification technique for single-event SFDES, which is called the Equation-Systems-Based Technique [47]. The name reflects the fact that event transition matrices are identified through solving sets of equations relating pre-event state vector to post-event state vector. The important difference between the two techniques is that now we do not require or set pre-event state vectors to any particular levels. The second technique is capable of using whatever pre-event state vectors available to identify a single-event SFDES model that uses the max-product fuzzy inference (it does not work for the max-min fuzzy inference).

These two methods were devised exclusively for identifying single-event SFDES. Consequently, they are not directly applicable to the identification of multi-event SFDES, which are the focal systems of interest in this paper.

Can the Equation-Systems-Based Technique contribute to our endeavor to identify multi-event SFDES using max-product fuzzy inference? Indeed, it can. With this objective in mind, we shall first mathematically establish that the max-product operation is associative. This property leads us to introduce the notion of an Equivalent Overall Event Transition Matrix for a sequence of events in a multi-event FDES. This new concept lays a foundation for developing an innovative three-step method in Subsection C of Section V below for identifying multi-event SFDES.

The first two steps of the three-step process involve: (1) determining the number of fuzzy automata in a SFDES comprising R multi-event FDES, where the value of R is unknown, and (2) calculating their occurrence frequencies. Utilizing the notation of the Equivalent Overall Event Transition Matrix, we can treat the multi-event SFDES as R single-event SFDES from the standpoint of the pre-event state of the first event and the post-event state of the last event in each of the R multi-event FDES. Leveraging this approach, the Equation-Systems-Based Technique is employed to achieve both objectives.

The third step of the new method involves acquiring the transition matrices for each of the R multi-event FDES, which collectively constitute the multi-event SFDES. This is accomplished by employing the aforementioned stochastic gradient descent-based algorithms previously developed in our work [43], leveraging the known counts of events within each FDES. The algorithms were developed for multi-event FDES, and consequently, they are incapable of identifying multi-event SFDES. However, through the notation of the Equivalent Overall Event Transition Matrix, the challenge of identifying a multi-event SFDES comprising R multi-event FDES can be effectively reframed as a sequential identification process of the R multi-event FDES individually. This transformation renders the algorithms entirely applicable.

Finally, we theoretically analyze the event transition matrices learned by the gradient descent-based algorithms, elucidating the interconnections between these matrices, their Equivalent Overall Event Transition Matrix, and the Equivalent Overall Event Transition Matrix constituted by the target event transition matrices. This foundational comprehension of the algorithms was previously unattainable during their development due to the absence of the Equivalent Overall Event Transition Matrix notation.

In the next section, we will introduce SFDES, which will be followed by a problem statement in Section III. In Section IV, we will prove the max-product operation to be associative and establish the concept of the Equivalent Overall Event Transition Matrix for a sequence of consecutive events. In Section V, we will utilize this concept to develop the novel three-step method for identifying multi-event SFDES. Simulation results will be presented in Section VI to demonstrate the method. Conclusions will be drawn in the last section.

II. Introduction to Stochastic Fuzzy Discrete Event Systems

A SFDES is composed of two or more FDES, each of which is represented by an fuzzy automaton. The fuzzy automata take place randomly one at a time based on their occurrence probabilities. If two or more events occur consecutively for at least one of the fuzzy automata, the SFDES is said to be a multi-event SFDES (in contrast, a SFDES is of the single-event type if each of the fuzzy automata has only one event).

Like its crisp counterpart, a fuzzy automaton [6] is mathematically represented by

$$G = (\mathbf{Q}, \Sigma, \varphi, \mathbf{q_o})$$

where $\mathbf{Q}$ is a state vector containing N individual fuzzy states. More specifically, $\mathbf{Q} = [Q_1, Q_2, \ldots, Q_N]$. Membership value of Q_i, denoted by S_i, is in [0, 1], and the vector $\mathbf{\Theta} = [S_1, S_2, \ldots, S_N]$ represents the overall state of the fuzzy automaton, which is referred as the system state. $\mathbf{q_o}$ is the initial (fuzzy) state. Σ is a set of Ω fuzzy events, each of which is denoted by $\tilde{\mathbf{\Psi}}_k$ and is represented by an $N \times N$ event transition matrix:

$$\tilde{\mathbf{\Psi}}_k = \begin{bmatrix} a_{11k} & a_{12k} & \ldots & a_{1Nk} \\ a_{21k} & a_{22k} & \ldots & a_{2Nk} \\ \ldots & & & \\ a_{N1k} & a_{N2k} & \ldots & a_{NNk} \end{bmatrix}$$

where $1 \leq k \leq \Omega$ and all the matrix elements fall in [0, 1]. In this study, the elements in a row (or a column) of $\tilde{\mathbf{\Psi}}_k$ are not required to be summed to 1. $\varphi : \mathbf{Q} \times \Sigma \rightarrow \mathbf{Q}$ is an event transition mapping. After an event takes place, the system will transfer from a pre-event state (first such state is the initial state), through its event transition matrix, to a post-event system state, which can be computed by using a fuzzy inference operator.

To illustrate, assume pre-event state is $\mathbf{\Theta}_0 = [S_{10} \quad S_{20} \quad ... \quad S_{N0}]$ and event $\tilde{\mathbf{\Psi}}_k$ occurs. Then, post-event state of the system, denoted as $\mathbf{\Theta}_1 = [S_{11} \quad S_{21} \quad ... \quad S_{N1}]$, is:

$$\begin{aligned}\mathbf{\Theta}_1 &= \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_k \\ &= [S_{10} \quad S_{20} \quad ... \quad S_{N0}] \circ \begin{bmatrix} a_{11k} & a_{12k} & \dots & a_{1Nk} \\ a_{21k} & a_{22k} & \dots & a_{2Nk} \\ \dots & & & \\ a_{N1k} & a_{N2k} & \dots & a_{NNk} \end{bmatrix} \\ &= [S_{11} \quad S_{21} \quad ... \quad S_{N1}].\end{aligned}$$

The symbol $\circ$ denotes fuzzy inference operation, which in this paper is the max-product operation. Thus,

$$S_{j1} = max(S_{10}a_{1jk}, S_{20}a_{2jk}, ..., S_{N0}a_{Njk}). \quad (2)$$

For notional convenience, we define $\varphi(\mathbf{\Theta}_0, \tilde{\mathbf{\Psi}}_k) = \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_k$.

Each fuzzy automaton in a SFDES can have one sequence of consecutive events. If M events $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, ..., $\tilde{\mathbf{\Psi}}_M$ take place consecutively one after another, the system state after occurrence of the last event $\tilde{\mathbf{\Psi}}_M$, denoted as $\mathbf{\Theta}_M = [S_{1M} \quad S_{2M} \quad ... \quad S_{NM}]$, can be computed:

$$\mathbf{\Theta}_M = \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \cdots \circ \tilde{\mathbf{\Psi}}_M$$

and alternatively, it can be expressed as

$$\mathbf{\Theta}_M = \varphi(\mathbf{\Theta}_0, \tilde{\mathbf{\Psi}}_1 \tilde{\mathbf{\Psi}}_2 \dots \tilde{\mathbf{\Psi}}_M). \quad (3)$$

Since a SFDES has more than one fuzzy automaton, when necessary, we will use a subscription or superscription of h to index the h-th FDES or fuzzy automaton from now on (e.g., $\tilde{\mathbf{\Psi}}_k^h$ represents the k-th event of the h-th FDES/fuzzy automaton).

III. Problem Statement

Without loss of generality, assume a multi-event SFDES contains R (> 1) FDES with N individual states. Also, suppose that the value of R is unknown, but the value of N is available (in a real-world application, one usually knows the number of individual states in the system being modeled). The h-th FDES, represented by fuzzy automaton G_h with occurrence probability p_h, $h = 1, 2, \dots, R$, has M_h consecutive events represented by event transition matrices $\tilde{\mathbf{\Psi}}_1^h$, $\tilde{\mathbf{\Psi}}_2^h$, ..., $\tilde{\mathbf{\Psi}}_{M_h}^h$. It is required that at least one of the fuzzy automata has two or more consecutive events (otherwise, it is a single-event SFDES). When the h-th FDES acts on a pre-event state, its M_h consecutive events will take place one after another with the order $\tilde{\mathbf{\Psi}}_1^h$, $\tilde{\mathbf{\Psi}}_2^h$, ..., $\tilde{\mathbf{\Psi}}_{M_h}^h$. Neither the transition matrices nor the occurrence probabilities are known; they need to be identified. M_h is assumed to be known for all h.

To identify the SFDES, we feed a series of H random pre-event states, one at a time, to it. They will be acted upon randomly by the R FDES according to their occurrence probabilities. One pre-event state will be taken by one FDES only. We record every pre-event state along with the corresponding post-event state exhibited by the SFDES. The occurrence of any event in any of the R event sequences is unobservable, and only the post-event state of the last event $\tilde{\mathbf{\Psi}}_{M_h}^h$ is available. That is, only the outcome of $\varphi(\mathbf{\Theta}_0, \tilde{\mathbf{\Psi}}_1^h \tilde{\mathbf{\Psi}}_2^h \dots \tilde{\mathbf{\Psi}}_{M_h}^h)$, denoted as $\mathbf{\Theta}_{M_h,h}$, is available. The pre- and post-event states are paired and are recorded as such. They are represented by $(\mathbf{\Theta}_0^j, \mathbf{\Theta}_{M*}^j)$, $j = 1, 2, \dots, H$. Here, the subscription $M*$ means it can be the post-event state of the last event of any one of the R consecutive event sequences.

Data availability plays a critical role in any system identification and machine learning. This study is no exception. We assume ample state pairs are available in this study. That is, H can be as large as it needs to be.

In summary, the following assumptions are made regarding the unknown aspects of a SFDES to be identified: (1) the value of R, (2) all event transition matrices, and (3) the occurrence probability of any one of the R FDES within the SFDES. Also unknown are (1) which FDES acts on which pre-event state, and (2) post-event states of all the events in any event sequence except the last event. Conversely, N is presumed to be known, as well as M_h for $h = 1, 2, \dots, R$. Additionally, H is assumed to be sufficiently large. It is worth noting that these assumptions render the multi-event SFDES largely unknown to the modeler initially, thereby underscoring the relevance of the new technique developed in this paper to real-world problems, consequently enhancing its practical utility.

With these assumptions, we ask the following questions: (1) how to determine the number of fuzzy automata in the SFDES (i.e., the value of R), (2) how to determine fuzzy automata's occurrence frequencies as a way to estimate their occurrence probabilities, and (3) how to identify all the event transition matrices of the R fuzzy automata?

We develop an innovative three-step SFDES identification technique below that will answer all these questions. We first introduce a new concept: the "equivalent overall event transition matrix" as we need it to represent a sequence of consecutive events in a collective manner. Development of the three-step technique relies on this notion.

IV. Equivalent Overall Event Transition Matrix for Collectively Representing An Event Sequence

In this section, we will first prove the max-product operation to be associative and will then use this property to introduce the notion of the Equivalent Overall Event Transition Matrix. As will be shown in the next section, this new concept can be utilized to effectively transform the problem of identifying a multi-event SFDES to the problem of identifying a single-event SFDES, paving the way to answer the aforementioned three questions. To this end, we have

Lemma 1: The max-product operation employed in multi-event SFDES or multi-event FDES is associative. In other words, for any system state $\mathbf{\Theta}_0$ and any fuzzy event transition matrices $\tilde{\mathbf{\Psi}}_1$ and $\tilde{\mathbf{\Psi}}_2$, the following equations are valid:

$$\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 = (\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1) \circ \tilde{\mathbf{\Psi}}_2 = \mathbf{\Theta}_0 \circ (\tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2)$$

The proof of this lemma is provided in Appendix.

Extending Lemma 1 to cover three or more events, we have **Theorem 1**: In multi-event SFDES or multi-event FDES, given any system state $\mathbf{\Theta_0}$ and any sequence of fuzzy events represented by event transition matrices $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, $\ldots$, $\tilde{\mathbf{\Psi}}_M$, where $M > 1$, the following holds true:

$$\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \circ \cdots \circ \tilde{\mathbf{\Psi}}_M = \mathbf{\Theta}_0 \circ (\tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \circ \cdots \circ \tilde{\mathbf{\Psi}}_M). \tag{4}$$

Proof: We first introduce the following notation:

$$\tilde{\mathbf{\Psi}}_{1k} := \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \circ \cdots \circ \tilde{\mathbf{\Psi}}_k.$$

Because of Lemma 1, the left side of (4) can be written as

$$\begin{aligned}
&\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \circ \cdots \circ \tilde{\mathbf{\Psi}}_M \\
&= (\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2) \circ \cdots \circ \tilde{\mathbf{\Psi}}_M \\
&= \mathbf{\Theta}_0 \circ (\tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2) \circ \cdots \circ \tilde{\mathbf{\Psi}}_M \\
&= \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_{12} \circ \tilde{\mathbf{\Psi}}_3 \circ \cdots \circ \tilde{\mathbf{\Psi}}_M \\
&= (\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_{12} \circ \tilde{\mathbf{\Psi}}_3) \circ \cdots \circ \tilde{\mathbf{\Psi}}_M \\
&= \mathbf{\Theta}_0 \circ (\tilde{\mathbf{\Psi}}_{12} \circ \tilde{\mathbf{\Psi}}_3) \circ \cdots \circ \tilde{\mathbf{\Psi}}_M \\
&= \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_{13} \circ \tilde{\mathbf{\Psi}}_4 \circ \cdots \circ \tilde{\mathbf{\Psi}}_M \\
&\quad \ldots \\
&= \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_{1(M-1)} \circ \tilde{\mathbf{\Psi}}_M \\
&= \mathbf{\Theta}_0 \circ (\tilde{\mathbf{\Psi}}_{1(M-1)} \circ \tilde{\mathbf{\Psi}}_M) \\
&= \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_{1M}.
\end{aligned}$$

Note that $\tilde{\mathbf{\Psi}}_{1M} = \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \circ \cdots \circ \tilde{\mathbf{\Psi}}_M$. Thus, the left and right sides of equation (4) are equal. **QED**

Corollary 1: In multi-event SFDES or multi-event FDES, given any system state $\mathbf{\Theta}_0$ and any sequence of fuzzy events represented by event transition matrices $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, $\ldots$, $\tilde{\mathbf{\Psi}}_M$, one can always find such an event transition matrix $\tilde{\mathbf{\Psi}}$:

$$\tilde{\mathbf{\Psi}} = \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \circ \cdots \circ \tilde{\mathbf{\Psi}}_M$$

that satisfies the following condition:

$$\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}} = \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \cdots \circ \tilde{\mathbf{\Psi}}_M.$$

Note that $\tilde{\mathbf{\Psi}} = \tilde{\mathbf{\Psi}}_{1M}$ and $\tilde{\mathbf{\Psi}}$ has the same dimensionality as the other matrices.

Example 1: Assume there is a multi-event SFDES and one of the fuzzy automata has the following three event transition matrices:

$$\tilde{\mathbf{\Psi}}_1 = \begin{bmatrix} 1.0 & 0.75 \\ 0.32 & 0.11 \end{bmatrix}, \tilde{\mathbf{\Psi}}_2 = \begin{bmatrix} 0.68 & 0.44 \\ 0.95 & 0.57 \end{bmatrix}, \tilde{\mathbf{\Psi}}_3 = \begin{bmatrix} 0.83 & 0.07 \\ 0.46 & 0.69 \end{bmatrix}$$

and $\mathbf{\Theta}_0 = [0.21 \quad 0.65]$. Use the given information to show the correctness of Corollary 1 and Theorem 1.

Using the notations in Theorem 1 and noting that $M = 3$, we get

$$\tilde{\mathbf{\Psi}}_{13} = \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \circ \tilde{\mathbf{\Psi}}_3 = \begin{bmatrix} 0.5914 & 0.3036 \\ 0.1806 & 0.097 \end{bmatrix}.$$

It can be easily verified that $\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 \circ \tilde{\mathbf{\Psi}}_3 = \mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_{13} = [0.1242 \quad 0.0638]$, which is expected per Corollary 1 or Theorem 1.

It is important to point out that $\tilde{\mathbf{\Psi}}$ is determined by $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, $\ldots$, and $\tilde{\mathbf{\Psi}}_M$ only and is unrelated to the pre-event state $\mathbf{\Theta}_0$. That means it is an intrinsic property of the events and the event transition matrices involved.

Definition: The event transition matrix $\tilde{\mathbf{\Psi}}$ in Corollary 1 is defined as the Equivalent Overall Event Transition Matrix of $\tilde{\mathbf{\Psi}}_1, \tilde{\mathbf{\Psi}}_2, \ldots, \tilde{\mathbf{\Psi}}_M$ because it provides a collective characterization of the event sequence.

As a concrete instance, in Example 1,

$$\tilde{\mathbf{\Psi}}_{13} = \begin{bmatrix} 0.5914 & 0.3036 \\ 0.1806 & 0.097 \end{bmatrix}$$

is the Equivalent Overall Event Transition Matrix of $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, and $\tilde{\mathbf{\Psi}}_3$.

With the nature of $\tilde{\mathbf{\Psi}}_{1M}$ in Theorem 1 being revealed, the theorem can be better understood. Basically it states that given a pre-event state, the post-event state of a series of M consecutive events can be calculated in two different ways and the result will be the same: (1) apply the M event transition matrices one at a time to the post-event state of the previous event, starting with the first transition matrix being applied to the pre-event state, and (2) obtain the Equivalent Overall Event Transition Matrix of the M event transition matrices first and then apply it to the pre-event state. Example 1 shows this point as a concrete example.

For the R fuzzy automata formulated in Section III, there are R Equivalent Overall Event Transition Matrices, designated as $\tilde{\mathbf{\Psi}}^h_{1M_h}, h = 1, 2, \ldots, R$, one for each of the fuzzy automata.

In light of Corollary 1, the following result is obvious.

Corollary 2: For a multi-event FDES with a sequence of M events, where the value of M, pre-event state of the first event, and post-event state of the last event are available, from the perspective of these pre- and post-event states, the M events can be collectively viewed as a single event whose event transition matrix is the Equivalent Overall Event Transition Matrix of the M event transition matrices.

Related to the multi-event fuzzy automaton in Example 1, assuming the only information available is: (1) there are three events, and (2) the pre-event state of event $\tilde{\mathbf{\Psi}}_1$ and post-event state of event $\tilde{\mathbf{\Psi}}_3$, then $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, and $\tilde{\mathbf{\Psi}}_3$ can be collectively treated as a single-event automaton whose event transition matrix is $\tilde{\mathbf{\Psi}}_{13}$.

V. Identification of Multi-Event Stochastic Fuzzy Discrete Event Systems

The notion of the Equivalent Overall Event Transition Matrix of an event sequence serves as a crucial link that bridges the significant gap between identification of the multi-event SFDES and identification of the single-event SFDES that we previously achieved already [46][47]. Owing to Corollary 2, one of the two identification techniques that we developed for the single-event SFDES identification, namely the Equation-Systems-Based Technique, becomes usable in the first step of the multi-event SFDES identification. It also makes the stochastic gradient descent algorithms that we developed previously for learning multi-event transition matrices in FDES [43] applicable to the identification of the event transition matrices in each of the R fuzzy automata in the multi-event SFDES.

Below, we will first provide a brief description of these pre-existing technique and algorithms, and will then show how to apply them, in combination with the notion of the Equivalent Overall Event Transition Matrix, to the identification of the multi-event SFDES.

A. Pre-existing Equation-Systems-Based Technique for Identifying Event Transition Matrices in Single-Event SFDES [47]

Our pre-existing Equation-Systems-Based Technique is capable of using pairs of pre-event state and post-event state generated by a single-event SFDES that uses the max-product fuzzy inference to: (1) determine how many fuzzy automata are in the single-event SFDES, (2) calculate the occurrence frequency of each fuzzy automaton, and (3) identify the event transition matrix of each fuzzy automaton. We named the technique the Equation-Systems-Based Technique because of the principle that the technique was based on - event transition matrices were identified through solving sets of equations relating the pre-event state to the post-event state. We proved theoretically that attaining these three goals was guaranteed if the number of pre-event-state-post-event-state pairs was large enough. Note that this is not a statistical approach or machine learning approach. The technique does not have an objective function to minimize. Element values of an event transition matrix are not estimated or learned through an iterative updating process. Rather, they are calculated from equations and the resulting element values are exact without any deviation to the underlying true values. There are a total of $N \times H$ systems of equations involved in the identification of a single-event SFDES having N individual states when H sample state pairs are available. There are no issues such as local minimums of an objective function, and there is no hyperparameter to set, making the technique easy to use.

The technique operates in two identification phases. In the first phase, values of all elements of event transition matrices are determined through equation sets without knowing which element belongs to which matrix. In the second phase, the elements are assigned to the event transition matrices through a mechanism called the Linked-Elements-Based Assignment Method. Moreover, in the process, the technique determines retrospectively which pair of pre-event state and post-event state was acted upon by which fuzzy automaton. It then uses this information to calculate the occurrence frequency of each and every one of the fuzzy automata (i.e., percentage of the pairs acted upon by an fuzzy automaton in the total number of pairs).

The technique works no matter how many fuzzy automata are in a SFDES, how larger their event transition matrices are, and how different or similar their occurrence probabilities may be.

B. Pre-existing Supervised Learning Algorithms for Learning Multi-Event Transition Matrices in FDES [43]

Our work in [43] is devoted to identification of multi-event FDES (not SFDES) that adopts the max-product fuzzy inference. We derived a set of stochastic-gradient-descent-based formulas capable of learning event transition matrices of consecutive events (e.g., $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, and $\tilde{\mathbf{\Psi}}_3$ of the multi-event automaton in Example 1). Number of events in an event sequence needs to be provided first so that the algorithms knows how many transition matrices need to be learned. Availability of sufficient pre-event-state-post-event-state pairs is a requirement for the learning to be complete and successful.

To simplify matter, learning rates for different event transition matrices of an automaton are assumed to be the same. Thus, there is only one hyperparameter to set for the algorithms, which is the learning rate λ for updating element values of the event transition matrices being learned. Optimal value of the hyperparameter needs to be found experimentally in a trial-and-error fashion. Our experience with simulations indicates it is generally not a difficult process.

We devised two variations of gradient descent learning algorithms, differing solely in the min() operator used within the max-product fuzzy inference. One variant substitutes the traditional "hard" max() with a "soft" max(). In the other variant, we developed a novel adaptive learning rate scheme, resulting in accelerated and more efficient learning. The degree of adaptation is adjustable through a hyperparameter τ. This latter variant will be utilized in a simulation example later.

It would take a large space to provide a detailed description of the learning algorithms as well as the Equation-Systems-Based Technique. Space limit precludes such a possibility. The reader is advised to find full information through the papers [47][43].

C. Identifying Multi-Event SFDES via a Novel Three-Step Method

Identification of a multi-event SFDES has never been attempted in the literature before. The three-step identification method that we develop in this paper for this goal represents one of the main contributions of this study.

In the first step, we apply the Equcation-Systems-Based Technique to the H state vector pairs, ($\mathbf{\Theta}_0^j$, $\mathbf{\Theta}_{M*}^j$), one at a time. The technique will: (1) yield the number of fuzzy automata in the SFDES (i.e., the value of R), and (2) identify R event transition matrices. The R matrices identified are actually the Equivalent Overall Event Transition Matrices of the R sequences of consecutive events. This is because in light of Corollary 2, these Equivalent Overall Event Transition Matrices collectively represent a **single-event** SFDES from the perspective of model input (i.e., $\mathbf{\Theta}_0^j$) and model output (i.e., $\mathbf{\Theta}_{M*}^j$). So, at this point, the number of fuzzy automata (i.e., the value of R) has been determined, which is the first of the three questions raised in Section III for multi-event SFDES identification.

The second step focuses on the second question, that is, how to determine the occurrence frequencies of the R fuzzy automata as a way to estimate their occurrence probabilities. Note that in the process of identifying the R Equivalent Overall Event Transition Matrices in the first step, the Equation-Systems-Based Technique has already sorted out, retrospectively and one by one, which of the H state vector pairs was acted upon by which fuzzy automaton. This information enables one to easily calculate the occurrence frequency for each of the R fuzzy automata.

We comment that the notion of the Equivalent Overall Event Transition Matrix is of fundamental importance as it makes these two identification steps possible.

The third step is to identify all the event transition matrices of the R fuzzy automata, which is the third question in Section III. Recall that there are M_h matrices to be identified for the h-th fuzzy automaton, and the value of M_h is assumed to be known. Because each fuzzy automaton is a multi-event FDES, our pre-existing stochastic-gradient-descent-based learning algorithms mentioned above can be utilized to accomplish the task.

Other than the value of M_h, we also need to provide the learning algorithms with the ($\mathbf{\Theta}_0^j$, $\mathbf{\Theta}_{M*}^j$) pairs that belong to the h-th fuzzy automata. In this regard, recall in the first step above, the Equation-Systems-Based Technique sorts out, retrospectively and one by one, which pair of ($\mathbf{\Theta}_0^j$, $\mathbf{\Theta}_{M*}^j$) was acted upon by which fuzzy automaton. That means we can do the same here also and use the resulting pairs to build a set of ($\mathbf{\Theta}_0^j$, $\mathbf{\Theta}_{M*}^j$) pairs for each of the R fuzzy automata. Specifically, suppose the number of ($\mathbf{\Theta}_0^j$, $\mathbf{\Theta}_{M*}^j$) pairs tied to the h-th fuzzy automaton is L_h, where $L_1+L_2+\cdots+L_R = H$. Those pairs are denoted as $(\mathbf{\Theta}_0^k, \mathbf{\Theta}_{M_h}^k)$, $k = 1, 2, \ldots, L_h$.

The learning algorithms will learn the transition event matrices for one fuzzy automaton a time. With the value of M_h and $(\mathbf{\Theta}_0^k, \mathbf{\Theta}_{M_h}^k)$, where $\mathbf{\Theta}_0^k = [S_{10}^k \quad S_{20}^k \quad \ldots \quad S_{N0}^k]$ and $\mathbf{\Theta}_{M_h}^k = [S_{1M_h}^k \quad S_{2M_h}^k \quad \ldots \quad S_{NM_h}^k]$, the learning algorithms learn the M_h underlying/true event transition matrices of the h-th fuzzy automaton, which are designated as $\tilde{\mathbf{\Psi}}_1^h$, $\tilde{\mathbf{\Psi}}_2^h$, $\ldots$, and $\tilde{\mathbf{\Psi}}_{M_h}^h$. Denote the corresponding actual transition matrices being iteratively modified as $\hat{\mathbf{\Psi}}_1^h$, $\hat{\mathbf{\Psi}}_2^h$, $\ldots$, and $\hat{\mathbf{\Psi}}_{M_h}^h$ with the Equivalent Overall Event Transition Matrix for the h-th fuzzy automaton being $\hat{\mathbf{\Psi}}_{1M_h}^h$. Also, denote the actual post-event state of the last transition matrix $\hat{\mathbf{\Psi}}_{M_h}$ in response to $\mathbf{\Theta}_0^k$ as $\hat{\mathbf{\Theta}}_{M_h}^k$, where $\hat{\mathbf{\Theta}}_{M_h}^k = [\hat{S}_{1M_h}^k \quad \hat{S}_{2M_h}^k \quad \ldots \quad \hat{S}_{NM_h}^k]$. The learning algorithms update the elements of the transition matrices to minimize the following objective function:

$$E_h := \frac{1}{2}\sum_{j=1}^{N}(\hat{S}_{jM_h}^k - S_{jM_h}^k)^2. \quad (5)$$

Learning is halted when the normalized sum of errors

$$E_h^{avg} := \frac{\sum_{k=1}^{L_h}\sum_{j=1}^{N}|\hat{S}_{jM_h}^k - S_{jM_h}^k|}{L_h N} \quad (6)$$

is less than a user-specified stopping threshold ϵ, which is the same regardless of h.

When the transition matrices of all the R fuzzy automata have been successfully learned, the third and final step of the identification process is completed.

The three-step identification method for multi-event SFDES utilizes the techniques that we have previously published and thus has solid theoretical base. Theoretically speaking, the method works no matter how small or large a multi-event SFDES is. Computing time for executing the method primarily depends on the magnitudes of N, R, M_h, and H. Our simulation experience suggests that a relatively new typical personal computer could complete the entire identification process for cases involving moderate magnitudes of N, R, M_h, and H within minutes. For more complex multi-event SFDES models with larger magnitudes of N, R, M_h, and H, the execution time is expected to be longer, but not prohibitively long. With the rapid advancement of computer hardware, computing time is not considered a bottleneck for the new method.

It is worth to mention that under certain conditions, the above assumption that the value of M_h is known and is provided to the learning algorithms may be circumvented. One such condition is that in real-world applications, values of certain elements of some event transition matrices have specific meanings or interpretations and are hence expected to be in certain ranges. The modeler may be able to use his domain expertise to judge whether the matrices learned for his particular application are sensible or not. In cases like this, even if M_h is unknown, the modeler can run the learning algorithms by assuming different values for M_h and then inspects the matrices learned to decide which M_h value is correct.

This identification method employs paired pre- and post-event state vectors of event sequences to facilitate the three-step identification process. Applicability of the method extends to practical systems characterized by event-driven states, provided the aforementioned assumptions hold true. The quality of the resulting multi-event SFDES model, gauged by disparities between actual and computed post-event state, hinges upon the fitness of the actual system to the SFDES framework, alongside its max-product inference. Obviously, if any of the assumptions proves false, the method's applicability diminishes.

D. *Examining Event Transition Matrices Learned in the Third Identification Step Using the Concept of the Equivalent Overall Event Transition Matrix*

Convergence of the learning of the event transition matrices in the third step depends on such factors as training samples, values of the learning rates, and initial values of the matrices to be learned. In the context of this study, convergence means progressive reduction of E_h^{avg}, for all h, through training, which will eventually end up with an acceptable small final value if the settings are proper. When this happens, we say the learning has converged.

Whatever training samples available will be used for training. The modeler has little control over them. Random matrices whose element values are in [0, 1] are used as the initial event transition matrices. The learning rates are empirically determined through trial-and-error, which is common in the machine learning field.

Like most machine learning techniques, establishing theoretical conditions for learning convergence is technically very challenging. Nevertheless, to a certain extent, convergence can be investigated through simulations and that is what we did when developing the algorithms [43]. As an example, one of the simulated FDES in [43] had three consecutive events with N being 4. Starting with random initial values for the matrices, the multi-event FDES was trained using 200 sample pairs. Through training, E_1^{avg} (the subscription is 1 because there is only one fuzzy automaton) reduced quickly with a final reading of mere 1.04402×10^{-10} when training was completed.

An important question to ask is whether a smaller value of E_h^{avg} always indicates smaller element-wise differences between the learned matrices $\hat{\mathbf{\Psi}}_1^h$, $\hat{\mathbf{\Psi}}_2^h$, $\ldots$, and $\hat{\mathbf{\Psi}}_{M_h}^h$ and their respective target matrices $\tilde{\mathbf{\Psi}}_1^h$, $\tilde{\mathbf{\Psi}}_2^h$, $\ldots$, and $\tilde{\mathbf{\Psi}}_{M_h}^h$? Unfortunately, the answer is no. This is because the post-event states of $\hat{\mathbf{\Psi}}_1^h$, $\hat{\mathbf{\Psi}}_2^h$, $\ldots$, and $\hat{\mathbf{\Psi}}_{M_h-1}^h$ are assumed to be unknown and only $\mathbf{\Theta}_0^k$ and $\mathbf{\Theta}_{M_h}^k$ are assumed to be available to the learning algorithms. Such uncompromising yet realistic assumptions make it impossible to directly force the elements of the matrices being learned, namely $\hat{\mathbf{\Psi}}_1^h$, $\hat{\mathbf{\Psi}}_2^h$, $\ldots$, and $\hat{\mathbf{\Psi}}_{M_h}^h$, to approximate the corresponding elements of the respective underlying event transition matrices. This is an inevitable consequence of the unavailability of the intermediate post-event states.

Example 2: Continue on Example 1. If the learning algorithms are given only the following information: (1) there are three events (i.e., $M = 3$), and (2) the pre-state of event $\tilde{\mathbf{\Psi}}_1$ and post-event state of event $\tilde{\mathbf{\Psi}}_3$, then the three event transition matrices learned may not be the same as $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, and $\tilde{\mathbf{\Psi}}_3$. This is understandable as post-event states of $\tilde{\mathbf{\Psi}}_1$ and $\tilde{\mathbf{\Psi}}_2$ are unavailable, making learning more difficult.

An important follow-up question is this: how to interpret/understand the learned event transition matrices since they may be different from their target transition matrices? We were aware of this issue and chose not to raise it in [43] as we did not have a reasonable answer then.

After the publication of [43], our research persisted in addressing the issue, attempting strategies to reconcile disparities between the learned matrices and the target matrices, all to no avail. Through simulations, we observed that despite notable discrepancies in the learned matrices due to diverse initial conditions and learning rates, the matrix resulting from the learned matrices through the max-product inference bore resemblance to the matrix computed by the target matrices using the same inference. This revelation led us to realize that the multi-event learning was, in fact, a form of single-event learning when viewed from the perspective of input-output state pairs. This realization prompted us to contemplate the concept of an Equivalent Overall Event Transition Matrix. To ensure mathematical rigor in this notation, we embarked on establishing the associativity of the max-product fuzzy inference, a task which we successfully accomplished.

The notion of the Equivalent Overall Event Transition Matrix and the related Corollary 2 developed above enable us to re-examine this issue from a fresh new perspective. We can now conclude that the learned matrices are optimized through training in such a way that the Equivalent Overall Event Transition Matrix that they constitute will approximate that constituted by the underlying target transition matrices.

We now formally prove this finding in a general setting.

Assume there are M consecutive events $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, $\ldots$, $\tilde{\mathbf{\Psi}}_M$ of a target fuzzy automaton in a multi-event SFDES (or FDES) to be learned and their Equivalent Overall Event Transition Matrix is $\tilde{\mathbf{\Psi}}_{1M} = [b_{ij}]$. The value of M is known. Sample pairs $(\mathbf{\Theta}_0^k, \mathbf{\Theta}_M^k)$, $k = 1, 2, \ldots, H$, are available for training, where $\mathbf{\Theta}_M^k = [S_{1M}^k \quad S_{2M}^k \quad \ldots \quad S_{NM}^k]$. The corresponding target events are denoted as $\hat{\mathbf{\Psi}}_1$, $\hat{\mathbf{\Psi}}_2$, $\ldots$, $\hat{\mathbf{\Psi}}_M$ with the Equivalent Overall Event Transition Matrix being $\hat{\mathbf{\Psi}}_{1M} = [d_{ij}]$. The post-event state of $\hat{\mathbf{\Psi}}_M$ is $\hat{\mathbf{\Theta}}_M^k = [\hat{S}_{1M}^k \quad \hat{S}_{2M}^k \quad \ldots \quad \hat{S}_{NM}^k]$. We have the following finding.

Theorem 2: Due to the nature of the training sample pairs, $\hat{\mathbf{\Psi}}_1$, $\hat{\mathbf{\Psi}}_2$, $\ldots$, $\hat{\mathbf{\Psi}}_M$ learned by the stochastic-gradient-descent-based algorithms are optimized in such a manner that their Equivalent Overall Event Transition Matrix $\hat{\mathbf{\Psi}}_{1M}$ approximates $\tilde{\mathbf{\Psi}}_{1M}$ constituted by the target event transition matrices.

Proof: The learning algorithms only know the value of M and the training sample pairs $(\mathbf{\Theta}_0^k, \mathbf{\Theta}_M^k)$. Due to Corollary 2, $\mathbf{\Theta}_0^k$ and $\mathbf{\Theta}_M^k$ are treated and indeed used by the algorithms as pre- and post-event states of a single event. Also owing to Corollary 2, the event transition matrix of that apparent single event is the Equivalent Overall Event Transition Matrix $\tilde{\mathbf{\Psi}}_{1M}$. Values of the elements in $\hat{\mathbf{\Psi}}_1$, $\hat{\mathbf{\Psi}}_2$, $\ldots$, $\hat{\mathbf{\Psi}}_M$ are adjusted by the learning algorithms with the aim of minimizing the differences between $\hat{\mathbf{\Psi}}_{1M}$ and $\tilde{\mathbf{\Psi}}_{1M}$ through minimizing E_h in (5). **QED**

Example 3: Continue on Example 2. Because $\tilde{\mathbf{\Psi}}_1$, $\tilde{\mathbf{\Psi}}_2$, and $\tilde{\mathbf{\Psi}}_3$ are unknown, their Equivalent Overall Event Transition Matrix $\tilde{\mathbf{\Psi}}_{13}$ is hence unknown. Through learning, $\hat{\mathbf{\Psi}}_1$, $\hat{\mathbf{\Psi}}_2$, and $\hat{\mathbf{\Psi}}_3$ are obtained, and their Equivalent Overall Event Transition Matrix can be computed (i.e., $\hat{\mathbf{\Psi}}_{13} = \hat{\mathbf{\Psi}}_1 \circ \hat{\mathbf{\Psi}}_2 \circ \hat{\mathbf{\Psi}}_3$). $\hat{\mathbf{\Psi}}_{13}$ will resemble to $\tilde{\mathbf{\Psi}}_{13}$, and the degree of the similarity is related to E_h^{avg} in (6). The smaller the E_h^{avg}, the higher the similarity.

The following result furthers Theorem 2 by quantifying the similarity between $\hat{\mathbf{\Psi}}_{1M}$ and $\tilde{\mathbf{\Psi}}_{1M}$.

Theorem 3: When learning converges, error of approximating $\tilde{\mathbf{\Psi}}_{1M}$ by $\hat{\mathbf{\Psi}}_{1M}$ as measured by the Euclidean norm is bounded:

$$||\tilde{\mathbf{\Psi}}_{1M} - \hat{\mathbf{\Psi}}_{1M}|| \leq \frac{N \times \Delta S_{max}}{S_{min}^0}$$

where

$$\Delta S_{max} = \max_{1 \leq k \leq H} (|S_{1M}^k - \hat{S}_{1M}^k|, |S_{2M}^k - \hat{S}_{2M}^k|, \ldots, |S_{NM}^k - \hat{S}_{NM}^k|) \tag{7}$$

and from non-zero S_{j0}^k only,

$$S_{min}^0 = \min_{1 \leq k \leq H} (S_{10}^k, S_{20}^k, \ldots, S_{N0}^k). \tag{8}$$

To a limit, $\hat{\mathbf{\Psi}}_{1M} \to \tilde{\mathbf{\Psi}}_{1M}$ when $\Delta S_{max} \to 0$.

Proof: Note that according to the Euclidean norm,

$$||\mathbf{\Theta}_M^k - \hat{\mathbf{\Theta}}_M^k|| = \sqrt{(S_{1M}^k - \hat{S}_{1M}^k)^2 + (S_{2M}^k - \hat{S}_{2M}^k)^2 + \cdots + (S_{NM}^k - \hat{S}_{NM}^k)^2}$$

Thus, for any value of k, because of (7), the error bound is

$$||\mathbf{\Theta}_M^k - \hat{\mathbf{\Theta}}_M^k|| \leq \sqrt{N} \Delta S_{max}.$$

The assumed learning convergence implies that as learning progresses, ΔS_{max} decreases. At the same time, the value of E_1^{avg} also becomes smaller and smaller and can reduce to such an extent that it is smaller than ϵ.

Because of (3) and Corollary 1, the following equations are true:

$$
\begin{aligned}
&||\mathbf{\Theta}_M^k - \hat{\mathbf{\Theta}}_M^k|| \\
=&||\varphi(\mathbf{\Theta}_0^k, \tilde{\mathbf{\Psi}}_1\tilde{\mathbf{\Psi}}_2\ldots\tilde{\mathbf{\Psi}}_M) - \varphi(\mathbf{\Theta}_0^k, \hat{\mathbf{\Psi}}_1\hat{\mathbf{\Psi}}_2\ldots\hat{\mathbf{\Psi}}_M)|| \\
=&||\varphi(\mathbf{\Theta}_0^k, \tilde{\mathbf{\Psi}}_{1M}) - \varphi(\mathbf{\Theta}_0^k, \hat{\mathbf{\Psi}}_{1M})||.
\end{aligned}
$$

Note that for $\varphi(\mathbf{\Theta}_0^k, \tilde{\mathbf{\Psi}}_{1M})$, which yields $\mathbf{\Theta}_M^k$,

$$S_{jM}^k = max(S_{10}^k b_{1j}, S_{20}^k b_{2j}, ..., S_{N0}^k b_{Nj})$$

and for $\varphi(\mathbf{\Theta}_0^k, \hat{\mathbf{\Psi}}_{1M})$, which produces $\hat{\mathbf{\Theta}}_M^k$,

$$\hat{S}_{jM}^k = max(S_{10}^k d_{1j}, S_{20}^k d_{2j}, ..., S_{N0}^k d_{Nj}).$$

In the early rounds of learning, ΔS_{max} may not be very small. At that time, it is likely that for $S_{jM}^k = S_{I_10}^k b_{I_1j}$ and $\hat{S}_{jM}^k = S_{I_20}^k d_{I_2j}$, $I_1 \neq I_2$. However, as learning is converging, ΔS_{max} will become smaller and smaller and will eventually reach such a point that $I_1 = I_2 = I$, $1 \leq I \leq N$, for $k = 1, 2, \ldots, H$. At that stage, for $\varphi(\mathbf{\Theta}_0^k, \tilde{\mathbf{\Psi}}_{1M}) - \varphi(\mathbf{\Theta}_0^k, \hat{\mathbf{\Psi}}_{1M})$,

$$|S_{jM}^k - \hat{S}_{jM}^k| = |S_{I0}^k b_{Ij} - S_{I0}^k d_{Ij}| = S_{I0}^k |b_{Ij} - d_{Ij}|$$

for $j = 1, 2, \ldots, N$. Due to (7), from the above equations, we have

$$S_{I0}^k |b_{Ij} - d_{Ij}| \leq \Delta S_{max}$$

which means when $S_{I0}^k \neq 0$ and owing to (8),

$$|b_{Ij} - d_{Ij}| \leq \frac{\Delta S_{max}}{S_{I0}^k} \leq \frac{\Delta S_{max}}{S_{min}^0}.$$

This last inequality leads us to:

$$
\begin{aligned}
||\mathbf{\Psi}_{1M} - \hat{\mathbf{\Psi}}_{1M}|| &= \sqrt{\sum_{j=1}^{N}\sum_{i=1}^{N}(b_{ij} - d_{ij})^2} \\
&\leq \sqrt{\sum_{j=1}^{N}\sum_{i=1}^{N}\left(\frac{\Delta S_{max}}{S_{min}^0}\right)^2} = \frac{N \times \Delta S_{max}}{S_{min}^0}.
\end{aligned}
$$

It is obvious that $\hat{\mathbf{\Psi}}_{1M} \to \tilde{\mathbf{\Psi}}_{1M}$ as $\Delta S_{max} \to 0$. **QED**

Theorem 3 establishes an error measure that characterizes the error between the target Equivalent Overall Event Transition Matrix and the Equivalent Overall Event Transition Matrix learned by the learning algorithms. The measure is related to the number of states N, pre-event state (which is represented by S_{min}^0), and maximal error between target and actual post-event states of the last event of the event sequence (which is characterized by ΔS_{max}). S_{min}^0 depends on the sample state pairs, which are given and unchangeable. Hence, to minimize the error measure is to minimize ΔS_{max}, which is intuitive and sensible.

Theorems 2 and 3 provide a much-needed new and insightful understanding of the event transition matrices produced by the learning algorithms for a multi-event FDES. Clearly, the notion of the Equivalent Overall Event Transition Matrix plays an essential role in this development.

VI. Illustrative Example

Development of the three-step multi-event SFDES identification technique utilizes: (1) the notion of the Equivalent Overall Event Transition Matrix, (2) the pre-existing Equation-Systems-Based Technique, and (3) the pre-existing stochastic-gradient-descent-based multi-event learning algorithms. Computer simulations were already performed previously to validate the theoretical development of the two pre-existing techniques for single-event SFDES and multi-event FDES. Their role in identifying multi-event SFDES is presented in detail above. The validity of the new method has been theoretically established, which can be verified in a mathematically rigorous manner. Computer simulation, in contrast, is much less effective in this regard and is indeed unsuitable.

The multi-event SFDES example provided below is meant to offer a concrete illustration demonstrating the step-by-step application of the new method. Because the method is innovative and there exists no similar method in the literature, a comparative study is neither warranted, nor possible.

A. Example

Parameters of the example multi-event SFDES are as follows: three individual states ($N = 3$), two fuzzy automata ($R = 2$), numbers of consecutive events for the first and second fuzzy automata are 3 ($M_1 = 3$) and 2 ($M_2 = 2$), respectively. The parameters are deliberately chosen to be relatively modest yet representative, as they offer insights that can be generalized to larger values when necessary. Despite their seemingly modest values, the chosen parameters encapsulate fundamental dynamics applicable across varying scales. This deliberate choice allows for a focused presentation of the new method, providing a clear understanding of its multi-step identification process. Moreover, the insights gained from analyzing this small-scale scenario can be extrapolated to larger configurations with confidence, as the underlying principles are the same.

The randomly-generated true/target event transition matrices of the first fuzzy automaton are

$$\tilde{\mathbf{\Psi}}_1^1 = \begin{bmatrix} 1.0 & 0.81 & 0.18 \\ 0.19 & 0.14 & 0.66 \\ 0.32 & 0.78 & 0.3 \end{bmatrix}, \quad \tilde{\mathbf{\Psi}}_2^1 = \begin{bmatrix} 0.82 & 0.28 & 0.32 \\ 0.21 & 0.62 & 0.49 \\ 0.83 & 0.45 & 0.64 \end{bmatrix}$$

$$\tilde{\mathbf{\Psi}}_3^1 = \begin{bmatrix} 0.86 & 0.68 & 0.29 \\ 0.54 & 0.01 & 0.01 \\ 0.29 & 0.91 & 0.48 \end{bmatrix}$$

with the true/target event transition matrices of the second fuzzy automaton being

$$\tilde{\mathbf{\Psi}}_1^2 = \begin{bmatrix} 0.17 & 0.85 & 0.48 \\ 0.63 & 0.67 & 0.7 \\ 0.38 & 0.68 & 0.16 \end{bmatrix}, \quad \tilde{\mathbf{\Psi}}_2^2 = \begin{bmatrix} 0.33 & 0.84 & 0.07 \\ 0.96 & 0.19 & 0.66 \\ 0.47 & 0.33 & 0.22 \end{bmatrix}.$$

There are a total of 400 sample pairs ($H = 400$) - 280 pairs are intended for the first fuzzy automaton and 120 pairs intended for the second automaton. That makes theoretical occurrence probabilities for the first and second fuzzy automata 0.7 and 0.3, respectively. For each sample pair, the post-event state vector is computed by using the pre-event state vector, the fuzzy event transition matrices involved, and the max-product fuzzy inference.

B. Simulation Programs and Settings

We modified our previous programs in MATLAB (version 2021a) that were for implementing the Equation-Systems-Based Technique for single-event FDES as well as the stochastic-gradient-descent-based learning algorithms for multi-event FDES. The modified programs were employed to identify the example multi-event SFDES.

Generators producing uniformly distributed random numbers were used to create: (1) the true/target event transition matrices, and (2) the sample pre-event-state-post-event-state pairs. MATLAB commands RandStream('mlfg6331_64') and randsample() were jointly used to distribute the pre-event state vectors to the fuzzy automata according to provided occurrence probabilities. In the mean time, they also created a random order list that reflected the occurrence probabilities so that the programs could use it to control which fuzzy automaton to act on which pre-event state.

The programs ran on a 2021 MacBook Pro 13" equipped with M1 CPU chips, 16 GB RAM, and the macOS Monterey. In the third identification step, the multi-event SFDES identification method employed the pre-existing stochastic-gradient-descent-based multi-event learning algorithms to learn the event transition matrices of the two fuzzy automata. The adaptive learning rate version of the algorithms was adopted with the hyperparameter τ empirically set to 3.5. The algorithms were provided with values of $M_1 = 3$ and $M_2 = 2$. The following settings were the same for learning either automaton: (1) the initial event transition matrices were all random, and (2) the learning process consisted of 100 epochs, each epoch comprising the training of the model using all the sample pairs. At the end of the 100th epoch, the overall learning error was measured by E_h^{avg} in (6), where h = 1 or 2 to indicate the first or second fuzzy automaton in the example.

C. Simulation Results

MATLAB commands RandStream('mlfg6331_64') and randsample() in the programs assign 270 and 130 sample pairs to the first and second fuzzy automata, respectively. This process and its outcome are random and are completely controlled and determined by the commands. This actual distribution of the 400 pairs slightly modifies the intended occurrence probabilities of 0.7 and 0.3 to 0.675 and 0.325 for the first and second fuzzy automata, respectively.

In the first identification step, the multi-event identification method utilizes the Equation-Systems-Based Technique and correctly finds out, from $N \times H = 1200$ systems of equations, that there are two fuzzy automata (i.e., $R = 2$), and also identifies the Equivalent Overall Event Transition Matrix for the event sequence of the first fuzzy automaton as

$$\hat{\mathbf{\Psi}}_{13}^1 = \begin{bmatrix} 0.7052 & 0.5576 & 0.2378 \\ 0.471108 & 0.384384 & 0.202752 \\ 0.261144 & 0.347802 & 0.183456 \end{bmatrix}$$

and the Equivalent Overall Event Transition Matrix for the event sequence of the second fuzzy automaton as

$$\hat{\mathbf{\Psi}}_{12}^2 = \begin{bmatrix} 0.816 & 0.1615 & 0.561 \\ 0.6432 & 0.5292 & 0.4422 \\ 0.6528 & 0.3192 & 0.4488 \end{bmatrix}.$$

The CPU time is 1.45 seconds. As anticipated, they are identical to the corresponding Equivalent Overall Event Transition Matrices that are theoretically computed using the target event transition matrices, which are $\tilde{\mathbf{\Psi}}_{13}^1 = \tilde{\mathbf{\Psi}}_1^1 \circ \tilde{\mathbf{\Psi}}_2^1 \circ \tilde{\mathbf{\Psi}}_3^1$ for the first automaton, and $\tilde{\mathbf{\Psi}}_{12}^2 = \tilde{\mathbf{\Psi}}_1^2 \circ \tilde{\mathbf{\Psi}}_2^2$ for the second fuzzy automaton.

In the second identification step, the multi-event SFDES identification method uses the Equation-Systems-Based Technique and correctly traces back and finds all the pre- and post-event state pairs that are associated with each of the two Equivalent Overall Event Transition Matrices. More specifically, it correctly finds the 270 and 130 sample pairs for the first and second fuzzy automata, respectively. Using these pairs, the method calculates the occurrence frequencies for the first and second fuzzy automata. The respective results, $170/400$ = 0.675 and $130/400$ = 0.325, are precisely accurate.

In the third identification step, with the 270 pre- and post-event state pairs associated with the first Equivalent Overall Event Transition Matrix and the learning rates set to 0.02 (this value was determined experimentally, without significant effort invested in finding the optimal value), the gradient descent multi-event learning algorithms learn the three event transitions matrices of the first fuzzy automaton with the following result:

$$\hat{\mathbf{\Psi}}_1^1 = \begin{bmatrix} 1.000 & 0.862 & 0.770 \\ 0.667 & 0.576 & 0.656 \\ 0.370 & 0.319 & 0.594 \end{bmatrix},$$

$$\hat{\mathbf{\Psi}}_2^1 = \begin{bmatrix} 1.000 & 0.506 & 0.214 \\ 0.074 & 0.898 & 0.564 \\ 0.756 & 0.405 & 0.984 \end{bmatrix},$$

$$\hat{\mathbf{\Psi}}_3^1 = \begin{bmatrix} 0.338 & 0.186 & 0.196 \\ 0.911 & 0.721 & 0.208 \\ 0.446 & 0.595 & 0.314 \end{bmatrix}.$$

At the end of the learning process (i.e., the end of the 100th epoch), $E_1^{Avg} = 1.88319 \times 10^{-14}$, indicating excellent convergence. The learning process takes 4.17 seconds of CPU time. Although the learned matrices differ from their target matrices, as anticipated based on our analyses above, the absolute value of the element-wise differences between the theoretical Equivalent Overall Event Transition Matrix $\tilde{\mathbf{\Psi}}_{13}^1$ and the actual one computed using $\hat{\mathbf{\Psi}}_{13}^1 = \hat{\mathbf{\Psi}}_1^1 \circ \hat{\mathbf{\Psi}}_2^1 \circ \hat{\mathbf{\Psi}}_3^1$ is under 10^{-12}. In other words, the learned Equivalent Overall Event Transition Matrix is extremely close to the true one. This is consistent with Theorem 3 given the very small E_1^{Avg}.

The gradient descent algorithms then learn the two event transition matrices of the second fuzzy automaton using the 130 sample state pairs. The learning rates are empirically set to 0.1. The learning process takes less than 3.1 seconds of

CPU time to produce the following event transition matrices

$$\hat{\mathbf{\Psi}}_1^2 = \begin{bmatrix} 0.253 & 0.912 & 0.825 \\ 0.827 & 0.719 & 0.544 \\ 0.499 & 0.730 & 0.660 \end{bmatrix},$$

$$\hat{\mathbf{\Psi}}_2^2 = \begin{bmatrix} 0.777 & 0.640 & 0.259 \\ 0.777 & 0.095 & 0.615 \\ 0.989 & 0.001 & 0.268 \end{bmatrix},$$

which are different from their corresponding target matrices. The absolute value of the element-wise differences between the theoretical Equivalent Overall Event Transition Matrix $\tilde{\mathbf{\Psi}}_{12}^2$ and the actual one computed using $\hat{\mathbf{\Psi}}_{12}^2 = \hat{\mathbf{\Psi}}_1^2 \circ \hat{\mathbf{\Psi}}_2^2$, nevertheless, is less than 10^{-15}. The learning process converges strongly, evidenced by $E_2^{Avg} = 1.76506 \times 10^{-16}$.

In summary, the simulation results for the example align well with our theoretical analyses.

VII. Conclusion

Multi-event SFDES can be useful in modeling of a class of practical systems in various industries. We develop a novel three-step method capable of modeling such systems. The development of the method relies on the notion of the Equivalent Overall Event Transition Matrix that we newly introduced and utilizes the Equation-Systems-Based Technique and stochastic-gradient-descent-based multi-event learning algorithms that we previously developed for identifying single-event SFDES and multi-event FDES, respectively (we are not aware of any new literature on these topics). This new notion enables us to effectively transform the problem of multi-event SFDES identification to the problems of single-event SFDES identification and identification of multiple multi-event FDES, paving the way for subsequent direct utilization of the two pre-existing techniques in the new three-step multi-event SFDES identification method. We carry out a theoretical analysis of the event transition matrices produced by the learning algorithms and reveal the interconnections between these matrices, the Equivalent Overall Event Transition Matrices constituted by them, and the target event transition matrices.

Biomedical application challenges are notably conducive to modeling as multi-event SFDES. This is owing to the stochastic and unpredictable nature of disease progression, regardless of treatment. Moreover, states of patients and diseases inherently possess vagueness and imprecision. The new three-step identification method introduces a novel modeling approach previously unavailable, offering the potential to more effectively address the unmet needs within these domains, as well as in other industries.

As mentioned earlier, because the stochastic-gradient-descent-based algorithms are allowed to use only pre-event state of the first event and post-event state of the last event of an event sequence, which is an uncompromising yet realistic assumption that we have chosen to adopt, the event transition matrices of a multi-event FDES it learns may not be the same as the underlying event transition matrices (but the Equivalent Overall Event Transition Matrix it learns can be very close to the underlying Equivalent Overall Event Transition Matrix). The three-step identification method utilizes these learning algorithms; hence, the identification of multi-event SFDES is bound by the same limitations. How to improve the multi-event FDES identification outcome under this substantial constraint is an important but challenging issue for future research.

VIII. Appendix

Proof of Lemma 1:

We need to prove

$$\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 = (\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1) \circ \tilde{\mathbf{\Psi}}_2 = \mathbf{\Theta}_0 \circ (\tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2). \quad (9)$$

We will prove the first part of the equations first. Note that

$$\begin{aligned}
&(\mathbf{\Theta_0} \circ \tilde{\mathbf{\Psi}}_1) \circ \tilde{\mathbf{\Psi}}_2 \\
&= [S_{10} \quad S_{20} \quad ... \quad S_{N0}] \circ \begin{bmatrix} a_{111} & a_{121} & ... & a_{1N1} \\ a_{211} & a_{221} & ... & a_{2N1} \\ ... & & & \\ a_{N11} & a_{N21} & ... & a_{NN1} \end{bmatrix} \\
&\circ \begin{bmatrix} a_{112} & a_{122} & ... & a_{1N2} \\ a_{212} & a_{222} & ... & a_{2N2} \\ ... & & & \\ a_{N12} & a_{N22} & ... & a_{NN2} \end{bmatrix} \\
&= [S_{11} \quad S_{21} \quad ... \quad S_{N1}] \circ \begin{bmatrix} a_{112} & a_{122} & ... & a_{1N2} \\ a_{212} & a_{222} & ... & a_{2N2} \\ ... & & & \\ a_{N12} & a_{N22} & ... & a_{NN2} \end{bmatrix} \\
&= [S_{12} \quad S_{22} \quad ... \quad S_{N2}]
\end{aligned}$$

where

$$S_{j1} = max(S_{10}a_{1j1}, S_{20}a_{2j1}, ..., S_{N0}a_{Nj1}) \quad (10)$$

and

$$S_{j2} = max(S_{11}a_{1j2}, S_{21}a_{2j2}, ..., S_{N1}a_{Nj2}). \quad (11)$$

Obviously, calculation of $\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2$ will follow the exactly same steps. Thus,

$$\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2 = (\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1) \circ \tilde{\mathbf{\Psi}}_2.$$

We then prove $(\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1) \circ \tilde{\mathbf{\Psi}}_2 = \mathbf{\Theta}_0 \circ (\tilde{\mathbf{\Psi}}_1 \circ \tilde{\mathbf{\Psi}}_2)$. Substituting (10) into (11), one gets, for $(\mathbf{\Theta}_0 \circ \tilde{\mathbf{\Psi}}_1) \circ \tilde{\mathbf{\Psi}}_2$,

$$\begin{aligned}
S_{j2} = max(&max(S_{10}a_{111}a_{1j2}, S_{20}a_{211}a_{1j2}, ..., S_{N0}a_{N11}a_{1j2}), \\
&max(S_{10}a_{121}a_{2j2}, S_{20}a_{221}a_{2j2}, ..., S_{N0}a_{N21}a_{2j2}), \\
&\cdots \\
&max(S_{10}a_{1N1}a_{Nj2}, S_{20}a_{2N1}a_{Nj2}, ..., S_{N0}a_{NN1}a_{Nj2}))
\end{aligned}$$

which can be simplified to

$$\begin{aligned}
S_{j2} = max(&S_{10}a_{111}a_{1j2}, S_{20}a_{211}a_{1j2}, ..., S_{N0}a_{N11}a_{1j2}, \\
&S_{10}a_{121}a_{2j2}, S_{20}a_{221}a_{2j2}, ..., S_{N0}a_{N21}a_{2j2}, \\
&\cdots \\
&S_{10}a_{1N1}a_{Nj2}, S_{20}a_{2N1}a_{Nj2}, ..., S_{N0}a_{NN1}a_{Nj2}).
\end{aligned} \quad (12)$$

On the other hand, for $\Theta_0 \circ (\tilde{\Psi}_1 \circ \tilde{\Psi}_2)$,

$$\tilde{\Psi}_1 \circ \tilde{\Psi}_2 = \begin{bmatrix} a_{111} & a_{121} & ... & a_{1N1} \\ a_{211} & a_{221} & ... & a_{2N1} \\ ... \\ a_{N11} & a_{N21} & ... & a_{NN1} \end{bmatrix} \circ \begin{bmatrix} a_{112} & a_{122} & ... & a_{1N2} \\ a_{212} & a_{222} & ... & a_{2N2} \\ ... \\ a_{N12} & a_{N22} & ... & a_{NN2} \end{bmatrix}.$$

Let

$$\tilde{\Psi} = \tilde{\Psi}_1 \circ \tilde{\Psi}_2$$

where $\tilde{\Psi}$ is defined as

$$\tilde{\Psi} = \begin{bmatrix} b_{11} & b_{12} & ... & b_{1N} \\ b_{21} & b_{22} & ... & b_{2N} \\ ... \\ b_{N1} & b_{N2} & ... & b_{NN} \end{bmatrix}.$$

For the max-product operations in $\tilde{\Psi}_1 \circ \tilde{\Psi}_2$, it is easy to see that

$$b_{ij} = max(a_{i11}a_{1j2}, a_{i21}a_{2j2}, \ldots, a_{iN1}a_{Nj2}).$$

Therefore,

$$\Theta_0 \circ (\tilde{\Psi}_1 \circ \tilde{\Psi}_2) = \Theta_0 \circ \tilde{\Psi} = [S'_{12} \quad S'_{22} \quad ... \quad S'_{N2}]$$

where

$$\begin{aligned} S'_{j2} &= max(S_{10}b_{1j}, S_{20}b_{2j}, ..., S_{N0}b_{Nj}) \\ &= max(S_{10}max(a_{111}a_{1j2}, a_{121}a_{2j2}, \ldots, a_{1N1}a_{Nj2}), \\ &\qquad S_{20}max(a_{211}a_{1j2}, a_{221}a_{2j2}, \ldots, a_{2N1}a_{Nj2}), \\ &\qquad \vdots \\ &\qquad S_{N0}max(a_{N11}a_{1j2}, a_{N21}a_{2j2}, \ldots, a_{NN1}a_{Nj2})) \\ &= max(S_{10}a_{111}a_{1j2}, S_{10}a_{121}a_{2j2}, \ldots, S_{10}a_{1N1}a_{Nj2}), \\ &\qquad S_{20}a_{211}a_{1j2}, S_{20}a_{221}a_{2j2}, \ldots, S_{20}a_{2N1}a_{Nj2}), \\ &\qquad \vdots \\ &\qquad S_{N0}a_{N11}a_{1j2}, S_{N0}a_{N21}a_{2j2}, \ldots, S_{N0}a_{NN1}a_{Nj2}). \end{aligned}$$

Comparing S'_{j2} with S_{j2} in (12), one sees $S'_{j2} = S_{j2}$. **QED**

REFERENCES

[1] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Springer Nature, 3rd ed., 2021.

[2] W. M. Wonham, K. Cai, *et al.*, "Supervisory control of discrete-event systems," 2019.

[3] W. Wonham, K. Cai, and K. Rudie, "Supervisory control of discrete-event systems: A brief history," *Annual Reviews in Control*, vol. 45, pp. 250–256, 2018.

[4] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM journal on control and optimization*, vol. 25, no. 1, pp. 206–230, 1987.

[5] F. Lin and W. M. Wonham, "On observability of discrete-event systems," *Information sciences*, vol. 44, no. 3, pp. 173–198, 1988.

[6] F. Lin and H. Ying, "Modeling and control of fuzzy discrete event systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 32, no. 4, pp. 408–415, 2002.

[7] Y. Cao, M. Ying, and G. Chen, "State-based control of fuzzy discrete-event systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 2, pp. 410–424, 2007.

[8] F. Lin and H. Ying, "State-feedback control of fuzzy discrete-event systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 3, pp. 951–956, 2009.

[9] Y. Cao and M. Ying, "Supervisory control of fuzzy discrete event systems," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 2, pp. 366–371, 2005.

[10] D. Qiu, "Supervisory control of fuzzy discrete event systems: a formal approach," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 35, no. 1, pp. 72–88, 2005.

[11] W. Deng and D. Qiu, "Supervisory control of fuzzy discrete-event systems for simulation equivalence," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 178–192, 2014.

[12] W. Deng and D. Qiu, "Bifuzzy discrete event systems and their supervisory control theory," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 6, pp. 2107–2121, 2015.

[13] Y. Cao and M. Ying, "Observability and decentralized control of fuzzy discrete-event systems," *IEEE Transactions on Fuzzy Systems*, vol. 14, no. 2, pp. 202–216, 2006.

[14] W. Deng and D. Qiu, "State-based decentralized diagnosis of bi-fuzzy discrete event systems," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 4, pp. 854–867, 2016.

[15] A. Jayasiri, G. K. Mann, and R. G. Gosine, "Generalizing the decentralized control of fuzzy discrete event systems," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 4, pp. 699–714, 2011.

[16] X. Yin, "A belief-evolution-based approach for online control of fuzzy discrete-event systems under partial observation," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 6, pp. 1830–1836, 2016.

[17] A. O. Mekki, F. Lin, and H. Ying, "On detectabilities of fuzzy discrete event systems," *IEEE Transactions on Fuzzy Systems*, vol. 30, no. 2, pp. 426–436, 2020.

[18] E. Kilic, "Diagnosability of fuzzy discrete event systems," *Information Sciences*, vol. 178, no. 3, pp. 858–870, 2008.

[19] F. Liu and D. Qiu, "Diagnosability of fuzzy discrete-event systems: A fuzzy approach," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 2, pp. 372–384, 2009.

[20] B. Benmessahel, M. Touahria, F. Nouioua, J. Gaber, and P. Lorenz, "Decentralized prognosis of fuzzy discrete-event systems," *Iranian Journal of Fuzzy Systems*, vol. 16, no. 3, pp. 127–143, 2019.

[21] T. Zhu, F. Liu, and C. Xiao, "Reliable fuzzy prognosability of decentralized fuzzy discrete-event systems and verification algorithm," *Information Sciences*, 2023.

[22] B. Benmessahel, M. Touahria, and F. Nouioua, "Predictability of fuzzy discrete event systems," *Discrete Event Dynamic Systems*, vol. 27, pp. 641–673, 2017.

[23] W. Deng, J. Yang, C. Jiang, and D. Qiu, "Opacity of fuzzy discrete event systems," in *2019 Chinese control and decision conference (CCDC)*, pp. 1840–1845, IEEE, 2019.

[24] W. Deng, D. Qiu, and J. Yang, "Opacity measures of fuzzy discrete event systems," *IEEE Transactions on Fuzzy Systems*, vol. 29, no. 9, pp. 2612–2622, 2020.

[25] M. Nie and W. W. Tan, "Theory of generalized fuzzy discrete-event systems," *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 1, pp. 98–110, 2014.

[26] X. Du, H. Ying, and F. Lin, "Theory of extended fuzzy discrete-event systems for handling ranges of knowledge uncertainties and subjectivity," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 2, pp. 316–328, 2008.

[27] S. Raczynski, "The space of models, semi-discrete events with fuzzy logic," in *Models for Research and Understanding: Exploring Dynamic Systems, Unconventional Approaches, and Applications*, pp. 207–228, Springer, 2023.

[28] S. Zhang and J. Chen, "Modelling of fuzzy discrete event systems based on a generalized linguistic variable and their generalized possibilistic kriple structure representation," in *Advances in Natural Computation, Fuzzy Systems and Knowledge Discovery: Proceedings of the ICNC-FSKD 2022*, pp. 437–446, Springer, 2023.

[29] F. Lin, H. Ying, R. D. MacArthur, J. A. Cohn, D. Barth-Jones, and L. R. Crane, "Decision making in fuzzy discrete event systems," *Information Sciences*, vol. 177, no. 18, pp. 3749–3763, 2007.

[30] H. Ying, F. Lin, R. D. MacArthur, J. A. Cohn, D. C. Barth-Jones, H. Ye, and L. R. Crane, "A fuzzy discrete event system approach to determining optimal hiv/aids treatment regimens," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 4, pp. 663–676, 2006.

[31] H. Ying, F. Lin, R. D. MacArthur, J. A. Cohn, D. C. Barth-Jones, H. Ye, and L. R. Crane, "A self-learning fuzzy discrete event system for hiv/aids treatment regimen selection," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 37, no. 4, pp. 966–979, 2007.

[32] R. Huq, G. K. Mann, and R. G. Gosine, "Behavior-modulation technique in mobile robotics using fuzzy discrete event system," *IEEE Transactions on Robotics*, vol. 22, no. 5, pp. 903–916, 2006.
[33] R. Liu, Y.-X. Wang, and L. Zhang, "An fdes-based shared control method for asynchronous brain-actuated robot," *IEEE Transactions on Cybernetics*, vol. 46, no. 6, pp. 1452–1462, 2015.
[34] K. W. Schmidt and Y. S. Boutalis, "Fuzzy discrete event systems for multiobjective control: Framework and application to mobile robot navigation," *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 5, pp. 910–922, 2012.
[35] L. Danmei, L. Weichun, Z. Hui, and S. Shihuang, "A fuzzy discrete event system control and decision making in air conditioning system," in *2008 Chinese control and decision conference*, pp. 3147–3151, IEEE, 2008.
[36] D. Li, W. Lan, H. Zhou, and S. Shao, "Control of fuzzy discrete event systems and its application to air conditioning system," *International Journal of Modelling, Identification and Control*, vol. 8, no. 2, pp. 122–129, 2009.
[37] N. J. Khan, G. Ahamad, M. Naseem, and Q. R. Khan, "Fuzzy discrete event system (fdes): A survey," in *Renewable Power for Sustainable Growth: Proceedings of International Conference on Renewal Power (ICRP 2020)*, pp. 531–544, Springer, 2021.
[38] D. Qiu and F. Liu, "Fuzzy discrete-event systems under fuzzy observability and a test algorithm," *IEEE Transactions on Fuzzy Systems*, vol. 17, no. 3, pp. 578–589, 2008.
[39] H. Ying, F. Lin, and R. Sherwin, "Fuzzy discrete event systems with gradient-based online learning," in *2019 IEEE international conference on fuzzy systems (FUZZ-IEEE)*, pp. 1–6, IEEE, 2019.
[40] H. Ying and F. Lin, "Online self-learning fuzzy discrete event systems," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 9, pp. 2185–2194, 2019.
[41] H. Ying and F. Lin, "Learning fuzzy automaton's event transition matrix when post-event state is unknown," *IEEE Transactions on Cybernetics*, vol. 52, no. 6, pp. 4993–5000, 2020.
[42] H. Ying and F. Lin, "Self-learning fuzzy automaton with input and output fuzzy sets for system modelling," *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2022.
[43] F. Lin and H. Ying, "Supervised learning of multievent transition matrices in fuzzy discrete-event systems," *IEEE Transactions on Cybernetics*, 2022.
[44] Y. Cao, Y. Ezawa, G. Chen, and H. Pan, "Modeling and specification of nondeterministic fuzzy discrete-event systems," *Decision Making Under Constraints*, pp. 45–58, 2020.
[45] F. Lin and H. Ying, "Modeling and control of probabilistic fuzzy discrete event systems," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 6, no. 2, pp. 399–408, 2021.
[46] H. Ying and F. Lin, "Stochastic fuzzy discrete event systems and their model identification," *IEEE Transactions on Cybernetics, in press*.
[47] H. Ying and F. Lin, "Identification of single-event stochastic fuzzy discrete event systems: An equation-systems-based approach," *IEEE Transactions on Fuzzy Systems, in press*.