Stochastic Fuzzy Discrete Event Systems and Their Model Identification

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

***Abstract*—We introduce a new class of fuzzy discrete event systems called stochastic fuzzy discrete event systems (SFDES), which is significantly different from the probabilistic fuzzy discrete event systems (PFDES) in the literature. It offers an effective modeling framework for applications that are unsuitable for the PFDES framework. A SFDES is comprised of multiple fuzzy automata that occur randomly one at time with different occurrence probabilities. It uses either the max-product fuzzy inference or the max-min fuzzy inference. This paper focuses on single-event SFDES - each of the fuzzy automata of such a SFDES has one event. Assuming nothing is known about a SFDES, we develop an innovative technique capable of determining number of fuzzy automata and their event transition matrices as well as estimating their occurrence probabilities. The technique, called Prerequired-Pre-Event-State-Based Technique, creates and uses merely N particular pre-event state vectors of dimension N to identify event transition matrices of M fuzzy automata, involving a total of MN^2 unknown parameters. One necessary and sufficient condition and three sufficient conditions are established for the identification of SFDES with different settings. The Technique does not have any adjustable parameter or hyperparameter to set. A numerical example is provided to concretely illustrate the Technique.**

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

I. Introduction

The theory of fuzzy discrete event systems (FDES) is an extension of the theory of discrete event systems (DES), which is introduced in the 1980s [1][2][3]. A DES consists of discrete states, discrete events, and transitions from states to states as events occur in a sequence. The theory of DES can be used to effectively model a class of practical systems that cannot be modeled in the traditional system theory using differential or difference equations. DES are driven by occurrences of events. An event records substantial qualitative changes in the system, which is a distinctive characteristic common to DES. One branch of the DES theory is the supervisory control developed in the 1980s [1-5]. Fundamental concepts such as controllability [1] and observability [2] are introduced that provide a necessary and sufficient condition for the existence of a supervisor. Since then, other topics, including online control, robust control, hierarchical control, decentralized control, and limited-lookahead control, have been investigated. Supervisory control of DES has been applied to manufacturing systems, communication networks, computer systems, power systems, transportation systems, and other practical systems.

In DES, states are binary (1 or 0, yes or no). However, many event-driven systems in practice have states that are not binary, but continuous (both yes and no to a certain degree at the same time). This is especially true in healthcare, where health state of a person is often ambiguous and subjective. We cannot say that a person's health is excellent for certain, rather we can only say that his health is excellent to an extent.

The best way to describe and model ambiguity and subjectivity is by fuzzy sets and fuzzy logic [6]. Therefore, we generalize DES by introducing FDES to describe event-driven systems whose states are ambiguous and subjective [7]. This is done by using fuzzy automata rather than (crisp) automata. In a fuzzy automaton, "state" and "event" are extended to "fuzzy state" and "fuzzy event". Formally, a fuzzy automaton can be represented mathematically as

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

The elements of G are as follows. $\mathbf{Q}$ is the fuzzy state vector space of dimension N that all fuzzy state vectors belong to. $\mathbf{q_o}$ is the initial fuzzy state vector, representing the fuzzy state before the occurrence of any event (pre-event). Σ is the set of fuzzy events. Each event is given by an $N \times N$ event transition matrix. $\varphi : \mathbf{Q} \times \Sigma \rightarrow \mathbf{Q}$ is the event transition mapping executed by fuzzy inference. Either the max-product fuzzy inference or the max-min fuzzy inference can be used.

FDES generalizes a binary state to a fuzzy state by allowing the values of the elements in a state vector to be in the interval of $[0, 1]$, rather than either 0 or 1 as in DES. Furthermore, elements of an event transition matrix in FDES are also allowed to be in $[0, 1]$ rather than $\{0, 1\}$ as in DES. In this way, fuzzy state and fuzzy event can have partial memberships. The event transition mapping is also generalized using the max-product fuzzy inference or the max-min fuzzy inference. It is not difficult to see that FDES/fuzzy automaton contain DES/automaton as a special case. Besides fuzzy automata, we also investigate observability, optimal control, and parallel composition of FDES in [7].

Since the publication of [7], other researchers have extended the theory of FDES in a number of important directions, including supervisory control [8][9][11][31], state-feedback control [18], decentralized control [12][13][34], state-based control [17], online control [16], diagnosability [13][14][15], detectabilities [45], predictability [32], prognosis [25][48], opacity [35], and others [49][50]. Furthermore, the FDES with type-1 fuzzy sets are extended to those with type-2 fuzzy sets

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.

that can better handle ambiguity [19]. Relationship between controllability of FDES and that of DES is investigated [30]. A generalized theory of FDES is also proposed [30]. We also apply the FDES theory to decision making [20]. In particular, we investigate how to use the FDES theory in optimal regimen selection in HIV/AIDS treatments and obtain good retrospective clinical results [21][22]. Other researchers apply the FDES theory to air conditioning system [41][42], mobile robots [27][29][33], and other areas. For a recent survey of results in FDES, the reader is referred to [40].

One key element in a fuzzy automaton is the event transition matrix. It tells how the system changes states after the occurrence of an event. It also determines observability [10], predictability [32], and other properties of the system. One way to generate the event transition matrix is for the modeler to ask domain experts (e.g., physicians) for a specific application in hand and then translate the domain knowledge to the making of the matrix. This is done, for example, in [21][22]. However, manual generation of event transition matrices is a challenging and time-consuming task. It is often the bottleneck in applying the FDES theory to real-world problems.

To overcome this bottleneck, stochastic-gradient-descent-based online learning algorithms are recently developed to generate event transition matrices using pre- and post-event state vector pairs and their associated variables. Depending on what data are available in practice, four learning algorithms are developed. The first algorithm can be used when true pre- and post-event states are available [24]. The second algorithm can be employed when true post-event state is available and true pre-event state is not but is known to be somehow associated with (input) variables whose values are available [23][24]. The third algorithm can be utilized when true pre-event state is available, but true post-event state is not. However, each of the individual post-event states is known to be somehow associated with a (output) variable whose value is available [38]. The fourth algorithm can be used when both true pre-event state and true post-event state are unavailable but they are somehow associated with variables whose values are available [46]. Learning algorithms to learn multi-event transition matrices simultaneously when a sequence of events occurs is also developed [43]. To the best of our knowledge, these are the only results on the identification of FDES in the current literature.

In all the works mentioned above, the FDES considered are deterministic in the sense that post-event state vector is computed using pre-event state vector and event transition matrix without any randomness involved in the process. There are only limited works on nondeterministic FDES and probabilistic FDES (PFDES for short). For example, nondeterministic FDES in studied in [26] and PFDES is investigated in [44]

In this paper, our study is focused on a new and innovative class of probabilistic FDES, which is called the stochastic FDES (SFDES for short). A SFDES is comprised of multiple fuzzy automata, each of which represents a fuzzy event and has its own occurrence probability. Given a pre-event state vector, we do not know the post-event state vector beforehand because it depends on which fuzzy event will happen. It is a random process. Of course, the post-event state vector can be computed using the event transition matrix of the fuzzy automaton representing the event, if available, after the event is known to occur.

While both SFDES and PFDES [44] handle randomness, they are substantially different as they are created for modeling different types of applications. A PFDES consists of only one fuzzy automaton; which one of multiple event trajectories will take place next from current state is probabilistic. In contrast, a SFDES can be comprised of many fuzzy automata; which one to occur is random and depends on its occurrence probability. This capability represents a significant advantage of SFDES over PFDES and makes the former more suitable for certain applications that are difficult to be tackled by the latter. To the best of our knowledge, the notion of SFDES has never been proposed or mentioned in the literature. It is one of the novelties of this paper.

A logical question to follow is how to identify the model of a SFDES? In this paper, we develop an identification technique called the Prerequired-Pre-Event-State-Based Technique. It creates and employs only N particular pre-event state vectors for complete identification of M randomly-occurring fuzzy automata in a single-event SFDES, each of which is represented by an $N \times N$ event transition matrix. The total number of matrix elements to be identified is MN^2.

In the next section, we will formally introduce SFDES and single-event SFDES. We will then present our identification technique, and establish two necessary and sufficient conditions and two sufficient conditions for it to identify single-event SFDES under different settings in Section III. A numerical example is supplied to demonstrate how exactly the Technique works in Section IV. Section V concludes the paper.

II. Introduction to Stochastic Fuzzy Discrete Event Systems

A SFDES consists of M fuzzy automata ($M > 1$), each acts only according to its own occurrence probability that is independent of system's pre-event state or anything else. A fuzzy automaton model [7] has N individual fuzzy states whose membership values are all in [0, 1]. The membership values form an $1 \times N$ vector to collectively represent the overall state of the automaton, which is referred as the system state. After an event has happened, the system will transfer from pre-event state (first such state is initial state), through its $N \times N$ event transition matrix $\tilde{\mathbf{\Psi}}$, to a new, post-event system state. Formally, a SFDES is given by

$$SFDES = \{(G_k, p_k) : k = 1, 2, ..., M\}, \tag{2}$$

where G_k is a fuzzy automaton and p_k is the probability of the occurrence of G_k. It is required that $p_1 + p_2 + ... + p_M = 1$.

In this paper, we focus on one important class of the SFDES, which is the single-event SFDES, and will develop an identification technique for it. By "single event", we mean there is no consecutive events occurring for any fuzzy automaton in a SFDES. In other words, each fuzzy automaton has one and only one event. Events of different fuzzy automata are all independent of one another in terms of when they will occur. Occurrence order of the events is random and the same event may take place repeatedly and/or consecutively. From application standpoint, single-event SFDES can be useful to model practical processes and systems, including those in biomedicine (e.g., diagnosis and treatment of various diseases). For example, progression of many diseases can be treated as single events and they are highly uncertain in a stochastic sense. Because of this, outcome of disease treatment, as measured by patient clinical state, is often difficult to predict with a high degree of certainty.

The SFDES definition (2) is broader and covers more than the single-event SFDES. It covers multi-event SFDES, in which two or more events of a fuzzy automaton can occur consecutively in a random fashion. Identification of such SFDES is desirable but seems to be technically more challenging. It is an interesting open research question. The work presented in this paper will be helpful in a future study of this and other related questions.

To identify a single-event SFDES is to identify the event matrices of the M fuzzy automata and determine their occurrence probabilities. Denote the event matrix for the k-th automaton as

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

where all the elements fall in [0, 1]. Note that the memberships in a row (or a column) of $\tilde{\boldsymbol{\Psi}}_k$ are not required to be summed to 1. Denote the probability of $\tilde{\boldsymbol{\Psi}}_k$ occurring as p_k.

Suppose there are H pre-event fuzzy state vectors, which are defined as:

$$\boldsymbol{\Theta_h} = [S_1^h \quad S_2^h \quad \ldots \quad S_N^h], h = 1, 2, ..., H.$$

Suppose also that event represented by $\tilde{\boldsymbol{\Psi}}_k$ occurs when pre-event state is $\boldsymbol{\Theta_h}$. The post-event fuzzy state, denoted as $\hat{\boldsymbol{\Theta}}_{\mathbf{k}}^{\mathbf{h}}$, can be computed by using the compositional rule of inference in fuzzy logic theory:

$$\begin{aligned} \hat{\boldsymbol{\Theta}}_{\mathbf{k}}^{\mathbf{h}} &= \boldsymbol{\Theta_h} \circ \tilde{\boldsymbol{\Psi}}_k \\ &= [S_1^h \quad S_2^h \quad \ldots \quad S_N^h] \circ \begin{bmatrix} a_{11k} & a_{12k} & \ldots & a_{1Nk} \\ a_{21k} & a_{22k} & \ldots & a_{2Nk} \\ \ldots & & & \\ a_{N1k} & a_{N2k} & \ldots & a_{NNk} \end{bmatrix} \\ &= [\hat{S}_{1k}^h \quad \hat{S}_{2k}^h \quad \ldots \quad \hat{S}_{Nk}^h]. \end{aligned}$$

The symbol $\circ$ denotes fuzzy inference operation, which can be the popular max-product operation or max-min operation. If the max-product inference is used,

$$\hat{S}_{jk}^h = max(S_1^h a_{1jk}, S_2^h a_{2jk}, ..., S_N^h a_{Njk}), \tag{3}$$

or if the max-min inference is employed,

$$\begin{aligned} \hat{S}_{jk}^h = &max(min(S_1^h, a_{1jk}), min(S_2^h, a_{2jk})) \\ &..., min(S_N^h, a_{Njk})). \end{aligned} \tag{4}$$

III. Prerequired-Pre-Event-State-Based Technique

Suppose there is a single-event SFDES, which can be in various form (e.g., mathematical formulations, computer program, physical systems). For this SFDES, its event transition matrices, how many of them, and their occurrence probabilities, are all assumed to be unknown. Note that N is not an inherent parameter of a SFDES. Rather it is a design parameter whose value needs to be chosen by the model developer according to the nature of the application of interest.

The tasks of our identification technique consist of: (1) creating and applying pre-event state vectors to the SFDES and recording their corresponding post-event state vectors, and (2) utilizing the the resulting pairs of the pre- and post-event state vectors to determine the event matrices and estimate their occurrence probabilities. We name the technique the Prerequired-Pre-Event-State-Based Technique because it creates N peculiar pre-event state vectors and reply on them to achieve SFDES model identification.

Let's first consider a special SFDES that consists of only one fuzzy automaton. This actually is a FDES and there is no randomness caused by multiple automata. Only one event matrix, $\tilde{\boldsymbol{\Psi}}_1$, is to be identified. We create a series of N $1 \times N$ special pre-event state vectors:

$$\begin{aligned} \boldsymbol{\Theta_1} &= [1 \quad 0 \quad \ldots \quad 0] \\ \boldsymbol{\Theta_2} &= [0 \quad 1 \quad \ldots \quad 0] \\ &\vdots \\ \boldsymbol{\Theta_N} &= [0 \quad 0 \quad \ldots \quad 1]. \end{aligned} \tag{5}$$

We feed them one at a time to FDES in the same order as shown in (5) (i.e., $\boldsymbol{\Theta_1}$, $\boldsymbol{\Theta_2}$,…,$\boldsymbol{\Theta_N}$). Completion of feeding all of them constitutes one round of feeding. We state the identification outcome as follows:

Theorem 1: A necessary and sufficient condition for identifying $\tilde{\boldsymbol{\Psi}}_1$ of a FDES that uses either max-product or max-min fuzzy inference method is to apply all the N pre-event state vectors in (5) to the FDES one at a time in any order.

Proof: We first prove the sufficiency. One round of the feeding of the N pre-event state vectors in the same order as listed in (5) will produce $\hat{\boldsymbol{\Theta}}_{\mathbf{1}}^{\mathbf{1}} = [a_{111} \quad a_{121} \quad \ldots \quad a_{1N1}]$, $\hat{\boldsymbol{\Theta}}_{\mathbf{1}}^{\mathbf{2}} = [a_{211} \quad a_{221} \quad \ldots \quad a_{2N1}]$, ..., $\hat{\boldsymbol{\Theta}}_{\mathbf{1}}^{\mathbf{N}} = [a_{N11} \quad a_{N21} \quad \ldots \quad a_{NN1}]$, which are the N rows of the event matrix being identified, from the first row to the last row. This is the case regardless which of the two fuzzy inference methods mentioned above is used because the value of $\hat{S}_{jk}^h$ produced by (3) or (4) is identical. By making and applying these particular N pre-event state vectors once, one can completely identify the event matrix simply through observing the post-event state vectors. This holds true regardless of the feeding order of the N pre-event state vectors to the FDES (there are a total of $N!$ different ways of feeding).

The necessity of the condition is obvious - the matrix cannot be completely identified if at least one pre-event state vector in (5) is not fed to the FDES. **QED**

Now let's us deal with the general case and assume that there are M ($M > 1$) fuzzy automata in a single-event SFDES and we do not know them. Nor do we know the value of M and occurrence probabilities of the automata. To identify the event transition matrices, we apply the N pre-event state vectors in (5), $\mathbf{\Theta_1}, \ldots, \mathbf{\Theta_N}$, one at a time orderly to the SFDES. Which fuzzy automaton will produce post-event vector in response to any one of these pre-state vectors is purely random and obeys the occurrence probabilities of the fuzzy automata. We repeatedly feed these state vectors to the SFDES round after round. Sufficiently many rounds of feeding will produce M different $\mathbf{\hat{\Theta}_k^h}$, for any specific k, and they will occur at different probabilities (which is an assumed condition) that resemble the underlying probabilities p_k of the fuzzy automata. First, selecting the N $\mathbf{\hat{\Theta}_k^h}$, $h = 1, 2, ..., N$, that have the same or very similar probability, and then use them to form the event matrix $\tilde{\mathbf{\Psi}}_1$ (its h-th row is the selected $\mathbf{\hat{\Theta}_k^h}$). Repeating this process $M-2$ times will lead to construction/identification of all the remaining $M-1$ fuzzy automata in the SFDES.

Then, we can estimate the occurrence probability of each fuzzy automaton by retrospectively using the same data that has just been employed to determine the M event matrices. From the data, we can determine which post-event state vector is produced by which event matrix. Let f_k^s be the number of post-event state vectors yielded by $\tilde{\mathbf{\Psi}}_k$ after s-th round of the feeding of the pre-event state vectors. We can consequently calculate the corresponding estimated occurrence probability of each event matrix, $\hat{p}_k^s$, by using the following formula:

$$\hat{p}_k^s = \frac{f_k^s}{\sum_{k=1}^{m} f_k^s}, \qquad k = 1, 2, \ldots, M. \tag{6}$$

This probability is an estimation of the underlying occurrence probability. The more rounds of feeding of pre-event state vectors, the closer the computed probability will be to the underlying probability. The model identification is considered successful and the identification process will end once changes of all the estimated occurrence probabilities are less than a modeler-specified error bound ϵ:

$$|\hat{p}_k^s - \hat{p}_k^{s-1}| < \epsilon, \qquad k = 1, 2, \ldots, M. \tag{7}$$

The corresponding s rounds of vector-feeding is designated as $s^*(\epsilon)$, which is a function of ϵ. We use this function-like notation on purpose to explicitly indicate the dependence of the total number of rounds of the vector feeding needed on ϵ. Generally speaking, the smaller the ϵ, the larger the $s^*(\epsilon)$.

Because $\mathbf{\hat{\Theta}_k^h}$ will be the same no matter which one of the two fuzzy inference methods is used, this identification technique works for SFDES with either inference method.

The above identification process can be summarized in the following algorithm.

Algorithm 1: **Input:** $SFDES = \{(G_k, p_k) : k = 1, 2, ..., M\}$, error bound ϵ, minimum number of iteration s_o

Output: M, $\tilde{\mathbf{\Psi}}_k$ and $\hat{p}_k$, $k = 1, 2, ..., M$

```
1:  for h = 1, 2, ..., N do
2:      Θ_h = [0 ... 1 ... 0] (the h-th element = 1);
3:  end for
4:  s = 1;
5:  for h = 1, 2, ..., N do
6:      Input Θ_h to SFDES;
7:      Obtain output Θ̂_s^h from SFDES;
8:  end for
9:  s = s + 1;
10: if s < s_o then
11:     Go to Line 5;
12: else
13:     M = the number of distinct Θ̂_s^h;
14:     Θ̂_k^h = the k-th distinct Θ̂_s^h;
15:     f_k^s = the number of the k-th distinct Θ̂_s^h;
16:     Ψ̃_k = [(Θ̂_k^1)' ... (Θ̂_k^N)']';
17:     p̂_k^s = f_k^s / Σ_{k=1}^{m} f_k^s;
18:     if |p̂_k^s − p̂_k^{s−1}| ≥ ε then
19:         Go to Line 5;
20:     else
21:         p̂_k = p̂_k^s;
22:         Go to Line 25;
23:     end if
24: end if
25: End.
```

An alternative stopping criterion is to do a pre-determined number of rounds of vector feeding regardless whether (7) is met or not.

The theorem below summarizes the results.

Theorem 2: A sufficient condition for identifying $\tilde{\mathbf{\Psi}}_k$ of a single-event SFDES and determining its estimated occurrence probabilities that satisfy a modeler-specified error bound ϵ is $s^*(\epsilon)$ rounds of feeding of the pre-event state vectors (5) to the SFDES. This holds true regardless of which of the two fuzzy inference methods is employed.

Proof: The work presented above establishes the correctness of this conclusion. **QED**

We now extend our investigation to cover the situation when two or more fuzzy automata of a single-event SFDES have an identical occurrence probability. When this is the case, one will become aware of it when selecting N $\mathbf{\hat{\Theta}_k^h}$ to assemble the event matrices involved. For instance, suppose a SFDES consists of six fuzzy automata and two of them occur with an identical occurrence probability. At the stage of deciding which N $\mathbf{\hat{\Theta}_k^h}$ have the same (or very similar) probabilities and hence belong to the same event matrix, one would encounter, for any given h, two different $\mathbf{\hat{\Theta}_k^h}$ that have same (or very similar) probabilities. Thus, there is no way to know for sure which $\mathbf{\hat{\Theta}_k^h}$ is for which of the two event matrices.

Without loss of generality, assume there are m out of M fuzzy automata that have the same occurrence probability ($m \le M$). Following the identifying process mentioned above, we have identified, row by row, a total of $M \times N$ rows of elements, $m \times N$ of which are for the m event matrices. Also, we know which m obtained rows correspond to which specific row of the m event matrices. The only problem is that we do not know how to assign the m rows correctly to the m event matrices, one row for each of them. We now present an approach to matching the rows with the event matrices.

For convenience, we refer arbitrarily the m event matrices as the first, second, ..., m-th event matrix when we consider them sequentially. For the first event matrix, there are m possible assignments for any row vector. Therefore, there are a total of $m \times N$ possible assignment combinations for the N rows in the matrix. For the second event matrix, there are $m-1$ possible assignments for any row vector, and thus a total of $(m-1) \times N$ possible assignment combinations, and so on. Consequently, for the m event matrices, the total number of different row assignment combinations is: $\Omega = m \times N + (m-1) \times N + (m-2) \times N + \cdots + 2N$. Ω is expected to be rather modest because m is usually small. For instance, if $N = 10$ and $m = 3$, $\Omega = 50$.

The matching process proceeds as follows. We first create a series of statistically unbiased random pre-event state vectors. A pre-event state vector is deemed statistically unbiased if each of the N elements of the vector can be treated as a uniform random number in [0, 1]. Otherwise, the vector is said to be statistically biased. A set of state vectors is statistically unbiased if, and only if, all the vectors in the set are statistically unbiased. Whether a given set is biased can be judged. One empirical way is that if the set is unbiased, numbers like 0, 0.5, and 1 (or values close to them) should not be missing and also the average value of the same individual state membership across H pre-event state vectors should be around 0.5. This is a simpler test, and there are more rigorous statistical tests in the literature for this purpose.

We then feed these pre-event state vectors one at a time to the SFDES being identified. The post-event state vector and the corresponding pre-event state vector form a pair as a result. We create many such pairs of pre- and post-event state vectors. By "many," we mean each of the fuzzy automata with an identical probability produces at least one, but preferably multiple, pair of pre- and post-event state vector. We isolate those pairs that are produced by the m fuzzy automata from the rest of the pairs through the differences in the occurrence frequencies. The pairs found will be used as test pairs. We write a computer program that will implement only the fuzzy automata with the identical occurrence probability. We run the program that starts with any one of the Ω combinations of row element assignment. The program will read in the pre-event state vectors of the test pairs one at a time and calculate the post-event state vector. It will then compare the result with the post-event state vector of the test pair. If they are the same, this process will repeat and the program will read in the next pre-event state vector. If they are different, it means this particular combination of row element assignment being tested does not represent the true m fuzzy automata. The program will abandon this combination and use any one of the remaining combinations to start the process again. The program will stop when all the post-event state vectors produced by the event matrices formed by one of the Ω combinations of row element assignment are found to be exactly the same as those in the test pairs. At that point, the m event matrices have been successfully identified.

In what follows, we present this matching process in a mathematically rigorous manner.

Denote the set of all different row assignment combinations as

$$Y = \{(\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i) : i = 1, 2, ..., \Omega\}.$$

Denote the set of statistically unbiased test pairs as

$$X = \{(\mathbf{\Theta}_j, \hat{\mathbf{\Theta}}_j) : j = 1, 2, ..., \Gamma\}. \tag{8}$$

We say that a test pair $x = (\mathbf{\Theta}_j, \hat{\mathbf{\Theta}}_j)$ satisfies a combination $y = (\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i)$ if there exists an event matrix $\tilde{\mathbf{\Psi}}_k^i$ in y such that $\hat{\mathbf{\Theta}}_j = \tilde{\mathbf{\Psi}}_k^i \circ \mathbf{\Theta}_j$. In other words,

$$(\exists \tilde{\mathbf{\Psi}}_k^i \in y) \hat{\mathbf{\Theta}}_j = \mathbf{\Theta}_j \circ \tilde{\mathbf{\Psi}}_k^i.$$

Theorem 3: Assume that $\tilde{\mathbf{\Psi}}_k$ and p_k of a single-event SFDES using either fuzzy inference are to be identified, where $k = 1, 2, \ldots, m$, $p_i = p_j$, and $1 \leq i, j \leq m$. If the test set X is large enough such that there exists one and only one combination $y = (\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i) \in Y$ that is satisfied by all the test pairs $x \in X$, then $\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i$ are the true m event matrices to be identified.

Proof: Assume that the test set X is large enough such that there exists one and only one combination $y = (\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i) \in Y$ that is satisfied by all the test pairs $x \in X$. We prove that $\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i$ are the m event matrices to be identified by contradiction.

Suppose that $\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i$ are not the m event matrices to be identified. Since we assume that all the rows of all the matrices have been correctly identified through the identifying process described before Theorem 2 is introduced and Y contains all the possible combinations, there exists one combination $y' = (\tilde{\mathbf{\Psi}'}_1^i, ..., \tilde{\mathbf{\Psi}'}_m^i) \in Y$ such that $\tilde{\mathbf{\Psi}'}_1^i, ..., \tilde{\mathbf{\Psi}'}_m^i$ are the m event matrices to be identified and $y' \neq y$.

Since $\tilde{\mathbf{\Psi}'}_1^i, ..., \tilde{\mathbf{\Psi}'}_m^i$ are the m event matrices to be identified, it must be satisfied by all the test pairs $x \in X$. Since $y = (\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i) \in Y$ is also satisfied by all test pairs $x \in X$, we have two different combinations, namely y and y', both of which are satisfied by all the test pairs $x \in X$. This contradicts the assumption that there exists one and only one combination $y = (\tilde{\mathbf{\Psi}}_1^i, ..., \tilde{\mathbf{\Psi}}_m^i) \in Y$ that is satisfied by all the test pairs $x \in X$.

The estimated occurrence probability for the identical occurrence probability shared by the m event matrices, denoted as $\hat{p}$, is determined through the mean of the m individual $\hat{p}_k^{s_k}$:

$$\hat{p} = \frac{\sum_{k=1}^{m} \hat{p}_k^{s_k}}{m} \tag{9}$$

where $\hat{p}_k^{s_k}$ is calculated by (6). **QED**

Generally speaking, the larger the m and/or N, the larger the set of test pairs X should be.

We comment that to identify a SFDES containing fuzzy automata with rather similar (i.e., about the same but not identical) occurrence probabilities, one may also use this test-pairs technique to perform the row assignment task. Finally, even if all the fuzzy automata of a SFDES appear to have distinctively different occurrence probabilities, one may still use this technique to further validate the matrices identified through Theorem 2, if desired. In either case, it is an optional step.

Combining Theorems 2 and 3, we arrive at the following conclusion on identification of any single-event SFDES regardless of event occurrence probabilities of its automata:

Corollary 1: A sufficient condition for identifying $\tilde{\mathbf{\Psi}}_k$

and estimating p_k of a single-event SFDES, where $k = 1, 2, \ldots, M$, is performing sufficient rounds of feeding of the pre-event state vectors (5) and applying enough test pairs (8) to the SFDES. This holds true regardless of which of the two fuzzy inference methods is employed.

IV. Numerical Example

Correctness of the Prerequired-Pre-Event-State-Based Technique can be clearly seen and systematically verified in a mathematical and rigorous manner, as shown in the previous section. Thus, simulation is not warranted for this purpose.

Alternatively, we chose to provide a numerical example to concretely show, step-by-step, how the Technique functions. For brevity, we choose to use a hypothetical, but representative, single-event SFDES model that consists of the following three fuzzy automata:

$$\tilde{\mathbf{\Psi}}_1 = \begin{bmatrix} 0.551 & 0.946 & 0.209 \\ 0.857 & 0.184 & 0.439 \\ 0.681 & 0.591 & 0.421 \end{bmatrix}$$

$$\tilde{\mathbf{\Psi}}_2 = \begin{bmatrix} 0.810 & 0.317 & 0.144 \\ 0.416 & 0.321 & 0.869 \\ 0.380 & 0.621 & 0.961 \end{bmatrix}$$

$$\tilde{\mathbf{\Psi}}_3 = \begin{bmatrix} 0.466 & 0.313 & 0.391 \\ 0.756 & 0.183 & 0.838 \\ 0.988 & 0.691 & 0.343 \end{bmatrix}$$

with underlying $p_1 = 0.15$, $p_2 = 0.35$, $p_3 = 0.5$. Either the max-product fuzzy inference (3) or the max-min fuzzy inference (4) can be used; they produce the same result, as explained earlier.

Table I shows in detail how the Technique feeds the three pre-event state vectors in Column 1 to the SFDES randomly one at a time to get the post-event state vectors listed in Column 2, which are actually the rows of the event transition matrices. The Technique then assigns them correctly to the fuzzy automata based on the similar event occurrence frequencies (i.e., f_r^{100} in the table). At the end, the Technique identifies all the three matrices correctly - the matrices identified in the fourth column of the table are exactly the same as the true matrices above.

The last column of the table shows how these frequencies are utilized to estimate the underlying event occurrence probabilities. The results are $p_1 = 0.157$, $p_2 = 0.34$, and $p_3 = 0.503$, which are slightly different from the given underlying probabilities. If more accurate estimations are desired, they can be achieved by feeding each of the three pre-event state vectors more times (e.g., 500 times instead of the 100 times).

V. Conclusion

We introduce a new and innovative class of FDES models called stochastic fuzzy discrete event systems (SFDES), which can be used to model practical systems in various industries that are unsuitable for the PFDES modeling framework. Focusing on single-event SFDES, we study how to identify such a model without any a priori knowledge, and develop the Prerequired-Pre-Event-State-Based Technique. It works with single-event SFDES that uses either the max-product fuzzy inference or the max-min fuzzy inference. Two necessary and sufficient conditions (Theorems 1 and 3) and two sufficient conditions (Theorem 2 and Corollary 1) are established for the identification of single-event SFDES with different settings. A numerical example is provided to more concretely illustrate the Technique.

While single-event SFDES is a stochastic model, the Technique is deterministic and, better yet, does not have any adjustable parameter to set or hyperparameter to experiment with.

References

[1] 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, 1987, pp. 206-230.

[2] F. Lin and W. M. Wonham, “On observability of discrete-event systems,” Information sciences, vol. 44, no. 3, 1988, pp. 173-198.

[3] C.G. Cassandras and S. Lafortune, Introduction to discrete event systems (Second Edition), Springer Science & Business Media, 2008.

[4] W.M. Wonham and K. Cai, Supervisory control of discrete-event systems, Springer Science & Business Media, 2019.

[5] W.M. Wonham, K. Cai, and K. Rudie, “Supervisory control of discrete-event systems: A brief history,” Annual Reviews in Control, vol. 45, no. 1, 2018, pp. 250-256.

[6] G. Klir and B. Yuan, Fuzzy Sets and Fuzzy Logic, Prentice Hall, 1995.

[7] 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, 2002, pp. 408-415.

[8] 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, 2005, pp. 366-371.

[9] 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, 2005, pp. 72-88.

[10] 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, 2009, pp. 578-89.

[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, 2015, pp. 178-192.

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

[13] W. Deng, and D. Qiu, “State-based decentralized diagnosis of bi-fuzzy discrete event systems,” IEEE Transactions on Fuzzy Systems, vol. 25, no. 4, 2017, pp. 854-867.

Pre-event state vectors randomly fed to SFDES 100 times each	Observed post-event state vectors	Number of observations (f_r^{100})	Event transition matrices assembled based on similar magnitudes of three f_r^{100}	Estimated event occurrence probability
[1 0 0]	[0.466 0.313 0.391]	53 (f_1^{100})	$\begin{bmatrix} 0.551 & 0.946 & 0.209 \\ 0.857 & 0.184 & 0.439 \\ 0.681 & 0.591 & 0.421 \end{bmatrix}$	$(f_2^{100} + f_6^{100} + f_8^{100})/300$ = 0.157
	[0.551 0.946 0.209]	14 (f_2^{100})		
	[0.810 0.317 0.144]	33 (f_3^{100})		
[0 1 0]	[0.416 0.321 0.869]	34 (f_4^{100})	$\begin{bmatrix} 0.810 & 0.317 & 0.144 \\ 0.416 & 0.321 & 0.869 \\ 0.380 & 0.621 & 0.961 \end{bmatrix}$	$(f_3^{100} + f_4^{100} + f_9^{100})/300$ = 0.34
	[0.756 0.183 0.838]	50 (f_5^{100})		
	[0.857 0.184 0.439]	16 (f_6^{100})		
[0 0 1]	[0.988 0.691 0.343]	48 (f_7^{100})	$\begin{bmatrix} 0.466 & 0.313 & 0.391 \\ 0.756 & 0.183 & 0.838 \\ 0.988 & 0.691 & 0.343 \end{bmatrix}$	$(f_1^{100} + f_5^{100} + f_7^{100})/300$ = 0.503
	[0.681 0.591 0.421]	17 (f_8^{100})		
	[0.380 0.621 0.961]	35 (f_9^{100})		

TABLE I: The step-by-step process of identifying the event transition matrices of the three fuzzy automata of the example SFDES and estimating their occurrence probabilities.

[14] E. Kilic, “Diagnosability of fuzzy discrete event systems,” Information Sciences, vol. 178, no. 3, 2008, pp. 858-870.

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

[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, 2017, pp. 1830-1836.

[17] 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, 2007, pp. 410-424.

[18] 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, 2010, pp. 951-956.

[19] 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, 2009, pp. 316-328.

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

[21] 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, 2006, pp. 663-676.

[22] 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, 2007, pp. 966-979.

[23] H. Ying, F. Lin, and R. Sherwin, “Fuzzy discrete event systems with gradient-based online learning,” Proceedings of 2019 IEEE International Conference on Fuzzy Systems, New Orleans, LA, USA, June 23-26, 2019.

[24] H. Ying, and F. Lin, “Online Self-learning Fuzzy Discrete Event Systems,” IEEE Transactions on Fuzzy Systems, vol. 9, 2020,pp. 2185-2194.

[25] B. Benmessahel, M. Touahria, F. Nouioua, J. Gaber, P. Lorenz, “Decentralized prognosis of fuzzy discrete-event systems,” Iran. J. Fuzzy Syst. ,vol. 16, no. 3, 2019, pp. 127–143.

[26] Y. Cao, Y. Ezawa, G. Chen, H. Pan, “Modeling and specification of nondeterministic fuzzy discrete-event systems,” In: Ceberio M., Kreinovich V. (eds) Decision Making under Constraints. Studies in Systems, Decision and Control, vol 276. Springer, Cham, 2020.

[27] R. Huq, G.K.I. Mann, R.G. Gosine, “Behavior-modulation technique in mobile robotics using fuzzy discrete event system,” IEEE Trans. Robot. , vol. 22, no. 5, 2006, pp. 903–916.

[28] J.P. Liu, Y.M. Li, “The relationship of controllability between classical and fuzzy discrete-event systems,” Inform. Sci. , vol. 178, no. 21, 2008, pp. 4142–4151.

[29] R. Liu, Y.X. Wang, L. Zhang, “An FDES-based shared control method for asynchronous brain-actuated robot,” IEEE Trans. Cybern. vol. 46, no. 6, 2016, pp. 1452–1462.

[30] M. Nie and W. Tan, “Theory of generalized fuzzy discrete-event systems,” IEEE Transactions on Fuzzy Systems, vol. 23, no. 1, 2015, pp. 98-110.

[31] W. Deng and D. Qiu, “Bi-fuzzy discrete event systems and their supervisory control theory,” IEEE Transactions on Fuzzy Systems, vol. 23, no. 6, 2015, pp. 2107-2121.

[32] B. Benmessahel, M. Touahria, and F. Nouioua, “Predictability of fuzzy discrete event systems,” Discrete Event Dynamic Systems, vol. 27, no. 4, 2017, pp. 641-673.

[33] K. Schmidt and Y. Boutalis, “Fuzzy discrete event systems for multiobjective control: Framework and application to mobile robot navigation,” IEEE Transactions on Fuzzy Systems, vol. 20, no. 5, 2012, pp. 910-922.

[34] A. Jayasiri, G. Mann, and R. Gosine, “Generalizing the decentralized control of fuzzy discrete event systems,” IEEE Transactions on Fuzzy Systems, vol. 20, no. 4, 2012, pp. 699-714.

[35] W. Deng, J. Yang, C. Jiang, and D. Qiu, “Opacity of fuzzy discrete event systems,” Proc. of the 31st Chinese Control And

Decision Conference, 2019, pp. 1846-1851.

[36] D.P. Bertsekas, "Minimax methods based on approximation," in Proceeding of the 1976 Johns Hopkins Conference on Information Sciences and Systems, 1976.

[37] S. Xu, "Smoothing methods for minmax problems," Computational Optimization and Applications, vol. 20, 2001, pp. 267-279.

[38] H. Ying, and F. Lin, "Learning Fuzzy Automaton's Event Transition Matrix When Post-Event State Is Unknown," IEEE Transactions on Cybernetics, in press. doi: 10.1109/TCYB.2020.3026022.

[39] J.-S. Jang, C.-T. Sun, and E. Mizutani, Neuro-Fuzzy and Soft Computing, Prentice Hall, 1997.

[40] N. J. Khan, G. Ahamad, M. Naseem, and Q. R. Khan, "Fuzzy Discrete Event System (FDES): A Survey," In Renewable Power for Sustainable Growth, pp. 531-544. Springer, Singapore, 2021.

[41] Danmei, Li, Weichun Lan, Hui Zhou, and Shihuang Shao, "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.

[42] Danmei, Li, Weichun Lan, Hui Zhou, and Shihuang 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, 2009, pp. 122-9.

[43] F. Lin, and H. Ying, "Supervised Learning of Multi-Event Transition Matrices in Fuzzy Discrete Event Systems," IEEE Transactions on Cybernetics, in press. doi: 10.1109/TCYB.2022.3161664.

[44] 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, 2022, pp. 399-408.

[45] A. Mekki, F. Lin, and H. Ying, "On Detectabilities of Fuzzy Discrete Event Systems," IEEE Transactions on Fuzzy Systems, vol. 30, no.2, 2022, pp. 426-436.

[46] H. Ying, and F. Lin, "Self-Learning Fuzzy Automaton with Input and Output Fuzzy Sets for System Modeling," IEEE Transactions on Emerging Topics in Computational Intelligence, in press. doi: 10.1109/TETCI.2022.3192890.

[47] W. Deng, D. Qiu and J. Yang, "Opacity Measures of Fuzzy Discrete Event Systems," IEEE Transactions on Fuzzy Systems, vol. 29, no. 9, 2021, pp. 2612-2622. [48] Tianxiang Zhu, Fuchun Liu, and Cuntao Xiao, "Reliable fuzzy prognosability of decentralized fuzzy discrete-event systems and verification algorithm," Information Sciences, vol. 627: pp. 220-237, 2023.

[49] 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: Xiong, N., Li, M., Li, K., Xiao, Z., Liao, L., Wang, L. (eds), Advances in Natural Computation, Fuzzy Systems and Knowledge Discovery, ICNC-FSKD 2022, Lecture Notes on Data Engineering and Communications Technologies, vol 153, 2023, Springer, Cham.

[50] S. Raczynski, "The Space of Models, Semi-Discrete Events with Fuzzy Logic," In: Models for Research and Understanding. Simulation Foundations, Methods and Applications, 2022, Springer, Cham.

Hao Ying (S88-M90-SM97-F12) received the B.S. and M.S. degrees in electrical engineering from Donghua University, Shanghai, China, in February 1982 and July 1984, respectively, and the Ph.D. degree in biomedical engineering from the University of Alabama at Birmingham, USA, in 1990. He is a Professor with the Department of Electrical and Computer Engineering, Wayne State University, USA. He has authored one book and coauthored another on fuzzy control, and also published 130 journal papers and over 160 conference papers. His Google h-index is 52 with the total number of citations for all his publications being more than 11,000. He is ranked among top 25 percent of the 100,000 most-cited authors across all 22 scientific fields, which are selected from nearly 7 million scientists worldwide. He is a recipient of the 2023 IEEE CIS Fuzzy Systems Pioneer Award with the citation "For fundamental contributions to model-free fuzzy control theory and its biomedical applications." Professor Ying is an Associate Editor or a Member of Editorial Board for 15 international journals, including the IEEE Transactions on Fuzzy Systems and the IEEE Transactions on Systems, Man, and Cybernetics: Systems. He is a Member of the 2023 Fellows Committee of the IEEE Computational Intelligence Society (was also on the committee in 2020 and 2021), and was on the same committee of the IEEE Systems, Man, Cybernetics Society in 2016, 2017, and 2020. He is a Member of the Fuzzy Systems Technical Committee of the IEEE Computational Intelligence Society for many years. He was a Program Chair/Co-Chair for four international conferences and was a Program/Technical Committee Member for more than 140 international conferences.

Feng Lin (S85-M88-SM07-F09) received his B.Eng. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 1982, and the M.A.Sc. and Ph.D. degrees in electrical engineering from the University of Toronto, Toronto, ON, Canada, in 1984 and 1988, respectively. He was a Post-Doctoral Fellow with Harvard University, Cambridge, MA, USA, from 1987 to 1988. Since 1988, he has been with the Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI, USA, where he is currently a Professor. His current research interests include discrete event systems, hybrid systems, neural networks, robust control, and their applications in alternative energy, biomedical systems, machine learning, and automotive control. He authored a book entitled "Robust Control Design: An Optimal Control Approach" and coauthored a paper that received a George Axelby outstanding paper award from the IEEE Control Systems Society. He was an associate editor of IEEE Transactions on Automatic Control.