Journal of Intelligent & Fuzzy Systems 23 (2012) 129–141
DOI:10.3233/IFS-2012-0501
IOS Press

On modeling of fuzzy hybrid systems

Xinyu Du, Hao Ying* and Feng Lin
Department of Electrical and Computer Engineering, Wayne State University, Detroit, MI, USA

Abstract. A hybrid system is a system containing a mixture of discrete event components and continuous variable components. The existing hybrid system modeling methods are effective to handle crisp cases but can be difficult to represent deterministic uncertainties and subjectivity inherited in many real-world applications. We generalize the crisp hybrid system framework to a fuzzy hybrid system framework by using fuzzy set theory; the latter contains the former as a special case. We utilize fuzzy sets, type-1 and type-2, to capture and represent uncertainties in the hybrid system's states and variables. We develop algorithms to calculate fuzzy states and their transitions and propose a parallel composition method for modeling a (complex) fuzzy hybrid system through composing its components. This new formal, mathematical framework, capable of modeling a hybrid system with fuzzy states and various types of continuous dynamic processes, regardless whether they are available explicitly or implicitly (e.g., fuzzy systems and neural networks), establishes a basis for systematic study of the fuzzy hybrid systems. It can also be employed for computer simulation investigation, analogous to the discrete event simulation methodology. An example fuzzy hybrid system involving fuzzy differential equations as continuous variable component is provided to illustrate the new theory.

Keywords: Hybrid systems, discrete event systems, continuous variable systems, fuzzy sets, type-2 fuzzy sets

1. Introduction

Since 1980 s, discrete event systems have been studied extensively in the literature. Several important properties such as controllability [36] and observability [22] have been studied that plays the key roles in supervisory control of discrete event systems [7, 22–24, 30, 36]. Discrete event systems have been used to model many practical systems such as manufacturing systems, computer systems, communication protocols, database management systems, and robotic systems. These systems have discrete states and events. Their evolutions are described by sequences of events that describe significant qualitative, as opposed to quantitative, changes in the systems. These systems are event-driven, rather than time-driven. They cannot be described by differential or difference equations, as common in continuous variables systems.

Although very useful, discrete events alone are sometimes not sufficient to describe complex systems encountered in the real work, as many real world systems contain both discrete events and continuous variables. Therefore, since 1990, researchers have been studying hybrid systems that contain both discrete events and continuous variables [1]. Not surprisingly, hybrid systems are much more difficult to study. Even some basic properties such as controllability and observability have not been well understood and defined in hybrid systems. Most work accomplished in hybrid systems is on modeling of hybrid systems. Several models have been proposed, include the hybrid automata [1, 2, 13, 16, 33], the hybrid machine [8, 9, 14, 15, 19, 20, 25, 28, 29, 32], the hybrid Petri-net [3], and the hybrid bond-graph [31]. Among them, the hybrid machine model has been used to investigate safety, liveness, and Zenoness of hybrid systems.

*Corresponding author. Hao Ying, Department of Electrical and Computer Engineering, Wayne State University, 5050 Anthony Wayne Drive, Room 3144, Detroit, MI 48202, USA. Tel.: +1 313 577 3738; Fax: +1 313 577 1101; E-mail: hao.ying@wayne.edu.

In the current framework of hybrid systems, values of discrete states and continuous variables of a hybrid machine must be crisp numbers. In many real-world applications, however, such a requirement cannot always be satisfied. For example, when sensors for a state observer are inaccurate, the system's states can become uncertain and cannot be described by precise numbers. In other applications, especially those in biomedicine, states or state variables may not be directly measurable but can be roughly estimated by human through subjective observation (e.g., pain level or extent of headache experienced by a person/patient) [26, 38]. To address this type of problems, fuzzy set theory can be used. Using fuzzy set theory, the system's state (e.g., a patient's health status) can be uncertain and vague even in a deterministic sense. For instance, it is vague when a patient's health is said to be "good". Furthermore, the transition from one state to another is also vague. It is hard to describe how exactly a patient's condition has changed from "good" to "bad". Fuzzy set theory can easily incorporate subjective observation, judgment, and interpretation by a doctor or a patient. Therefore, it is very useful to introduce fuzzy sets in hybrid systems. However, introducing fuzzy sets in hybrid systems significantly increases the complexity of an already complex problem.

In this paper, we undertake the challenging problem of modeling fuzzy hybrid systems. Our approach is inspired by our early work on fuzzy discrete event systems (FDES). In the first paper in FDES [21], we applied fuzzy sets to the traditional DES theory to establish a fuzzy discrete event system theory, which was extended later to a more general theory called the extended fuzzy discrete event system theory [10]. We studied state feedback control of FDES, and investigated optimal control and observability of FDES. Since the publication of [21], the supervisory control, controllability, observability and diagnosability of the FDES have been extensively investigated by other researchers [5, 6, 27, 34, 35, 37] as well.

It is well known that the theory of hybrid systems is much more difficult to develop than the theory of discrete event systems. Similarly, a theory of fuzzy hybrid systems is expected to be much more difficult to establish than the theory of fuzzy discrete event systems [11]. In this paper, we will start with type-1 fuzzy sets and then investigate more difficult type-2 fuzzy sets. We will build elementary fuzzy hybrid machines first and then combine them to form composite fuzzy hybrid machines by the parallel composition. This work will lay a foundation for future investigation of such important issues as (optimal) control of fuzzy hybrid systems, controllability, and observability.

2. Background on hybrid systems and fuzzy discrete event systems

Before modeling fuzzy hybrid systems, we first provide a brief overview of the hybrid machine approach as well as the FDES theory because they serve as the basis for our generalization later. More detailed information is available in [14, 21].

2.1. Hybrid machine approach

In the hybrid machine theory, a hybrid system is modeled by a composite hybrid machine (CHM) which consists of concurrent operation of multiple elementary hybrid machines (EHMs). An EHM is a basic hybrid machine that cannot be decomposed into smaller EHMs any further. An EHM is represented as [14]:

$$EHM = (Q,\ \Sigma,\ D,\ E,\ \Psi,\ (q_0,\ \mathbf{x}_0)) \tag{1}$$

where Q is the set of all discrete states with linguistic labels, and their initial state is $q_0 \in Q$. Σ is the set of all input events and output events, which are also represented by linguistic labels. The input events, denoted by $\underset{\smile}{\sigma}$, are from the environment in which the EHM operates (anything outside the EHM is considered to be in the environment). The output events, designated as $\overset{\frown}{\sigma}$, are actions impacting the environment from the EHM. D is the set of continuous dynamics denoted by d_q where the subscript q signifies that the dynamics is tied with state q. If state q is not associated with any dynamics, then $d_q = \emptyset$. D is described by

$$\begin{aligned} &D = \{d_q = (\mathbf{x}_q,\ \mathbf{y}_q,\ \mathbf{u}_q,\ \mathbf{g}_q,\ \mathbf{h}_q) :\ q \in Q\} \\ &\dot{\mathbf{x}}_q = \mathbf{g}_q\ (\mathbf{x}_q, \mathbf{u}_q) \\ &\mathbf{y}_q = \mathbf{h}_q\ (\mathbf{x}_q, \mathbf{u}_q) \end{aligned} \tag{2}$$

where $\mathbf{x}_q$, $\mathbf{u}_q$, and $\mathbf{y}_q$ are the vectors of states, input variables, and output variables, respectively. Their dimensions can be different from one state to another; therefore the subscript q is needed. $\mathbf{g}_q$ is required to be a Lipschitz continuous function and $\dot{\mathbf{x}}_q = \mathbf{g}_q\ (\mathbf{x}_q, \mathbf{u}_q)$ is a general state space equation. $\mathbf{h}_q$ is a continuous function, and $\mathbf{y}_q = \mathbf{h}_q\ (\mathbf{x}_q, \mathbf{u}_q)$ is a general output equation. Together, the two equations completely describe d_q. We use $\mathbf{x_0} = \mathbf{x}_{q_0}(t_0)$ to define the initial value of the state vector.

E in (1) is a set of state transition paths defined as

$$E = \{e = (q, G \wedge \underset{\smile}{\sigma} \to \widehat{\sigma}, q', \mathbf{x}^0_{q'}) : q, q' \in Q\}$$

where $\wedge$ is the binary logical AND operator. The transition occurs when the system leaves state q and goes to state q'. The transition occurs only when the input event $\underset{\smile}{\sigma}$ is received by the EHM and the conditions set in dynamic guard G are satisfied. G is defined as a Boolean combination of inequalities specifying conditions involving the state variable or output variable in D that represents the system's state (we designate such a variable by $s(t)$, or simply s, whose range is application-dependant and without loss of generality, we assume $s(t) \in \Re$). The arrow above indicates that once the transition occurs the output event $\widehat{\sigma}$ will be transmitted to the environment immediately. Note that whether $\widehat{\sigma}$ exists or not depends on application; it may be absent in some applications. $\mathbf{x^0_{q'}}$ is the initial value for $\mathbf{x_{q'}}$.

Ψ in (1) is a set of so called invariants, each of which is a Boolean combination of inequalities that define a discrete state resulted from $s(t)$ in the inequalities.

A run of an EHM will generate a sequence of states $q_0 \overset{e_1,t_1}{\to} q_1 \overset{e_2,t_2}{\to} q_2 \overset{e_2,t_2}{\to} \cdots$ where $e_i \in E$ is the i-th transition and t_i represents time when the i-th transition occurs.

All the EHMs in a hybrid system run in parallel via the parallel composition mechanism, denoted by $\|$, to form a CHM. The detail on the parallel composition can be found in [14].

2.2. *Fuzzy discrete event systems*

A DES can be regarded as a special hybrid system when the dynamics D is absent (i.e., $d_q = \emptyset$). An FDES is a generalized DES where states are allowed to be fuzzy. To include FDES as part of (1), we first need to formulize the states, events, transitions and initial value in (1) by using a fuzzy finite automaton (FFA). We defined an FFA as [21]:

$$FFA = (Q, \Sigma, \delta, \mathbf{q_0})$$

where Q just like in (1), is the set of states. But here, states are generalized to allow for both crisp states and fuzzy states. In the FFA theory, we employ matrices to represent states and events, which is not the case for the conventional finite automata theory. This need rises because an FDES is allowed to be in more than one state simultaneously. We stick to the following convention in this paper – a symbol representing a vector or a matrix variable is boldfaced. States, fuzzy or not, are represented by a fuzzy state vector $\mathbf{q} = [\mu_1, \mu_2, \ldots, \mu_N] \in Q$ with N being the total number of states (the vector contains 0's and 1's only for a nonfuzzy state). The element $\mu_i \in [0, 1]$ represents the possibility (i.e., membership) of the system being in state i. $\mathbf{q}_0 \in Q$ is the initial fuzzy state vector. An event in Σ does not only have a linguistic label, as defined in (1), but now also has a transition matrix $\sigma = [a_{ij}]_{N\times N}$ associated with it, where $a_{ij} \in [0, 1]$ represents the likelihood (i.e., membership) of the system moving from state i to state j.

$\delta : Q \circ \Sigma \to Q$ is a transition mapping that describes what event may occur at the current time and what the resulting new state is. Here, the symbol $\circ$ represents some fuzzy operation specified by δ. It includes, but not limited to, the popular max-product or max-min fuzzy inference operation in fuzzy set theory [18]. Which operation to use is application-dependent and is up to the system developer. Max-product or max-min is an operation in which the operators max(), product(), and min() are used. Product() means multiplication of the arguments while max() (or min()) operation picks up the largest (or smallest) argument among the arguments involved. If the current fuzzy state vector is $\mathbf{q}$ and event σ occurs, then the new fuzzy state vector $\mathbf{q}'$ can be computed: $\mathbf{q}' = \mathbf{q} \circ \boldsymbol{\sigma}$.

2.3. *Illustrative examples of a hybrid system and an FDES*

To better understand the notations for the hybrid machine approach and FDES, let us use a water tank system consisting of a pump, a tank, and a valve as an example (the system is shown in half of Fig. 1). The system can be modeled by parallel composition of three EHMs named Pump, Tank, and Valve:

$$CHM = Pump||Tank||Valve$$

Each EHM is in the form of (1). For EHM Tank, there are three crisp states on water level with linguistic labels "High", "Medium" and "Low". That is,

$$Q = \{High, Medium, Low\}$$

Assume that the water level is initially "Low", then $q_0 =$ "Low". There are four events, which correspond to operation of turning the Pump "*On*" and "*Off*" and the Valve "*On*" and "*Off*". That is,

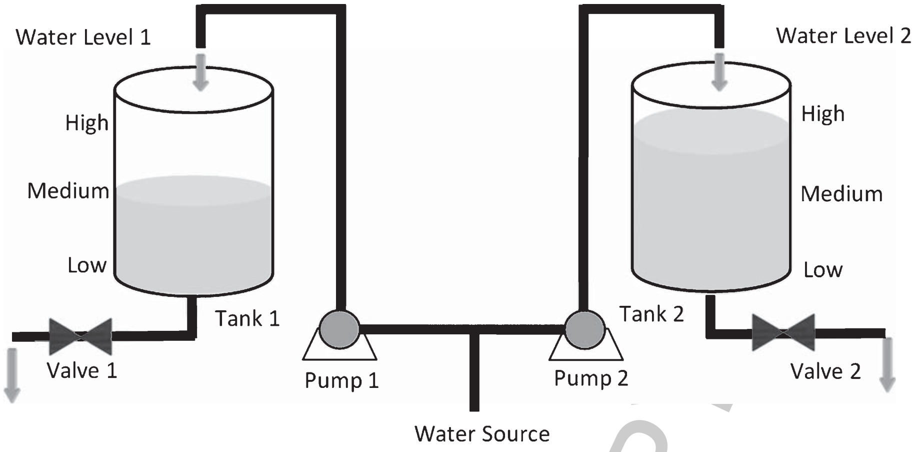

Fig. 1. A two-water-tanks hybrid system.

$$\Sigma = \{pump\ On,\ pump\ Off,\ valve\ On,\ valve\ Off\} \tag{3}$$

There are three continuous dynamics in D, each of which is associated with one of the three discrete states. The state space equation and output equation for the three dynamics are the same:

$$\dot{x} = in - out$$
$$y = x \tag{4}$$

where y is the output variable and x is a state variable, both of which represent the tank's water level. *in* and *out* are input variables representing inflow and outflow water rates, respectively. Note that the first equation is the second equation in (2), and the second equation is the third equation in (2).

Furthermore, Ψ contains three invariants, named ψ_{High}, ψ_{Medium} and ψ_{Low}, one for each of the three states:

$$\psi_{High} = y \geq \lambda_2 \quad \text{(for "High" water level)}$$
$$\psi_{Medium} = y < \lambda_2 \ \wedge \ y \geq \lambda_1 \quad \text{(for "Medium" water level)}$$
$$\psi_{Low} = y < \lambda_1 \quad \text{(for "Low" water level)}$$

where λ_1 and λ_2 are pre-defined water levels.

Assume that there are four dynamic guards as follows:

$G_1 = y < \lambda_2$ (for transition from "High" to "Medium")

$G_2 = y \geq \lambda_2$ (for transition from "Medium" to "High")

$G_3 = y < \lambda_1$ (for transition from "Medium" to "Low")

$G_4 = y \geq \lambda_1$ (for transition from "Low" to "Medium")

A state transition takes place when one of the four input events in (3) happens and the condition set in one of the four dynamic guards is met. The states before and after transition are two of the three states. The EHM Pump has two states:

$$Q = \{On,\ Off\}$$

The event set is $\sum = \{pump\ \ On,\ \ pump\ Off\}$. These two events are input events for the Pump and output events for the Tank. The dynamics are

$in = 0$ for state "*Off*"

$in = \theta$ for state "*On*"

where θ is a specific nonzero pump flow rate. There are no dynamic guard and invariant set for the Pump. Thus, all the state transitions are triggered by the input events alone.

The EHM Valve is similar to the EHM Pump except for different label names and values.

If the dynamic Equation (4) is unknown, the tank system can still be treated either as a DES or an FDES. Let us model the water-tank system as an FDES. The fuzzy state vector will look like

$$\mathbf{q} = \begin{array}{c} \begin{array}{ccc} \scriptstyle High & \scriptstyle Medium & \scriptstyle Low \end{array} \\ [\begin{array}{ccc} 0.1 & 0.1 & 0.8 \end{array}] \end{array} \in Q$$

where the numbers denote the membership grade for each state. The four events are now characterized by four different matrices. For instance, the event *pump On* may be

$$\sigma = \begin{bmatrix} 0.2 & 0.2 & 0.1 \\ 0.9 & 0.2 & 0.1 \\ 0.6 & 0.9 & 0.1 \end{bmatrix}$$

If the max-product operation is used for δ, we can compute the new fuzzy state vector:

$$\mathbf{q}' = \mathbf{q} \circ \boldsymbol{\sigma} = \begin{bmatrix} 0.1 & 0.1 & 0.8 \end{bmatrix} \circ \begin{bmatrix} 0.2 & 0.2 & 0.1 \\ 0.9 & 0.2 & 0.1 \\ 0.6 & 0.9 & 0.1 \end{bmatrix}$$

$$= \begin{bmatrix} 0.09 \; 0.72 \; 0.08 \end{bmatrix}$$

which means after the occurrence of the event *pump On*, the water level is mostly in the "Medium" state, due to the highest membership 0.72 among the three, although it is also in the "High" and "Low" states somewhat.

3. Modeling fuzzy hybrid systems by fuzzy hybrid machines

Based on the theories of the hybrid machines and FDES, we formally define the fuzzy hybrid machines for modeling the fuzzy hybrid systems in this section.

3.1. Developing fuzzy elementary hybrid machines when type-1 fuzzy sets are enough to represent knowledge

3.1.1. Defining the fuzzy elementary hybrid machines

A fuzzy hybrid system is modeled as a fuzzy composite hybrid machine (FCHM) consisting of fuzzy elementary hybrid machines (FEHMs) by a modified parallel composition method. To cover all possible scenarios, we need to divide FEHMs into two types according to state exclusivity. If all the states of an FEHM are nonexclusive (e.g., Tank states "High", "Medium" and "Low"), we call the FEHM the nonexclusive state FEHM. On the other hand, if all the states are exclusive (e.g., Pump states "*On*" and "*Off*"), we call such an FEHM the exclusive state FEHM. We also divide FEHMs into two types based on the dynamic equations - with dynamic equation type and without dynamic equation type (i.e., DES). Hence there are total four types of FEHMs, namely (1) exclusive state with dynamic equations, (2) exclusive state without dynamic equations, (3) nonexclusive state with dynamic equations, and (4) nonexclusive state without dynamic equations.

As a generalized form of EHMs and FFAs, an FEHM contains all the components in EHMs and FFAs and is represented as a 7-tuple:

$$FEHM = (Q,\ \Sigma,\ \delta,\ D,\ E,\ \Psi, (\mathbf{q}_0,\ \mathbf{x}_0)) \tag{5}$$

Much of the meanings of the symbols in (5) are the same as before. We only need to point out the differences and generalizations.

Differences for Q: For the exclusive state type of an FEHM, μ_i in fuzzy state vector $\mathbf{q} = [\mu_1 \ \cdots \ \mu_N] \in Q$ can only be either 0 or 1, and furthermore only exactly one μ_i can be 1 at any instance of time. For the nonexclusive state type, μ_i is a number between 0 and 1 indicating that the system can simultaneously be in different states at any time instance. Unlike FFA, $\mathbf{q}$ is time-dependent when the dynamic equations are present.

Differences for Σ: For an FEHM without dynamic equations, $\boldsymbol{\sigma} = [a_{ij}]_{N\times N}$ represents the transition result. For an FEHM with dynamic equations, a fuzzy state vector is calculated directly from state variables in the model without using a fuzzy event matrix. To make the notation consistent with that for the FEHM without dynamic equations, we let the fuzzy event matrix be the identical matrix $\mathbf{I}$ in this case so that the occurrence of an event will not affect the fuzzy state vector.

Differences for D: All the variables can now be fuzzy numbers (i.e., fuzzy sets) instead of real numbers. The subscript $\mathbf{q}$ of d is boldfaced to indicate the fact that $\mathbf{q}$ is a vector, instead of a scalar in the case of (2). For an FEHM without dynamic equations, D is empty by definition. For an FEHM with dynamic equations, there are two cases. If the states are exclusive, $d_{\mathbf{q}}$ can be either the same or different from one another. On the other hand, if the states are nonexclusive, the FEHM can be in multiple states simultaneously. Hence, all $d_{\mathbf{q}}$ have to be the same (otherwise, we will be unable to determine which dynamic equation is currently running). For an FEHM, $\mathbf{g}_q$ and $\mathbf{h}_q$ are allowed to be either explicitly available or implicitly available. An example for the latter category is a typical neural network, which is considered a black-box model in that it provides the desired input-output mapping but fails to reveal explicitly the mathematical relation.

Differences for E: Only the transition of an FEHM with dynamic equations is described by E. For an

FEHM without dynamic equations, the transition is defined by δ. For a nonexclusive state FEHM, there will be no inequalities in the dynamic guard G because the system can be in more than one state with different membership grades at any time. Therefore, G is always true. For the same reason, there will be no inequalities in the invariant set Ψ since we don't need it to define states. In contrast, for an FEHM with exclusive state, G and Ψ are needed. They however cannot be fuzzified owing to the fact that the states are exclusive.

Like an EHM, a run of an FEHM will produce a sequence of states $\mathbf{q}_0 \xrightarrow{e_1,t_1} \mathbf{q}_1 \xrightarrow{e_2,t_2} \mathbf{q}_2 \xrightarrow{e_3,t_3} \ldots$

An FEHM can have two forms of input, input events $\breve{\boldsymbol{\sigma}}$ and input variables $\mathbf{u}$, and three forms of output, output events $\widehat{\boldsymbol{\sigma}}$, output state vector $\mathbf{q}$, and output variables $\mathbf{y}$.

From the definition of an FEHM, we can easily prove that the DES, FDES, and continuous variable dynamic system models, traditional or fuzzy (e.g., a Mamdani fuzzy model), are special cases of the FEHM. This is because when there is no dynamic equation in an FEHM, D, E and all the time-dependant variables disappear. At that time, (5) becomes an automaton for DES or FDES:

$$FEHM = (Q,\ \Sigma,\ \delta,\ \mathbf{q}_0) = FFA.$$

On the other hand, when there is no event and discrete state in an FEHM, Q, Σ, E, and δ will cease to exist, and the FEHM will reduce to a continuous variable dynamic system:

$$\begin{aligned} FEHM &= (d,\ \mathbf{x}_0) \\ &= (\mathbf{x}(t),\ \mathbf{y}(t),\ \mathbf{u}(t),\ \mathbf{g}(\mathbf{x},\mathbf{u}),\ \mathbf{h}(\mathbf{x},\mathbf{u}),\ \mathbf{x}_0) \end{aligned}$$

Finally, if we restrict the membership grades in the fuzzy state vector to be binary (i.e., 0 or 1), the FEHM will become an EHM.

3.1.2. *Computing fuzzy state vectors and state transition matrices*

For an exclusive state FEHM, the membership grade of a state, 0 or 1, is directly obtained from the current system status. For instance, if the Pump is "*On*" at present, the fuzzy state vector is $\mathbf{q} = \begin{array}{c} \begin{array}{cc} \scriptstyle On & \scriptstyle Off \end{array} \\ [\begin{array}{cc} 1 & 0 \end{array}] \end{array}$. For a nonexclusive state FEHM, the membership grade for state, μ_i, is calculated using the state variable $s(t)$ and the fuzzy set F_i pre-defined for state i. In FEHM, the value of $s(t)$ can be either a crisp number or a fuzzy number. Loosely speaking, a fuzzy number is just a fuzzy set. Due to the current status of fuzzy theory, a fuzzy number here is necessarily restricted to satisfy the following mild requirements [18]: (1) it is a normal, continuous fuzzy set (i.e., its height must be 1), (2) its support set is bounded, (3) its α-cut set is a closed interval, and (4) the support set of the strong 0-cut set is bounded. These requirements eliminate those fuzzy sets that do not make any practical sense (and hence never used in the literature any way). Since a crisp number is a special case of a fuzzy number, to be general, we assume the value of $s(t)$ to be fuzzy number H. The membership grade of μ_i is calculated as follows:

$$\mu_i = \sup_s(\gamma_{F_i \cap H}(s)) = \sup_s(\min(f_i(s),\ h(s)\ ))\quad (6)$$

where $f_i(s)$ is the membership function of F_i, $h(s)$ is the membership function of H, $\gamma_{F_i \cap H}$ represents the resulting membership function of $F_i \cap H$ produced by the Zadeh fuzzy set intersection operation $\cap$ (i.e., min()) and sup() is the operation to select the maximum membership grade for all s. To carry out (6), the α-cut set approach [12] must be first employed to represent the fuzzy sets and results are:

$$f_i(s) = \max_{\alpha \in [0,\ 1]} \alpha \otimes\ [\underline{f}_i^{-1}(\alpha),\ \overline{f}_i^{-1}(\alpha)],$$

$$h(s) = \max_{\alpha \in [0,\ 1]} \alpha \otimes [\underline{h}^{-1}(\alpha),\ \overline{h}^{-1}(\alpha)]$$

where $[\underline{f}_i^{-1}(\alpha),\ \overline{f}_i^{-1}(\alpha)]$ and $[\underline{h}^{-1}(\alpha),\ \overline{h}^{-1}(\alpha)]$ are the α-cut sets of $f_i(s)$ and $h(s)$, respectively, and $0 \le \alpha \le 1$. The α-cut sets are intervals (i.e., interval numbers) containing all the values of $s(t)$ whose membership grades are greater than or equal to α. Here, $\overline{f}_i^{-1}(\alpha)$, $\underline{f}_i^{-1}(\alpha)$, $\overline{h}^{-1}(\alpha)$, and $\underline{h}^{-1}(\alpha)$ are calculated from the inverse functions of $\overline{f}_i$, $\underline{f}_i$, $\overline{h}$ and $\underline{h}$, respectively, where $\overline{f}_i$ and $\underline{f}_i$ (or $\overline{h}$ and $\underline{h}$) are respectively the monotonic increasing portion and monotonic decreasing portion of $f_i(s)$ (or $h(s)$). $\alpha \otimes\ [*]$ means that the membership grades in $[*]$ are set at the level of α.

The calculations in (6) will then be performed as follows. When $f_i(s)$ does not intersect with $h(s)$ (i.e., when $\overline{f}^{-1}(0) < \underline{h}^{-1}(0)$ or $\underline{f}^{-1}(0) > \overline{h}^{-1}(0)$), $\mu_i = 0$. When $f_i(s)$ and $h(s)$ intersect, the highest membership grade of their intersection is the result for (6). At that point, we have either $\overline{f}^{-1}(\alpha) = \underline{h}^{-1}(\alpha)$ or $\underline{f}^{-1}(\alpha) = \overline{h}^{-1}(\alpha)$. Thus, $\mu_i = \alpha$. As a special case, if $h(s)$ is a singleton fuzzy set (i.e., its membership grade is 1 at only one value of $s(t)$ and 0 elsewhere), then we have a simple result for (6): $\mu_i = f_i(s)$. For better understanding of the above steps, Fig. 2 shows the notations and the outcome of (6) when fuzzy sets of our arbitrary choices are used.

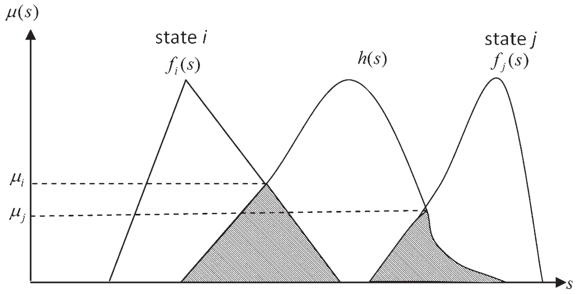

Fig. 2. An illustration of notations and operations in (6) using arbitrary type-1 fuzzy numbers.

We use (6) to determine a_{ij} in the state transition matrix in a similar fashion. The only difference is that $f_i(s)$ is replaced by $f_{ij}(s)$, which is a pre-defined fuzzy number indicating the vague possibility for the FEHM to transfer from state i to state j.

3.2. Developing fuzzy elementary hybrid machines when type-2 fuzzy sets are necessary to represent part of knowledge

In some real-world applications, it may be impossible to obtain a fuzzy state vector or a fuzzy event matrix using type-1 fuzzy sets (two-dimensional membership functions) for some state or state transition and hence type-2 fuzzy sets with three-dimensional membership functions (the additional dimension is the membership of membership) must be employed because (1) a domain expert may fail to accurately describe a state or state transition with a single type-1 fuzzy set and may prefer to use a bundle of type-1 fuzzy sets instead, (2) when a group of experts with distinct opinions is involved, they should not be forced to reach a consensus type-1 fuzzy set just for the sake of ease of the system development, and (3) the initial condition or parameters in the dynamic equations may be best described by a bundle of type-1 fuzzy numbers instead of single one. In EFHM, letting the type-1 fuzzy number F_i and H in (6) be type-2 fuzzy sets which are denoted as $\tilde{F}_i$ and $\tilde{H}$ (in this paper, we place a $\sim$ on an upper-case letter to designate a type-2 fuzzy set and to differentiate it from its type-1 counterpart), respectively, we can extend the FEHMs in Section 3.1 to deal with these issues. More specifically, let $\tilde{F}_i$ and $\tilde{H}$ be

$$\tilde{F}_i = \{(s, y),\ \varphi_{\tilde{F}_i}(s, y) | \forall s \in \Re,\ \forall y \in \cup\mu_{\tilde{F}_i}\},$$
$$\tilde{H} = \{(s, y),\ \varphi_{\tilde{H}}(s, y) | \forall s \in \Re,\ \forall y \in \cup\mu_{\tilde{H}}\}$$

where y is the primary membership grade of s and $\varphi_{\tilde{F}_i}$ and $\varphi_{\tilde{H}}$ represent the secondary membership functions of the type-2 fuzzy sets. The relationship between each value of s and y is an internal $\mu_{\tilde{F}_i}$ (i.e., primary memberships). The union of all the primary memberships forms the FOU (footprint of uncertainty) denoted by $\cup\mu_{\tilde{F}_i}$. The collections of all the left and right terminal points of the intervals form respectively the upper and lower primary membership functions. As an example, for $\tilde{F}_i$, the upper membership function $\overline{\mu}_{\tilde{F}_i}(s) = \cup\max(\mu_{\tilde{F}_i}(s))$ and $\underline{\mu}_{\tilde{F}_i}(s) = \cup\min(\mu_{\tilde{F}_i}(s))$. When the type-2 fuzzy sets are utilized, the result of (6) becomes a type-1 fuzzy number, denoted as V_i, instead of the numerical number μ_i . The universe of discourse of V_i is the membership grade indicating the possibilities that the system is in state i. With this generalization, all the variables and parameters in the FEHM can be represented by type-1 fuzzy numbers, a general form of real numbers. More concretely, μ_i and a_{ij} in Q and Σ are replaced by V_i and A_{ij} (type-1 fuzzy number), respectively. Also, all the variables and parameters in $d_{\mathbf{q}}$ are now allowed to be numerical numbers, type-1 fuzzy sets or type-2 fuzzy sets, depending on application.

Accordingly, the operations in Sections 3.1.2 and 3.1.3 need to change to type-1 fuzzy set operations. For δ, the max, min and product operations are replaced respectively by type-1 fuzzy number operations $\max_S$, $\min_S$ and $product_S$, which are implemented via the α -cut operations [18]:

$$\begin{aligned}\max_S(V_i, A_{ij}) &= \max_S\Big(\max_{\alpha\in[0,\,1]} \alpha\otimes[\underline{v}_i^{-1}(\alpha), \overline{v}_i^{-1}(\alpha)],\\ &\qquad \max_{\alpha\in[0,\,1]} \alpha\otimes[\underline{a}_i^{-1}(\alpha), \overline{a}_i^{-1}(\alpha)]\Big)\\ &= \max_{\alpha\in[0,\,1]} \otimes[\max(\underline{v}_i^{-1}(\alpha), \underline{a}_i^{-1}(\alpha)),\\ &\qquad \max(\overline{v}_i^{-1}(\alpha), \overline{a}_i^{-1}(\alpha))]\end{aligned}$$

$$\begin{aligned}\min_S(V_i, A_{ij}) &= \min_S\Big(\max_{\alpha\in[0,\,1]} \alpha\otimes[\underline{v}_i^{-1}(\alpha), \overline{v}_i^{-1}(\alpha)],\\ &\qquad \max_{\alpha\in[0,\,1]} \alpha\otimes[\underline{a}_i^{-1}(\alpha), \overline{a}_i^{-1}(\alpha)]\Big)\\ &= \max_{\alpha\in[0,\,1]} \alpha\otimes[\min(\underline{v}_i^{-1}(\alpha), \underline{a}_i^{-1}(\alpha)),\\ &\qquad \min(\overline{v}_i^{-1}(\alpha), \overline{a}_i^{-1}(\alpha))]\end{aligned}$$

$$\begin{aligned}&product_S(V_i, A_{ij})\\ &= product_S\Big(\max_{\alpha\in[0,\,1]} \alpha\otimes[\underline{v}_i^{-1}(\alpha), \overline{v}_i^{-1}(\alpha)],\end{aligned}$$

$$\max_{\alpha\in[0,\ 1]} \alpha \otimes [\underline{a}_i^{-1}(\alpha), \overline{a}_i^{-1}(\alpha)])$$
$$= \max_{\alpha\in[0,\ 1]} \alpha \otimes [\underline{v}_i^{-1}(\alpha) \times \underline{a}_i^{-1}(\alpha), \overline{v}_i^{-1}(\alpha) \times \overline{a}_i^{-1}(\alpha)]$$

where the lower terminal points $\underline{v}_i^{-1}$ and $\underline{a}_i^{-1}$ and the upper terminal points, $\overline{v}_i^{-1}$ and $\overline{a}_i^{-1}$ are calculated respectively from the inverse functions of $\underline{v}_i$, $\underline{a}_i$, $\overline{v}_i$, and $\overline{a}_i$, which are the monotonic increasing parts (for $\underline{v}_i$ and $\underline{a}_i$) and monotonic decreasing parts (for $\overline{v}_i$, and $\overline{a}_i$) of the membership functions of V_I and A_{ij}, respectively.

Equation (6) is changed to become the following:

$$V_i = \left\{y,\ \mu_{V_i}(y)|\mu_{V_i}(y) = \sup_y \left(\varphi_{\tilde{F}_i\cap\tilde{H}}(s, y)\right),\ \forall y \in \sup_s \left(\mu_{\tilde{F}_i\cap\tilde{H}}\right)\right\}$$

where $\mu_{V_i}(y)$ is the membership function of V_i. The sup() operation on $\mu_{\tilde{F}_i\cap\tilde{H}}$ results in an internal, $\left[\max\left(\underline{\mu}_{\tilde{F}_i\cap\tilde{H}}\right),\ \max\left(\overline{\mu}_{\tilde{F}_i\cap\tilde{H}}\right)\right]$ whereas the other sup() is to find the maximum secondary membership grades to form $\varphi_{\tilde{F}_i\cap\tilde{H}}(s, y)$ for each y value. Here, $\cap$ means the type-2 fuzzy set intersection [17] carried out as below to produce a type-2 fuzzy set $\tilde{T}$:

$$\begin{aligned}\tilde{T} = \tilde{F}_i \cap \tilde{H} &= \{\varphi_{\tilde{T}}(s, y)|\varphi_{\tilde{T}}(s, y)\\ &= \sup_y(\min(\varphi_{\tilde{F}_i}(s, y_{\tilde{F}_i}), \varphi_{\tilde{H}}(s, y_{\tilde{H}}))),\ \forall s \in \Re,\\ y &= \min(y_{\tilde{F}_i},\ y_{\tilde{H}}),\ \forall y_{\tilde{F}_i} \in \cup\mu_{\tilde{F}_i} \subseteq [0,\ 1],\\ &\forall\ y_{\tilde{H}} \in \cup\mu_{\tilde{H}} \subseteq [0,\ 1]\}\end{aligned}$$

where $y_{\tilde{F}_i}$ and $y_{\tilde{H}}$ are the primary membership of $\tilde{F}_i$ and $\tilde{H}$, respectively. The primary membership grade of $\tilde{T}$ is y (i.e.,$\min(y_{\tilde{F}_i},\ y_{\tilde{H}})$) and the secondary membership grade is $\min(\varphi_{\tilde{F}_i}(s, y_{\tilde{F}_i}), \varphi_{\tilde{H}}(s, y_{\tilde{H}}))$. If $\min(\varphi_{\tilde{F}_i}(s, y_{\tilde{F}_i}), \varphi_{\tilde{H}}(s, y_{\tilde{H}}))$ generates more than one value for the same value of y, the largest value will be selected as the result by the sup() operation.

3.3. Parallel composition of fuzzy elementary hybrid machines to form a fuzzy composite hybrid machine

An FCHM can be synthesized by parallel composition of more than one simpler FEHM:

$$FCHM = FEHM_1||FEHM_2||\cdots||FEHM_n$$

where n is the number of (simpler) FEHMs and the symbol $\|$ represents a parallel composition operation that we will develop now on the basis of the FDES parallel composition [21] Assuming that the i-th FEHM has N_i states, we define the composite state set as $Q = Q^1 \times Q^2 \times \cdots \times Q^n$ represented by the composite fuzzy state vector with $N_1 \times N_2 \times \cdots \times N_n$ elements (i.e., composite states) in it. The membership grade for a composite state is the minimum of the membership grades of all the n states of the n FEHMs, that is,

$\mu_{i_1,i_2,\ldots,i_n} = \min(\mu^1_{i_1},\ \mu^2_{i_2},\ \ldots,\ \mu^n_{i_n})$ for the FEHM involving type-1 fuzzy sets only

$V_{i_1,i_2,\ldots,i_n} = \min_S(V^1_{i_1},\ V^2_{i_2},\ \ldots,\ V^n_{i_n})$ for the FEHM involving type-2 fuzzy sets

We define the composite event set as $\Sigma = \Sigma^1 \cup \Sigma^2 \cup \cdots \cup \Sigma^n$ where $\cup$ is the crisp set union operation. The composite event matrix σ remains to be the same as the one for the FDES [21] and the extended FDES [10], which means its elements to be computed as follows: for the FEHM involving type-1 fuzzy sets only

$$a_{(i,j),(u,v)} = \begin{cases} \min(a^k_{iu},\ a^l_{jv}), & \text{if } \boldsymbol{\sigma} \text{ is an event in both FEHM}_k \text{ and FEHM}_l \\ a^k_{iu}, & \text{if } \boldsymbol{\sigma} \text{ is not an event in FEHM}_k \text{ and } j = v \\ a^l_{jv}, & \text{if } \boldsymbol{\sigma} \text{ is not an event in FEHM}_l \text{ and } i = u \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

and for the FEHM involving type-2 fuzzy sets

$$A_{(i,j),(u,v)} = \begin{cases} \min_S(A^k_{iu},\ A^l_{jv}), & \text{if } \boldsymbol{\sigma} \text{ is an event in both FEHM}_k \text{ and FEHM}_l \\ A^k_{iu}, & \text{if } \boldsymbol{\sigma} \text{ is not an event in FEHM}_k \text{ and } j = v \\ A^l_{jv}, & \text{if } \boldsymbol{\sigma} \text{ is not an event in FEHM}_l \text{ and } i = u \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

where k and l index to two FEHMs being composited (i.e., FEHM$_k$ and FEHM$_l$) while i, j, u, and v index their states. Note that an event of an FEHM with dynamic equations can be defined as an identity matrix. Thus, our parallel composition method can handle composition of FEHMs with or without dynamic system components.

We define the composite dynamic system set D as:

$$\begin{aligned} D &= \{d_{\mathbf{q}} = (\mathbf{x_q}, \mathbf{y_q}, \mathbf{u_q}, \mathbf{g_q}, \mathbf{h_q}) : \\ &\quad \mathbf{q} \in Q^1 \times Q^2 \times \cdots \times Q^n\} \\ \dot{\mathbf{x}}_{\mathbf{q}} &= \mathbf{g_q}(\mathbf{x_q}, \mathbf{u_q}) \\ \mathbf{y_q} &= \mathbf{h_q}(\mathbf{x_q}, \mathbf{u_q}) \end{aligned}$$

where

$$\begin{aligned} \mathbf{x_q} &= [\mathbf{x}^1_{\mathbf{q}_{j_1}}, \mathbf{x}^2_{\mathbf{q}_{j_2}}, \ldots, \mathbf{x}^n_{\mathbf{q}_{jn}}], \\ \mathbf{s} &= [s_1, s_2, \ldots, s_n], \\ \mathbf{u_q} &= [\mathbf{u}^1_{\mathbf{q}_{j_1}}, \mathbf{u}^2_{\mathbf{q}_{j2}}, \ldots, \mathbf{u}^n_{\mathbf{q}_{jn}}], \\ \mathbf{y_q} &= [\mathbf{y}^1_{\mathbf{q}_{j_1}}, \mathbf{y}^2_{\mathbf{q}_{j2}}, \ldots, \mathbf{y}^n_{\mathbf{q}_{jn}}], \end{aligned}$$

$$\mathbf{g_q} = \begin{bmatrix} \mathbf{g}^1_{\mathbf{q}_{j_1}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{g}^n_{\mathbf{q}_{jn}} \end{bmatrix}, \quad \mathbf{h_q} = \begin{bmatrix} \mathbf{h}^1_{\mathbf{q}_{j_1}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \ddots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{h}^n_{\mathbf{q}_{jn}} \end{bmatrix}$$

The notations of all the components of $d_{\mathbf{q}}$ are the same as those for a single FEHM, and their meanings are similar. Except for s, the subscript $\mathbf{q}_{j_i}$ and superscript i indicate that the element corresponds to the j_i-th fuzzy state vector of the i-th FEHM.

Parallel composition of two exclusive state FEHMs will yield an exclusive state FCHM whose composite states' membership grades will be 0 or 1. Composition of two nonexclusive state FEHMs will produce a nonexclusive state FCHM with the membership grades being between 0 and 1. Parallel composition of an exclusive state FEHM and a nonexclusive state FEHM will lead to a mixed type of FCHM whose membership grades are in [0, 1]. Note that in this case, the composite states corresponding to the states from the exclusive-state FEHM whose membership grades are 0 will have 0 membership grade indicating that the likelihood of this composite state is 0. For an FEHM without dynamic equations, there will be no D. Therefore, we do not need to count it in when carrying out the parallel composition operation. In summary, the parallel composition of two FEHMs will result in an FCHM that either belongs to one of the four types associated with the FEHMs or one of the two new types - mixture of exclusive and nonexclusive states with dynamic equations and mixture of exclusive and nonexclusive states without dynamic equations. These six types cover common fuzzy hybrid systems encountered in real-world applications.

4. An illustrative example – modeling the water tank system as a fuzzy hybrid system

For the complete two-tank system shown in Fig. 1, suppose that one's goal is to maintain the water level in each tank at the pre-specified level by turning on and off the two valves and two pumps intermittently and in turn. At any moment, only one valve and one pump can stay on and they must not belong to the same tank. This is a classical hybrid system that has been well studied under the following conditions [16]: (1) the water levels, pump flow rates, and valve flow rates are all precisely known through measurement all the time, (2) the dynamic equations and their parameter values are explicitly known, and (3) the target water levels are precisely given. Our objective for the example is significantly more challenging - how to still attain the control goal when none of the conditions holds. More specifically, we suppose that (1) the water levels and the four flow rates can only be estimated through human observation without using any measurement instrument, (2) the dynamic equations are fuzzy differential equations with their variables and parameters being type-1 fuzzy numbers, and (3) the target water levels are described by type-1 fuzzy numbers. These kinds of scenarios are not hypothetical; rather they represent practical obstacles often encountered in complex real-world applications (e.g., costs or hardware restrictions). With the new assumptions, the system becomes a fuzzy hybrid system.

4.1. The fuzzy hybrid machine model

There are total six FEHMs – two for Pumps, two for Tanks and two for Valves, where the Pumps and the Valves are exclusive state FEHMs with dynamic equations and the Tanks are nonexclusive state FEHMs with dynamic equations. All of them can be modeled using (5). For Pump1 and Pump2, $Q^1 = \{\mathbf{q}^1_1 = \begin{array}{cc} \scriptstyle pump1\ On & \scriptstyle pump1\ Off \\ [\,0 & 1\,] \end{array}, \mathbf{q}^1_2 = \begin{array}{cc} \scriptstyle pump1\ On & \scriptstyle pump1\ Off \\ [\,1 & 0\,] \end{array}\}$, $Q^2 = \{\mathbf{q}^2_1 = \begin{array}{cc} \scriptstyle pump2\ On & \scriptstyle pump2\ Off \\ [\,0 & 1\,] \end{array}, \mathbf{q}^2_2 = \begin{array}{cc} \scriptstyle pump2\ On & \scriptstyle pump2\ Off \\ [\,1 & 0\,] \end{array}\}$. $\sum^1 = \{\boldsymbol{\sigma}^1_1 = \boldsymbol{\sigma}^1_2 = \mathbf{I}_{2\times 2}\}$

where events $\boldsymbol{\sigma}^1_1$ and $\boldsymbol{\sigma}^1_2$ are "pump1 *On*" and "pump1 *Off*", respectively. $\mathbf{I}_{2\times 2}$ is the 2 by 2 identical matrix. Likewise, $\sum^2 = \{\boldsymbol{\sigma}^2_1 = \boldsymbol{\sigma}^2_2 = \mathbf{I}_{2\times 2}\}$ where the events are "pump2 *On*" and "pump2 *Off*". δ is not needed since all the six FEHMs are with dynamic equations D and transitions E. $D^1 = \{d^1_{\mathbf{q}^1_1} : IN_1 = 0,\ d^1_{\mathbf{q}^1_2} : IN_1 = \Gamma_1\}$, $D^2 = \{d^2_{\mathbf{q}^2_1} : IN_2 = 0,\ d^2_{\mathbf{q}^2_2} : IN_2 = \Gamma_2\}$. We use IN_1 and IN_2, instead of in_1 and in_2, to signify they are now type-1 fuzzy variables. Γ_1 and Γ_2 are constant pump flow rates characterized by type-1 fuzzy numbers. $E^1 = \{e^1_1 : (\mathbf{q}^1_1, \underset{\smile 1}{\boldsymbol{\sigma}^1}, d^1_{\mathbf{q}^1_1} \to \mathbf{q}^1_2, \overset{\frown 1}{\boldsymbol{\sigma}}_1,\ d^1_{\mathbf{q}^1_2}),\quad e^1_2 : (\mathbf{q}^1_2, \underset{\smile 2}{\boldsymbol{\sigma}^1}, d^1_{\mathbf{q}^1_2} \to \mathbf{q}^1_1, \overset{\frown 1}{\boldsymbol{\sigma}}_2,\ d^1_{\mathbf{q}^1_1})\}$ $\underset{\smile 1}{\boldsymbol{\sigma}^1} = \overset{\frown 1}{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}^1_1$, $\underset{\smile 2}{\boldsymbol{\sigma}^1} = \overset{\frown 1}{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}^1_2$, and $E^2 = \{e^2_1 : (\mathbf{q}^2_1, \underset{\smile 1}{\boldsymbol{\sigma}^2}, d^2_{\mathbf{q}^2_1} \to \mathbf{q}^2_2, \overset{\frown 2}{\boldsymbol{\sigma}}_1,\ d^2_{\mathbf{q}^2_2})$, $e^2_2 : (\mathbf{q}^2_2, \underset{\smile 2}{\boldsymbol{\sigma}^2}, d^2_{\mathbf{q}^2_2} \to \mathbf{q}^2_1, \overset{\frown 2}{\boldsymbol{\sigma}}_2,\ d^2_{\mathbf{q}^2_1})\}$,$\underset{\smile 1}{\boldsymbol{\sigma}^2} = \overset{\frown 2}{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}^2_1$, and $\underset{\smile 2}{\boldsymbol{\sigma}^2} = \overset{\frown 2}{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}^2_2$. As before, we use the low bracket and high bracket to indicate input and output, respectively.

For Valve1 and Valve2, $Q^3 = \{\mathbf{q}^3_1 = \begin{array}{cc} \scriptstyle valve1\ On & \scriptstyle valve1\ Off \\ [\,0 & 1\,] \end{array}, \mathbf{q}^3_2 = \begin{array}{cc} \scriptstyle vavle1\ On & \scriptstyle valve1\ Off \\ [\,1 & 0\,] \end{array}\}$, $Q^4 = \{\mathbf{q}^4_1 = \begin{array}{cc} \scriptstyle valve2\ On & \scriptstyle valve2\ Off \\ [\,0 & 1\,] \end{array},\ \mathbf{q}^4_2 = \begin{array}{cc} \scriptstyle valve2\ On & \scriptstyle valve2\ Off \\ [\,1 & 0\,] \end{array}\}$. $\sum^3 = \{\boldsymbol{\sigma}^3_1 = \boldsymbol{\sigma}^3_2 = \mathbf{I}_{2\times 2}\}$, $\sum^4 = \{\boldsymbol{\sigma}^4_1 = \boldsymbol{\sigma}^4_2 = \mathbf{I}_{2\times 2}\}$ with the events being listed above the vectors. $D^3 = \{d^3_{\mathbf{q}^3_1} : OUT_1 = 0,\ d^3_{\mathbf{q}^3_2} : OUT_1 = \Delta_1\}$, $D^4 = \{d^4_{\mathbf{q}^4_1} :\ OUT_2 = 0,\ \ d^4_{\mathbf{q}^4_2} : OUT_2 = \Delta_2\}$, where Δ_1 and Δ_2 are constant valve flow rates described by type-1 fuzzy numbers and OUT_1 and OUT_2 are type-1 fuzzy variables (its crisp counterpart is *out* in (4)). $E^3 = \{e^3_1 : (\mathbf{q}^3_1, \underset{\smile 1}{\boldsymbol{\sigma}^3}, d^3_{\mathbf{q}^3_1} \to \mathbf{q}^3_2, \overset{\frown 3}{\boldsymbol{\sigma}}_1,\ d^3_{\mathbf{q}^3_2}),\ e^1_2 : (\mathbf{q}^3_2, \underset{\smile 2}{\boldsymbol{\sigma}^3}, d^3_{\mathbf{q}^3_2} \to \mathbf{q}^3_1, \overset{\frown 3}{\boldsymbol{\sigma}}_2,\ d^3_{\mathbf{q}^3_1})\}$ where $\underset{\smile 1}{\boldsymbol{\sigma}^3} = \overset{\frown 3}{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}^3_1$ and $\underset{\smile 2}{\boldsymbol{\sigma}^3} = \overset{\frown 3}{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}^3_2$; $E^4 = \{e^4_1 : (\mathbf{q}^4_1, \underset{\smile 1}{\boldsymbol{\sigma}^4}, d^4_{\mathbf{q}^4_1} \to \mathbf{q}^4_2, \overset{\frown 4}{\boldsymbol{\sigma}}_1,\ d^4_{\mathbf{q}^4_2}),\ e^4_2 : (\mathbf{q}^4_2, \underset{\smile 2}{\boldsymbol{\sigma}^4}, d^4_{\mathbf{q}^4_2} \to \mathbf{q}^4_1, \overset{\frown 4}{\boldsymbol{\sigma}}_2,\ d^4_{\mathbf{q}^4_1})\}$, $\underset{\smile 1}{\boldsymbol{\sigma}^4} = \overset{\frown 4}{\boldsymbol{\sigma}}_1 = \boldsymbol{\sigma}^4_1$ and $\underset{\smile 2}{\boldsymbol{\sigma}^4} = \overset{\frown 4}{\boldsymbol{\sigma}}_2 = \boldsymbol{\sigma}^4_2$.

For Tank1 and Tank2, we consider two scenarios for the fuzzy state vectors. In the first scenario, a type-1 fuzzy set is employed to characterize state membership grade (see Section 3.1). Therefore, $Q^5 = \{\mathbf{q}^5_{j_5} = \left[\mu^5_{i_5}\right]\}$, $Q^6 = \{\mathbf{q}^6_{j_6} = \left[\mu^6_{i_6}\right]\}$ where j_5 and j_6 are non-negative integers, $i_5, i_6 \in \{1, 2, 3\}$, and $\mu^5_{i_5},\ \mu^6_{i_6} \in [0, 1]$. Each state vector contains three states linguistically labeled as High, Medium and Low denoted by F_1, F_2 and F_3, respectively.

In the second scenario, the state membership grade is a fuzzy number, instead of a numerical number, leading to $Q^5 = \{\mathbf{q}^5_{j_5} = \left[V^5_{i_5}\right]\}$ and $Q^6 = \{\mathbf{q}^6_{j_6} = \left[V^6_{i_6}\right]\}$ where $V^5_{i_5}$ and $V^6_{i_6}$ are type-1 fuzzy numbers. The set of all the input events and output events is

$$\sum^5 = \{\boldsymbol{\sigma}^5_{i_5} \in \sum^1 \cup \sum^3,\ i_5 \in \{1, 2, 3, 4\}\},$$

$$\sum^6 = \{\boldsymbol{\sigma}^6_{i_6} \in \sum^2 \cup \sum^4,\ i_6 \in \{1, 2, 3, 4\}\}.$$

The continuous dynamics (see (4)) are now represented by fuzzy differential equations:

$$D^5 = \left\{ d^5 = \left\{ \begin{array}{l} \dot{X}_1 = IN_1 - OUT_1 \\ Y_1 = X_1 \end{array} \right\} \right.,$$

$$D^6 = \left\{ d^6 = \left\{ \begin{array}{l} \dot{X}_2 = IN_2 - OUT_2 \\ Y_2 = X_2 \end{array} \right\} \right.,$$

where X_1, X_2, Y_1 and Y_2 are type-1 fuzzy variables as indicated by the upper-case letters. The state transitions are

$$E^5 = \{e^5 : (\mathbf{q}^5_{j_5}, \underset{\smile i_5}{\boldsymbol{\sigma}^5} \to \mathbf{q}^5_{j'_5})\},$$

$$E^6 = \{e^6 : (\mathbf{q}^6_{j_6}, \underset{\smile i_6}{\boldsymbol{\sigma}^6} \to \mathbf{q}^6_{j'_6})\}$$

where $\mathbf{q}^5_{j_5}, \mathbf{q}^5_{j'_5} \in Q^5$, $\underset{\smile i_5}{\boldsymbol{\sigma}^5} = \boldsymbol{\sigma}^5_{i_5} \in \sum^5$ and $\mathbf{q}^6_{j_6}, \mathbf{q}^6_{j'_6} \in Q^6$, $\underset{\smile i_6}{\boldsymbol{\sigma}^6} = \boldsymbol{\sigma}^6_{i_6} \in \sum^6$. There are no invariant and dynamic guard.

The next step is to apply the parallel composition to the six FEHMs to form the FCHM model for the entire system: $FCHM = Pump1\ ||Pump2\ ||\ Valve1\ ||\ Valve2\ ||\ Tank1||\ Tank2$ where $Q = Q^1 \times Q^2 \times \cdots \times Q^6$. Each composite state vector contains $2 \times 2 \times 2 \times 2 \times 3 \times 3 = 144$ states. The state vector space is infinite because the state vector spaces for both Q^5 and Q^6 are infinite. Note that $\sum = \sum^1 \cup \sum^2 \cup \sum^3 \cup \sum^4 \cup \sum^5 \cup \sum^6 = \sum^5 \cup \sum^6$. Since all the matrices in the FEHMs are identical matrices, the composite event matrices are 144 by 144 identical matrices as a result of the operations in (7) or (8), depending on which scenario is involved.

Note that $\mathbf{X} = [IN_1,\ IN_2,\ OUT_1,\ OUT_2, X_1,\ X_2]$, $\mathbf{S} = \mathbf{Y} = [Y_1,\ Y_2]$.

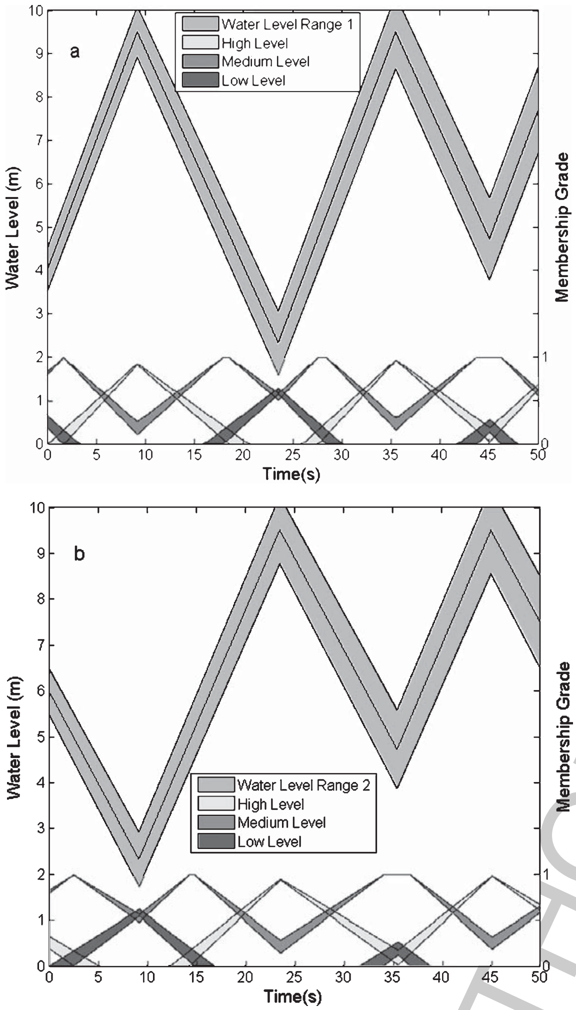

Fig. 3. Matlab® simulation results for the water tank fuzzy hybrid system ((a) Tank1 and (b) Tank2) when all the variables and parameters are type-1 fuzzy numbers and the pre-defined fuzzy sets in (6) are interval type-2 fuzzy sets. The solid curves in the shadows indicate the defuzzified water levels.

For **g**, we have $2^4 = 16$ possibilities. For instance, the combination of pump1 is "*On*", pump2 is "*Off*", valve1 is "*On*" and valve2 is "*Off*" is one of the 16. $\mathbf{g_q}$ is a 6×6 matrix with all the elements being 0 except for the one corresponding to the 5th row and 5th column being $\Gamma_1 - \Delta_1$. As for **h**, $\mathbf{h_q} = \begin{bmatrix} X_1 & 0 \\ 0 & X_2 \end{bmatrix}$.

4.2. Behavior of the fuzzy hybrid machine model

To examine the behavior of the model, we introduce a simple control scheme: when the water level in Tank1 is higher than ξ_1, Pump1 will be turned "*Off*" and Valve1 will be turned "*On*" to decrease the water level. At the same time Pump2 will be switched "*On*" and Valve2 will be switched "*Off*" to increase the water level in Tank2. Likewise, when the water level in Tank2 is more than ξ_2, Pump1 will be "*On*", Valve1 will be "*Off*", Pump2 will be "*Off*" and Valve2 will be "*On*". ξ_1 and ξ_2 are the crisp numbers representing the defuzzified values of the type-1 fuzzy numbers describing the unaided human observations of the water levels.

Table 1
The type-1 fuzzy numbers and the upper and lower primary membership functions of the interval type-2 fuzzy numbers used in the simulation study are assumed to be the following symmetrical triangular function: $\mu(x) = 0$ when $x < a$ or $x > c$; $\mu(x) = \frac{x-a}{b-a}$ when $x \geq a$; $\mu(x) = \frac{c-x}{c-b}$ when $x \leq c$. The parameter values are given in the table

Fuzzy number	a	b	c
Γ_1	0.49	0.5	0.51
Γ_2	0.39	0.4	0.41
Δ_1	0.59	0.6	0.61
Δ_2	0.49	0.5	0.51
X_{10}	3.5	4	4.5
X_{20}	5.5	6	6.5
$\overline{\mu}_{\tilde{F}_1} of \tilde{F}_1$	4.5	10	10
$\underline{\mu}_{\tilde{F}_1} of \tilde{F}_1$	5.5	10	10
$\overline{\mu}_{\tilde{F}_2} of \tilde{F}_2$	0	5	10
$\underline{\mu}_{\tilde{F}_2} of \tilde{F}_2$	0.5	5	9.5
$\overline{\mu}_{\tilde{F}_3} of \tilde{F}_3$	0	0	5.5
$\underline{\mu}_{\tilde{F}_3} of \tilde{F}_3$	0	0	4.5

Based on [4] along with the use of the α-cut set method, we were able to solve the fuzzy differential equations and produced a closed-form solution by using symmetric fuzzy numbers and the centroid defuzzifier. The derivation was lengthy and the solution was messy. They were hence not included here due to constraint of the space.

We developed a MATLAB program to simulate the behavior of the FHM model. Fig. 3 shows the simulation result under the following conditions: (1) the tank water level is limited to [0, 10] and $\xi_1 = \xi_2 = 9.5$ m; (2) the tank system parameters are symmetrical triangular fuzzy numbers, type-1 and interval type-2, as defined in Table 1, and (3) the centroid defuzzifier is employed. The water levels as well as the primary membership functions of the type-2 fuzzy sets describing the fuzzy states are shown in the same figure. One observes that at any instance of time, the exact water levels are unknown, but their ranges and fuzzy states can be precisely determined, demonstrating the power and practical utility of the proposed FHM modeling methodology, which is unachievable by the traditional crisp hybrid machine approach.

5. Conclusion

We have proposed a theoretical framework for modeling the fuzzy hybrid systems, thus generalizing the framework of the crisp hybrid machines to that of the fuzzy hybrid machines. The new framework contains the crisp hybrid systems, the DES, and the FDES as special cases. Not only can the new framework handle DES with fuzzy states and fuzzy transitions but also continuous variable systems whose internal mathematical structures are explicitly unavailable (e.g., fuzzy systems and neural networks) or the state variables are fuzzy (e.g., fuzzy differential equations). An α-cut set-based method has been developed to calculate the fuzzy state vector in the fuzzy hybrid machine. We have also extended the parallel composition method for the crisp hybrid machines to the fuzzy hybrid machines. As an illustrative example, a two-water-tank system is modeled in detail using the new theory and the model behavior is examined in computer simulation.

The new framework makes it possible to systematically investigate the behavior of a fuzzy hybrid system theoretically or in computer simulation and design a safe or optimal control scheme for it. In a sense, its role is the same as that played by the state space model in the linear dynamical system theory or the role of the discrete event automata in the DES theory. A logic next step is to explore control design, system controllability or observability for a class of (simple) fuzzy hybrid systems involving only linear dynamic systems.

Acknowledgment

This research is supported in part by NSF under grant ECS-0823865 and NIH under grant 1R01DA022730.

References

[1] R. Alur and D. Dill, Automata for modeling real-time systems, in: *Proceedings of Proceedings of the 17th International Colloquium on Automata, Languages and Programming*, (1990), 322–336.

[2] R. Alur, C. Courcoubetis, T.A. Henzinger and P.-H. Ho, Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems, in: *Hybrid Systems, Lecture Notes in Computer Science. 736: Springer Berlin/Heidelberg*, 1993, pp. 209–229.

[3] A. Bourjij, M. Zasadzinski, M. Darouach, G. Krzakala and M. Musset, On the use of hybrid Petri nets for control process safety: Application to a steam-boilers network simulator, in: *Proceedings of International Conference on Systems, Man and Cybenetics*, Le Touquet, France, (1993), 197–202.

[4] J.J. Buckley and T. Feuring, Fuzzy differential equation, *Fuzzy Sets and Systems* **110** (2000), 43–54.

[5] Y. Cao and M. Ying, Supervisory control of fuzzy discrete event systems, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* **35** (2005), 366–371.

[6] 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* **37** (2007), 410–424.

[7] C.G. Cassandras and S. Lafortune, *Introduction to discrete event systems*, Kluwer, Norwell, MA, 1999.

[8] C.G. Cassandras, D.L. Pepyne and Y. Wardi, Optimal control of a class of hybrid systems, *IEEE Transactions on Automatic Control* **46** (2001), 398–415.

[9] C.G. Cassandras, From discrete event to hybrid systems, in: *Proceedings of Sixth International Workshop on Discrete Event Systems*, Zaragoza, Spain, (2002), 3–8.

[10] X. Du, H. Ying and F. Lin, Theory of extended fuzzy discrete event system for handling ranges of knowledge uncertainties and subjectivity, *IEEE Transactions on Fuzzy Systems* **17** (2009), 316–328.

[11] X. Du, H. Ying and F. Lin, Fuzzy hybrid systems modeling, in: *Proc Annu North Ame Fuzzy Inf Process Soc Conf*, Toronto, ON, July, 2010.

[12] S. Heilpern, Representation and application of fuzzy numbers, *Fuzzy Sets and Systems* **91** (1997), 259–268.

[13] T.A. Henzinger, P. Kopke, A. Puri and P. Varaiya, What's decidable about hybrid automata, *Journal of Computer and System Sciences* **57** (1998), 94–124.

[14] M. Heymann, F. Lin and G. Meyer, Synthesis and viability of minimally interventive legal controllers for hybrid systems, *Discrete Event Dynamic Systems: Theory and Applications* **8** (1998), 105–135.

[15] M. Heymann, F. Lin, G. Meyer and S. Resmerita, Analysis of Zeno behaviors in hybrid systems, *IEEE Transactions on Automatic Control* **50** (2005), 376–384.

[16] K.H. Johansson, J. Lygeros, S. Sastry and M. Egerstedt, Simulation of Zeno hybrid automata, in: *Proceedings of the 38th IEEE Conference on Decision and Control*, Phoenix, AZ, (1999), 3538–3543.

[17] N.N. Karnik and J.M. Mendel, Operations on type-2 fuzzy sets, *Fuzzy Sets and Systems* **122** (2001), 327–348.

[18] G.J. Klir and B. Yuan, *Fuzzy sets and fuzzy logic: Theory and applications*, Prentice Hall PTR, Upper Saddle River, 1995.

[19] X.D. Koutsoukos, P.J. Antsaklis, J.A. Stiver and M.D. Lemmon, Supervisory control of hybrid systems, in: *Proceedings of the IEEE*, (2000), 1026–1049.

[20] G. Labinaz, M.M. Bayoumi and K. Rudie, A survey of modeling and control of hybrid systems, *International Federation of Automatic Control* **21** (1997), 79–92.

[21] F. Lin and H. Ying, Modeling and control of fuzzy discrete event systems, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* **32** (2002), 408–415.

[22] F. Lin and W.M. Wonham, On observability of discrete event systems, *Information Sciences* **44** (1988), 173–198.

[23] F. Lin and W.M. Wonham, Decentralized supervisory control of discrete-event systems, *Information Sciences* **44** (1988), 199–224.

[24] F. Lin and X. Chen, Estimation of transitional probabilities of discrete event systems from cross-sectional survey and its application in tobacco control, *Information Sciences* **180** (2010), 432–440.

[25] F. Lin, Theorem of abstraction for equivalent controllers in hybrid systems, *Information Sciences* **173** (2005), 181–195.

[26] 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* **177** (2007), 3749–3763.
[27] F.C. Liu and D.W. Qiu, Diagnosability of Fuzzy Discrete Event Systems: A Fuzzy Approach, *IEEE Transactions on Fuzzy Systems* **17** (2009), 372–384.
[28] J. Lygeros, C. Tomlin and S. Sastry, Controllers for reachability specifications for hybrid systems, *Automatica* **35** (1999), 349–370.
[29] T. Moor and J. Raisch, Supervisory control of hybrid systems within a behavioural framwork, *System & Control Letters* **38** (1999), 157–166.
[30] H. Mortazavian and F. Lin, Decentralized supervisory control of discrete event systems with nonhomogeneous control structure, *Information Sciences* **68** (1993), 233–246.
[31] P.J. Mosterman and G. Biswas, A theory of discontinuities in physical system models, *Journal of the Franklin Institute* **335B** (1998), 401–439.
[32] K.M. Passino and U. Ozguner, Modeling and analysis of hybrid systems: Examples, in: *Proceedings of IEEE International Symposium on Intelligent Control*, Arlington, Virginia, USA, (1991), 251–256.
[33] A. Puri and P. Varaiya, Modeling and verification of hybrid systems, in: *Proceedings of American control conference, seattle*, Washington, USA, (1995), 4466–4470.
[34] D. Qiu, Supervisory control of fuzzy discrete event systems: A formal approach, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, **35** (2005), 72–88.
[35] D. Qiu and F. Liu, Fuzzy discrete event systems under fuzzy observability and a test-algorithm, *IEEE Transactions on Fuzzy Systems* **17** (2009), 578–589.
[36] P. Ramadge and W.M. Wonham, Supervisory control of a class of discrete event processes, *SIAM J Control and Optimization* **25** (1987), 206–230.
[37] F. Wang, Z. Feng and P. Jiang, Reliable decentralized supervisory control of fuzzy discrete event systems, *Fuzzy Sets and Systems* **161** (2010), 1657–1668.
[38] 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* **10** (2006), 663–676.