A Process Algebra Approach to Fuzzy Reasoning

Liliana D'Errico   Michele Loreti

Dipartimento di Sistemi e Informatica
Università di Firenze, Italy
Email: {liliana.derrico, michele.loreti}@unifi.it

Abstract— *Fuzzy systems address the imprecision of the input and output variables, which formally describe notions like "rather warm" or "pretty cold", while provide a behaviour that depends on fuzzy data. This class of systems are classically represented by means of Fuzzy Inference Systems (FIS), a computing framework based on the concepts of fuzzy if-then rules and fuzzy reasoning. Even if FIS are largely used, these lack in compositionality. Moreover, the analysis of modeled behaviuors needs complex analytic tools. In this paper we propose a process algebraic approach to specification and analysis of fuzzy behaviours. Indeed, we introduce a Fuzzy variant of CCS (*Calculus of Communicating Processes*), that permits compositionally describing fuzzy behaviours. Moreover, we also show how standard process algebra formal tools, like modal logics and behavioural equivalences, can be used for supporting fuzzy reasoning.*

Keywords— Fuzzy Systems, Process Algebras, Compositional Fuzzy Reasoning

1 Introduction

Human perception of real world abounds with concepts without strictly defined constraints (examples are fat, very, more, slowly, old, etc). Such concepts can be described by means of Fuzzy Sets [1, 2]: classes of objects in which transit from membership to not membership gradually takes place.

Fuzzy Sets are widely used in control systems where the system behaviour can depend on data without precise values. In a *Fuzzy System* input values are described by means of variables that model data (temperature, speed) representing them as fuzzy sets, each of which identifies a different range of values (e.g. *cold*, *warm*, *hot*,...).

These systems are classically represented by means of Fuzzy Inference Systems (FIS) [3], a computing framework based on the concepts of fuzzy if-then rules, fuzzy set theory and fuzzy reasoning. A Fuzzy If-Then-Rule is a rule of the form *If x is A then y is B* where A and B are linguistic values defined by fuzzy sets on universes of discourse X and Y, respectively. Fuzzy reasoning is an inference procedure used to derive conclusions from a set of fuzzy If-Then-Rules and one or more conditions.

Defuzzification is the essential "process" to translate the fuzzy result in a crisp result. The most frequently used defuzzification strategy is the centroid of area defined as:

$$z_{coa} = \frac{\int_z \mu_{C'}(z)\, z\, dz}{\int_z \mu_{C'}(z)\, dz}$$

where $\mu_{C'}(z)$ is the aggregated output membership function. Other defuzzification strategies arise for specific applications: maximum membership, mean of maximum, largest of maximum, smallest of maximum, weighted average and so on. Generally speaking, these defuzzification methods are computation intensive Theoretical results are available [4, 5]. Even if Fuzzy Inference Systems are largely used, these lack in compositionality, in the sense that the rules interactions as well as how a rule interferes with the others is not completely clear. Moreover, it is difficult to compare different implementations of a system as well as verifying the properties satisfied by a given specification.

Process algebras are a set of mathematically rigorous languages with well defined semantics that permit modelling behaviour of concurrent and communicating systems. Verification of concurrent systems within the process algebraic approach can be performed by checking that processes enjoy properties described by some temporal logic's formulae.

In this paper we propose a process algebra approach to specification and analysis of fuzzy behaviours. Indeed, we introduce a Fuzzy variant of CCS (*Calculus of Communicating Processes*), that permits compositionally describing fuzzy behaviours. Operational semantics of Fuzzy CCS (FCCS) is described by means of *Fuzzy Labelled Transition Systems* [6] (FLTS). These are an extension of Labelled Transition Systems where fuzziness is used for modeling imprecision in concurrent systems.

Moreover, we also show how standard process algebra formal tools can be used for supporting fuzzy reasoning. Indeed, we define a fuzzy behavioural equivalence and a fuzzy modal logic that can be used for specifying and verifying properties of fuzzy systems.

The rest of the paper is organised as follows. In Section 2 we introduce Process Algebras conceptually. In Section 3 we recall the basic notions related to L-Fuzzy Sets while in Section 4 we present the Fuzzy CCS. In Section 5 we show how Fuzzy Hennessy-Milner Logic can be used for specifying and verifying properties of Fuzzy Systems. Finally, Section 6 concludes the paper.

2 Process Algebras

Process algebras are a set of mathematically rigorous languages with well defined semantics that permits describing and verifying properties of concurrent communicating systems. They can be seen as mathematical models of processes, regarded as agents that act and interact continuously with other similar agents and with their common environment. The agents may be real-word objects (even people), or they may be artefacts, embodied perhaps in computer hardware or software systems.

Process algebras provide a number of constructors for system description and are equipped with an operational semantics that describes systems evolution. Moreover, they often

ISBN: 978-989-95079-6-8

come equipped with observational mechanisms that permit identifying (through behavioural equivalences) those systems that cannot be taken apart by external observations. In some cases, process algebras have also complete axiomatizations, that capture the relevant identifications.

There has been a huge amount of research work on process algebras carried out during the last 25 years that started with the introduction of CCS [7], CSP [8] and ACP [9].

The main ingredients of a specific process algebra are:

- a minimal set of well thought operators capturing the relevant aspect of systems behaviour and the way systems are composed;
- a transition system associated with the algebra via structural *operational semantics* to describe the evolution of all systems that can be built from the operators;
- an equivalence notion that permits abstracting from irrelevant details of systems descriptions.

Verification of concurrent systems within the process algebraic approach is performed either by resorting to behavioural equivalences or by checking that processes enjoy properties described by some temporal logic's formulae. Equivalences are used for proving conformance of a process to specifications, expressed within the same process algebra notation. Verification with logical formulae is implemented by "model checking", an automatic method to prove properties verification.

In the former case two descriptions of a given system, one very detailed and close to the actual concurrent implementation, the other more abstract describing the abstract tree of relevant actions the system has to perform, are provided and tested for equivalence.

In the latter case, concurrent systems are specified as terms of a process description language while properties are specified as temporal logic formulae. Labelled Transition Systems are associated with terms via a set of structural operational semantics rules and model checking is used to determine whether the transition system associated with those terms enjoys the property specified by the given formulae.

Process algebras and modal logics have been largely used as tools for specifying and verifying properties of concurrent systems. This, also thanks to model checking algorithms, permits verifying whether a given specification satisfies the expected properties.

3 L-Fuzzy Sets

Human perception of real world abounds with concepts without strictly defined constraints (*fat*, *very*, *more*, *slowly*, *old*, etc…). Such concepts can be described by means of Fuzzy Sets: classes of objects in which transit from membership to not membership gradually takes place. A fuzzy set is a simple and intuitive generalization of the classical *crisp*[1] one.

Fuzzy sets are denoted by means of a generalised *membership function* that gives the *membership degree* of each element of the *universe*. This degree usually takes values in $[0, 1]$, the interval of real numbers from 0 to 1 inclusive. Although above range is the one most commonly used for representing membership degrees, any arbitrary set with some natural full or partial ordering can be used. Elements of this set are not required to be numbers as long as the ordering among them can be interpreted as representing various strengths of membership degree [10].

Let U be a universal set and L be a *complete lattice*, a *L-Fuzzy Set* [2, 11] A is denoted by a membership function $\mu_A : U \to L$.

Standard operations on sets and lattices L, like complement, intersection and union, can be generalised to L-Fuzzy Sets. These operations rely on the use of three function $c(\cdot)$, $i(\cdot,\cdot)$ and $u(\cdot,\cdot)$ that, respectively, give the measure of complement, intersection and union of fuzzy degrees.

The complement of a L-Fuzzy Set A, denoted by $\overline{A}$, is specified by a function $c : L \to L$ which assigns a value $\mu_{\overline{A}}(x) = c(\mu_A(x))$ to each membership degree $\mu_A(x)$. This assigned value is interpreted as the membership degree of the element x in the L-Fuzzy Set representing the negation of the concept represented by A.

Intersection and union of two L-Fuzzy Sets A and B are defined using functions $i : L \times L \to L$ and $u : L \times L \to L$. For each element a in the universal set U, these functions take as argument the pair consisting of the membership degrees of a in A and in B, respectively. Function i, also named *t-norm*, yields the membership degree of a in $A \cap B$, while u, also named *t-conorm*, returns the membership degree of a in $A \cup B$. Thus, $\mu_{A\cap B}(a) = i(\mu_A(a), \mu_B(a))$ while $\mu_{A\cup B}(a) = u(\mu_A(a), \mu_B(a))$.

Function c, i and u operating on L-Fuzzy Sets must be *continuous* on L and satisfy all axioms in Table 1, where 0 and 1 denote respectively the least and the greatest element in L while $\leq$ denotes the partial ordering on L. In the rest of this paper we will use $\mathcal{L}$ to denote a tuple $\langle L, c, i, u\rangle$ containing a complete lattice L together with its complement, intersection and union functions used for defining a family of L-Fuzzy Sets.

Table 1: Axioms of fuzzy operations

Axioms for $c(\cdot)$

$$c(1) = 0 \qquad c(0) = 1 \qquad \frac{x \leq y}{c(y) \leq c(x)} \qquad c(c(x)) = x$$

Axioms for $i(\cdot,\cdot)$

$$i(1,1) = 1 \qquad i(0,x) = 0 \qquad i(1,x) = x \qquad i(x,y) = i(y,x)$$

$$\frac{x_1 \leq x_2 \quad y_1 \leq y_2}{i(x_1,y_1) \leq i(x_2,y_2)} \qquad i(i(x,y),z) = i(x, i(y,z))$$

Axioms for $u(\cdot,\cdot)$

$$u(0,0) = 0 \qquad u(0,x) = x \qquad u(1,x) = 1 \qquad u(x,y) = u(y,x)$$

$$\frac{x_1 \leq x_2 \quad y_1 \leq y_2}{u(x_1,y_1) \leq u(x_2,y_2)} \qquad u(u(x,y),z) = u(x, u(y,z))$$

Fuzzy set theory was initially formulated by considering the

[1] Crisp set is defined to split individuals belonging to a certain universe into two groups: members (who surely belong to the set) and not members (who surely do not belong).

ISBN: 978-989-95079-6-8

complete lattice $[0,1]$, denoting the interval of real numbers from 0 to 1 (inclusive), and the following complement, intersection and union functions:

$$c\,(x) = 1 - x \quad i\,(x,y) = \min[x,y] \quad u\,(x,y) = \max[x,y]$$

It is easy to prove that these functions are continuous on $[0,1]$ and satisfy the axioms of Table 1.

4 A Fuzzy Process Algebra

Fuzzy systems are used for handling behaviours that depend on data without precise values that are described by means of variables taking values between 0 and 1. Variables, like temperature or speed, are represented by means of fuzzy sets each of which identifies a different range of values (e.g. *cold*, *warm*, *hot*,...).

As an example, we can consider a *Temperature Control System* (TCS). This is composed of an air conditioner (AC) system and a temperature sensor installed in the room. The TCS regulates the AC power, according to the values read from the sensor, to guarantee a suitable room temperature.

Following the fuzzy approach, room temperature can be modeled by considering different states identifying different range of values. For instance, *cold*, *warm* and *hot* like in figure 1.

Behaviour of TCS is described by a set of rules like: *if the temperature is warm then slightly increase the AC power*, *if the temperature is hot then increase the AC power*.

These systems are classically represented by means of Fuzzy Inference Systems (FIS).

In this section we present a fuzzy variant of CCS (*Calculus of Communicating Processes* [12]), named *Fuzzy CCS* (FCCS). In the new calculus the standard CCS actions are enriched with a fuzzy value modeling the enabling-degree. We aim at defining a formal language that permits compositionally describing fuzzy systems and that can be used for supporting fuzzy reasoning. This, also thanks to the use of standard formal tools like modal logics.

Like other process algebrae, FCCS provides a set of operators that permit describing the complete system starting from the specification of its subcomponents. Following the CCS approach, components interact with each other by means of *actions*, atomic and not interruptible steps, which represent input/output operations on communication ports or internal computations of the system. Let Λ be an infinite numerable set of labels or ports, not containing τ. A FCCS action can be: $a \in \Lambda$, the action to receive a signal on port a; $\bar{a}$ with $a \in \Lambda$, the action to deliver a signal on port a; τ, an internal computation step. We assume $\bar{\bar{\alpha}} \triangleq \alpha$, where $\alpha \in \Lambda \cup \{\bar{a} \mid a \in \Lambda\} \cup \{\tau\}$. Actions $\bar{\alpha}$ and α are said complementary, they represent input and output actions on the same channel.

The fundamental difference from CCS is the introduction of an attribute "x_i", that we define *action execution degree*. It is a fuzzy value able to represent, on a quality level, action behaviour. By this way CCS actions become fuzzy and therefore more representative of real world.

We define FCCS sintax by means of the following grammar:

$$Q \quad ::= \quad nil \mid X \mid \sum_{i \in I} (act_i,\, x_i).Q_i$$

$$P \quad ::= \quad Q \mid P_1 \mid P_2 \mid P \backslash A \mid P\,[f]$$

$$act \quad ::= \quad \bar{a} \mid a \mid \tau$$

Now a brief description of operators:

- nil is the *inactive* process.
- X is the *process constant*, if $X \triangleq P$ then X denotes the invocation of process P. It is useful in defining recursive processes.
- $\sum_{i \in I} (act_i,\, x_i).Q_i$ is the *choice or sum* operator and denotes a choice among i possible behaviours that evolve with action act_i and degree x_i.
- $P_1 \mid P_2$ is the *parallel composition* operator and represents the concurrent execution of processes P_1 e P_2. If during the composition two complementary actions match, the resulting composed action is the internal one τ.
- $P \backslash A$ is the *restriction* operator. $A \in \Lambda$ and $P \backslash A$ behaves like P exception made for the impossibility to interact using actions in A.
- $P\,[f]$ is the *relabelling* operator. $f : \Lambda \rightarrow \Lambda$ allows relabelling process actions, to ease the description of complex processes.

4.1 Fuzzy Operational Semantics

Operational semantics of FCCS processes is defined in term of Fuzzy Labelled Transition Systems [6]. These generalize Labelled Transition Systems by defining the transition relation in term of a L-Fuzzy Set. This approach permits modeling situations like: "the transition takes place *rarely*" or "the transition occurs *frequently*" which may be distinguished and treated as a consequence.

Definition 4.1 ($\mathcal{L}$-FLTS) *Let $\mathcal{L} = \langle L, c, i, u \rangle$, a Fuzzy Labelled Transition System $\mathcal{F}$ for $\mathcal{L}$ ($\mathcal{L}$-FLTS) is a tuple $\langle Q, A, \chi_{\rightarrow} \rangle$ where:*

- *Q is a set whose elements are called states*
- *A is a finite set whose elements represent actions*
- *$\chi_{\rightarrow} : (Q \times A \times Q) \rightarrow L$ is the total membership function.*

We will write $q_0 \xrightarrow{\alpha}_{\varepsilon} q_1$ to denote that a transition from state q_0 to state q_1 by action α has a membership degree ε to the automaton.

In FLTS next states are selected nondeterministically. The membership degree associated to each transition is used to give a measure to computations. This measure is not *exact* as the probability one induced by PLTS [13], but can be used as the base for *approximate* reasoning in the spirit of Fuzzy Logic. The membership degree associated to a transition can be thought of as a value describing how much this transition is *enabled* in the FLTS.

Semantics for FCCS processes is defined by considering function $\mathcal{N}$ associating to each process P and transition α the fuzzy set $\mathcal{P}$ of processes reachable from P with α; we identify $\mathcal{P}$ with the set

$$\{Q : \varepsilon \mid Q \text{ is reachable from } P \text{ with action } \alpha \text{ and degree } \varepsilon\}$$

ISBN: 978-989-95079-6-8

Function $\mathcal{N}$ is formally defined in Table 2 where we use $\mathcal{P}|Q$ (resp. $Q|\mathcal{P}$) denotes the fuzzy set obtained by composing each element of $\mathcal{P}$ in parallel with Q. Similarly, $\mathcal{P}|\mathcal{Q}$ is the fuzzy set containing the parallel composition of each P in $\mathcal{P}$ with each Q in $\mathcal{Q}$, where the membership degree of $P|Q$ in $\mathcal{P}|\mathcal{Q}$ is defined as $i\left(\mu_{\mathcal{P}}(P), \mu_{\mathcal{Q}}(Q)\right)$.

Table 2: $\mathcal{N}$ext Function

$$\mathcal{N}\left(\langle\alpha,\varepsilon\rangle.P,\beta\right) = \begin{cases} \{P:\varepsilon\} & if\ \alpha=\beta \\ \emptyset & else \end{cases}$$

$$\mathcal{N}\left(P+Q,\alpha\right) = \mathcal{N}\left(P,\alpha\right) \cup \mathcal{N}\left(Q,\alpha\right)$$

$$\mathcal{N}\left(P\backslash L,\alpha\right) = \begin{cases} \mathcal{N}\left(P,\alpha\right)\backslash L & if \alpha \notin L \\ \emptyset & else \end{cases}$$

$$\mathcal{N}\left(P\left[f\right],\alpha\right) = \bigcup_{\beta:f(\beta)=\alpha} \mathcal{N}\left(P,\beta\right)\left[f\right]$$

$$\mathcal{N}\left(P|Q,\alpha\right) = \begin{cases} \left[\mathcal{N}\left(P,\alpha\right)|Q\right] \cup \left[P|\mathcal{N}\left(Q,\alpha\right)\right] & if \alpha \neq \tau \\ \left[\mathcal{N}\left(P,\tau\right)|Q\right] \cup \left[P|\mathcal{N}\left(Q,\tau\right)\right] \cup & if \alpha = \tau \\ \left[\bigcup_{\alpha\in\Lambda}\left(\mathcal{N}\left(P,\alpha\right)|\mathcal{N}\left(Q,\bar{\alpha}\right)\right)\right] \cup & \\ \left[\bigcup_{\alpha\in\Lambda}\left(\mathcal{N}\left(P,\bar{\alpha}\right)|\mathcal{N}\left(Q,\alpha\right)\right)\right] & \end{cases}$$

$$\mathcal{N}\left(A,\alpha\right) = \mathcal{N}\left(P,\alpha\right) \qquad if\ A \triangleq P$$

We say that P can evolve to Q (written $P \succ Q$) if and only if there exists an action α such that if $\mathcal{P} = \mathcal{N}(P,\alpha)$, $\mathcal{P}(Q) \neq 0$. We use $\succ^*$ to denote the reflexive and transitive closure of $\succ$.

Let P be a FCCS process, we denote with $FLTS(P) = \langle S, \Lambda, \chi_{\rightarrow}\rangle$ the FLTS such that:

- $S = \{Q \mid P \succ^* Q\}$;
- $\chi_{\rightarrow}(P,\alpha,Q) = \mathcal{P}(Q)$, where $\mathcal{P} = \mathcal{N}(P,\alpha)$.

Standard behavioural equivalences, like for instance bisimulation, can be easily generalized in order to take into account fuzziness. This kind of equivalences are useful when one aims at comparing different specifications. Definition of Fuzzy Bisimulation is straightforward and it is somehow reminiscent of Stochastic Bisimulation [14].

Definition 4.2 (Fuzzy Bisimulation) *Let $\langle S, A, \chi_{\rightarrow}\rangle$ a Fuzzy LTS. An equivalence relation $R \subseteq S \times S$ is a* fuzzy bisimulation *if and only if for each p and q in S, for each equivalent class C of R in S, and for each transition label α:*

$$\chi_{\rightarrow}(p,\alpha,C) = \chi_{\rightarrow}(q,\alpha,C)$$

where:

$$\chi_{\rightarrow}(p,\alpha,C) = \bigvee_{p'\in C} \chi_{\rightarrow}(p,\alpha,p')$$

Definition 4.3 (Fuzzy Bisimilarity) *Let $\langle S, A, \chi_{\rightarrow}\rangle$ be a Fuzzy LTS. We say that $p, q \in Q$ are bisimilar ($p \sim_F q$), if there exists a fuzzy bisimulation R such that pRq.*

Relation $\sim_F$ is a *fuzzy bisimilarity* and can be defined as the largest fuzzy bisimulation, namely:

$$\sim_F \triangleq \bigcup\{R \mid R \quad \text{is a fuzzy bisimulation}\}$$

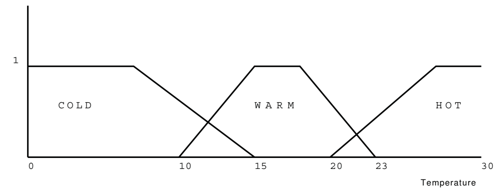

Figure 1: Temperature Fuzzy sets: COLD, NORMAL, HOT

Modeling Fuzzy Systems with FCCS We can now use FCCS for modeling the TCS described in section 4, where we consider three fuzzy sets for describing possible values of the room temperature. These sets are represented in Figure 1. Systems TCS is modeled in FCCS by splitting it into four subcomponents representing behaviours of the system. They "mime" TCS by interacting among each of them. Interactions are the result of synchronization between an input and an output action with the same name. The fuzzy value of the resulting action is calculated following semantics in Tab. 2. Process SYS_{st} is defined as follows:

$$SYS_{st} \triangleq (SENS_{st}|H|N|C)\backslash\{hot, warm, cold, inc, dec, noop\}$$

where st is the starting temperature while process $SENS_t$, which models the behaviour of temperature sensor, is defined as follows:

$$\begin{aligned} SENS_t \triangleq\ & \left\langle\overline{hot}, \mu_{hot}(t)\right\rangle.AC_t \\ & + \left\langle\overline{warm}, \mu_{warm}(t)\right\rangle.AC_t \\ & + \left\langle\overline{cold}, \mu_{cold}(t)\right\rangle.AC_t \\ & + \left\langle temp_t, 1\right\rangle.SENS_t \end{aligned}$$

$$\begin{aligned} AC_t \triangleq\ & \langle inc, 1\rangle.SENS_{t+1} \\ & + \langle dec, 1\rangle.SENS_{t-1} \\ & + \langle noop, 1\rangle.SENS_t \end{aligned}$$

Notice that in the processes above, non-determism is used for modeling the unpredictable changes in room temperature.

Controller behaviour is rendered by means of processes H, N and C that modify the AC power. These processes are defined as follows:

$$\begin{aligned} H &\triangleq \langle hot, 1\rangle.\left\langle\overline{inc}, 1\right\rangle.H \\ N &\triangleq \langle warm, 1\rangle.\left\langle\overline{noop}, 1\right\rangle.N \\ C &\triangleq \langle cold, 1\rangle.\left\langle\overline{dec}, 1\right\rangle.C \end{aligned}$$

For this system one could be interested on verifying that for each fixed starting temperature, the system always is able to reach a state where the room temperature is in a given range. In the next section, we will introduce a modal logic that will permit specifying and verifying properties of fuzzy systems.

An alternative description of this system could be obtained by considering process SYS^2_{st} defined as follows:

$$SYS^2_{st} \triangleq (SENS_{st}|CONT)\backslash\{hot, warm, cold, inc, dec, noop\}$$

ISBN: 978-989-95079-6-8

where, differently from the previous implementation, the controller is obtained as a single process $CONT$ that nondeterministically can behave like H, N and C defined above:

$$CONT \triangleq (H + N + C).CONT$$

It easy to prove that SYS_{st} and SYS^2_{st} provide the same behaviour. Namely, that $SYS_{st} \sim_F SYS^2_{st}$.

5 Fuzzy Hennessy-Milner Logic

Fuzzy Hennessy-Milner Logic (FHML) [6] is an extension of HML, which aims at specifying properties of concurrent systems whose behavior is detailed by means of FLTS. Let $\mathcal{L} = \langle L, c, i, u\rangle$, $\Phi_{\mathcal{L}}$ be the set of formulas φ defined by the following syntax:

$$\varphi ::= tt \mid \neg\,\varphi \mid \varphi \bowtie \varepsilon \mid \varphi_1 \wedge \varphi_2 \mid \langle\alpha\rangle\,\varphi \mid X \mid \nu X.\varphi$$

where $\varepsilon \in L$.

Table 3: Formulae semantics

$$\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![tt]\!]\delta(p) = 1$$

$$\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\neg\,\varphi]\!]\delta(p) = \breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta(p)$$

$$\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi_1 \wedge \varphi_2]\!]\delta(p) = i\left(\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi_1]\!]\delta(p), \mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi_2]\!]\delta(p)\right)$$

$$\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi \bowtie \varepsilon]\!]\delta(p) = \begin{cases} 1 & if\ \mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta(p) \bowtie \varepsilon \\ 0 & else \end{cases}$$

$$\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\langle\alpha\rangle\varphi]\!]\delta(p) = u_{q\in Q}\left(i\left(\chi_{\rightarrow}(p,\alpha,q), \mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta\,(q)\right)\right)$$

$$\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![X]\!]\delta = \delta(X)$$

$$\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\nu X.\varphi]\!]\delta = \cup\left\{\chi \mid \chi \leq \mathfrak{F}^{\delta,\varphi}_X(\chi)\right\}$$

$$\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![tt]\!]\delta(p) = 0$$

$$\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\neg\,\varphi]\!]\delta(p) = \mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta(p)$$

$$\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi_1 \wedge \varphi_2]\!]\delta(p) = u\left(\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi_1]\!]\delta(p), \breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi_2]\!]\delta(p)\right)$$

$$\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi \bowtie \varepsilon]\!]\delta(p) = \begin{cases} 1 & if\ \breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta(p) \not\bowtie \varepsilon \\ 0 & else \end{cases}$$

$$\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\langle\alpha\rangle\varphi]\!]\delta(p) =$$
$$= i_{q\in Q}\left(u\left(i\left(\chi_{\rightarrow}(p,\alpha,q), \breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta\,(q)\right), c\left(\chi_{\rightarrow}(p,\alpha,q)\right)\right)\right)$$

$$\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![X]\!]\delta = \delta(X)$$

$$\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\nu X.\varphi]\!]\delta = \cap\left\{\chi \mid \chi \geq^{\backsim} \mathfrak{F}^{\delta,\varphi}_X(\chi)\right\}$$

where

$$\mathfrak{F}^{\delta,\varphi}_X = \mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta[\chi/X] \qquad {}^{\backsim}\mathfrak{F}^{\delta,\varphi}_X = \breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta[\chi/X]$$

Operators are usual logical tt (true), $\neg$ (not), $\wedge$ (and). Moreover $\nu X.\varphi$ is Tarski's fixed point and $\langle\alpha\rangle\,\varphi$ the modal operator representing the property of evolving with action α to a state that behaves like φ. FHML extends HML by considering new operator $\varphi \bowtie \varepsilon$ ($\bowtie \in \{<,>\}$) that states about the satisfaction degree of a given formula in a given state. Such operator makes possible to describe the satisfaction degree of a formula in terms of an upper or a lower bound[2].

[2]This is somehow reminiscent of operator $[\varphi]_p$ proposed by Parma and Segala [15]

Other operators can be defined as macros in FHML. In the rest of this paper we let: $ff = \neg tt$, $\varphi_1 \vee \varphi_2 = \neg(\neg\varphi_1 \wedge \neg\varphi_2)$, $[\alpha]\varphi = \neg\langle\alpha\rangle\neg\varphi$ and $\mu X.\varphi = \neg\nu X\neg\varphi[\neg X/X]$.

Semantics of FHML is defined in term of functions $\mathcal{M}_{\mathcal{L},\mathcal{F}}$ and $\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}$ that for each formula φ yield the L-Fuzzy Set that gives the measure, respectively, of satisfaction and unsatisfaction of φ. This approach permits defining semantics of FHML in a general way without considering any special constraint on the underlying L-Fuzzy Sets. Indeed, in general, standard properties of sets do not hold in the case of L-Fuzzy Sets. For instance, the intersection between a L-fuzzy set and its complement could be not empty. Notice that, in general, $\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\neg\varphi]\!] \neq c\,(\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!])$.

To give a semantics to recursive formulae, interpretation functions $\mathcal{M}_{\mathcal{L},\mathcal{F}}$ and $\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}$ take also, as a parameter, a function $\delta : Q \rightarrow L$, that associates to each logical variable X a L-fuzzy set.

We assume the usual definitions on positive and negative variables in [6] for guaranteeing well-definedness of interpretation formulae.

Definition 5.1 (Well formed formula) *A formula is well formed if in each subformula of the form $\nu X.\varphi$, variable X occurs positive in φ.*

Interpretation functions are parameterized with respect to $\mathcal{L}$, used for defining the underlying L-Fuzzy Set and the relative operations, and with respect to the $\mathcal{L}$-FLTS $\mathcal{F}$ used for interpreting formulae. Note that, interpretation of FHML coincides with the standard interpretation of HML when considering standard Boolean lattices.

Definition 5.2 (Formulae Semantics) *Let $\mathcal{L} = \langle L, c, i, u\rangle$ and $\mathcal{F}_{\mathcal{L}} = \langle Q, A, \chi_{\rightarrow}\rangle$. Functions $\mathcal{M}_{\mathcal{L},\mathcal{F}} : \Phi_{\mathcal{L}} \rightarrow Q \rightarrow L$ and $\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}} : \Phi_{\mathcal{L}} \rightarrow Q \rightarrow L$ are inductively defined in Table 3.*

$\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\delta(q)$ denotes the satisfaction degree of formula φ by q. Formulae tt and ff are satisfied by every state with degree 1 and 0 respectively. A state p satisfies $\neg\varphi$ with degree ε if and only if p *does not satisfy* φ with degree ε. Fuzzy set $\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi_1 \wedge \varphi_2]\!]$ is defined as the intersection between $\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi_1]\!]$ and $\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi_2]\!]$. If a state p satisfies φ with a degree that is $<$ (resp. $>$) of ε then p satisfies $\varphi < \varepsilon$ (resp. $\varphi > \varepsilon$) with degree 1 (resp. 0). $\mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\langle\alpha\rangle\varphi]\!]\delta\,(p)$, which gives a measure of how p can reach with a transition labelled α a state satisfying φ, is defined as the disjunction, for each q, of $i(\chi_{\rightarrow}(p,\alpha,q), \mathcal{M}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\,(q))$. Where $\chi_{\rightarrow}(p,\alpha,q)$ returns the membership degree of the transition from p to q with label α to the behaviour of the system. Finally, interpretation of $\nu X.\varphi$ is defined as greates fixed point of the interpretation of φ ($\mathfrak{F}^{\delta,\varphi}_X$), which is monotone in the complete lattice of Q L-Fuzzy Sets. This, thanks to the Tarski's fixed point theorem [16], guarantees the well-definedness of functions $\mathcal{M}_{\mathcal{L},\mathcal{F}}$ and $\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}$.

The definition of function $\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}$ is similar and straightforward. However, more attention has to be paid for $\breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\langle\alpha\rangle\varphi]\!]$. Each state q, contributes to the unsatisfaction of $\langle\alpha\rangle\varphi$ by p as a factor that depends on the degree of the transition α from p to q and on the unsatisfaction degree of formula φ by q. This value is obtained as a disjunction of: $c\,(\chi_{\rightarrow}(p,\alpha,q))$ and $i\left(\chi_{\rightarrow}(p,\alpha,q), \breve{\mathcal{M}}_{\mathcal{L},\mathcal{F}}[\![\varphi]\!]\,(q)\right)$. The former indicates how much the transition from p to q with label

ISBN: 978-989-95079-6-8

α *does not belong* to the behaviour of the system. The latter, gives the measure of the unsatisfaction of φ when the action is executed.

Example 5.1 *FHML can be used for specifying that system process SYS_{st} can always reach a configuration where the room temperature is between* 18 *and* 20 *degrees. The following formula states that a configuration where temperature is between* 18 *and* 20 *degrees is eventually reached:*

$$\varphi = \mu X.\langle\overline{temp}_{18}\rangle tt \vee \langle\overline{temp}_{19}\rangle tt \vee \langle\overline{temp}_{20}\rangle tt \vee \langle\tau\rangle X$$

While formula:

$$\nu Y.\varphi \wedge [\tau]Y$$

states that φ is always satisfied.

Logical characterization of Fuzzy Bisimulation Formulae satisfaction induces an equivalence on the interpretation model and two states in a Fuzzy LTS are equivalent if (and only if) they satisfy the same set of formulae. A classical result relating modal logic and behavioural equivalence is the one in [12] showing that the equivalence induced by HML coincides with the bisimulation equivalence. A similar result can be proved when one considers FHML and Fuzzy Bisimulation. However, to prove this correspondence, one has to guarantee that the considered Fuzzy LTS is *finite-branching*.

Definition 5.3 (Finite-branching) *A Fuzzy LTS $\mathcal{F} = \langle S, A, \chi_{\rightarrow}\rangle$ is finite-branching if and only if $\forall p \in S$, $\forall \alpha \in A$ and $\forall \varepsilon \in \chi_{\rightarrow}$, the set $\left\{q' \in Q | q \xrightarrow{\alpha}_{\varepsilon} q'\right\}$ is finite.*

Theorem 1 *Let $\mathcal{F} = \langle S, A, \chi_{\rightarrow}\rangle$ be finite-branching. For each $p, q \in S$,*

$$p \sim_F q \quad \Leftrightarrow \quad \forall \varphi.\, \mathcal{M}[\![\varphi]\!](p) = \varepsilon \Leftrightarrow \mathcal{M}[\![\varphi]\!](q) = \varepsilon$$
$$\text{and } \mathcal{M}^{\smile}[\![\varphi]\!](p) = \varepsilon \Leftrightarrow \mathcal{M}^{\smile}[\![\varphi]\!](p) = \varepsilon$$

Due to lack of space, the proof is not reported.

6 Conclusions and Future Works

In this paper we have presented FCCS, a Fuzzy variant of CCS (*Calculus of Communicating Processes*), that aims at compositionally describing fuzzy behaviours. Operational semantics of FCCS has been defined by means of *Fuzzy Labelled Transition Systems* [6] (FLTS). These are an extension of Labelled Transition Systems where fuzziness is used for modeling uncertainty in concurrent systems. In the paper we have also shown how standard process algebras formal tools, like modal logics and behavioural equivalences, can be used for supporting fuzzy reasoning.

The idea to combine process algebrae and fuzzy sets is not new. For instance, in [17] is introduced a model based on interval values, to solve the problem of nondeterministic choices arising in concurrency and communication systems. However, differently from previous approaches, the present work introduces both a behavioural equivalence and a modal logic for reasoning about fuzzy specifications.

As a future work we plan to use the proposed approach for modeling the examples proposed in literature [18, 19, 20] where Fuzzy Theory is used for modeling the behaviour of "systems", like for instance those involving human interactions, where imprecision is a central feature.

References

[1] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[2] G.J. Klir and T.A. Folger. *Fuzzy sets, uncertainty, and information*. Prentice-Hall International, Inc, 1988.

[3] Jyh-Shing Roger Jang and Chuen-Tsai Sun. Neuro-fuzzy modeling and control. *Proceedings of the IEEE*, 83(3), 1995.

[4] Qiang Song and Robert P. Leland. Adaptive learning defuzzification techniques and applications. *Fuzzy Sets and Systems*, 81(3):321 – 329, 1996.

[5] Werner Van Leekwijck and Etienne E. Kerre. Defuzzification: criteria and classification. *Fuzzy Sets and Systems*, 108(2):159 – 178, 1999.

[6] Liliana D'Errico and Michele Loreti. Modeling fuzzy behaviours in concurrent systems. In L. Laura G. Italiano, E. Moggi, editor, *Proceedings of the 10th Italian Conference on Theoretical Computer Science ICTCS'07*, pages 94–105, 2007.

[7] R. Milner. *Communication and concurrency*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989.

[8] S. D. Brookes, C. A. R. Hoare, and A. W. Roscoe. A theory of communicating sequential processes. *J. ACM*, 31(3):560–599, 1984.

[9] J.A. Bergstra and J.W. Klop. Algebra of communicating processes with abstraction. *Theoretical Computer Science*, (37):77–121, 1985.

[10] L. A. Zadeh. Fuzzy logic = computing with words. *IEEE Transactions on Fuzzy Systems*, 4(2):103–111, 1996.

[11] J. A. Goguen. L-fuzzy sets. *J. Math. Anal. Appl.*, 18:145–174, 1967.

[12] Matthew Hennessy and Robin Milner. Algebraic laws for nondeterminism and concurrency. *J. ACM*, 32(1):137–161, 1985.

[13] Roberto Segala and Nancy Lynch. Probabilistic simulations for probabilistic processes. *Nordic J. of Computing*, 2(2):250–273, 1995.

[14] Ed Brinksma and Holger Hermanns. Process algebra and markov chains. In Ed Brinksma, Holger Hermanns, and Joost-Pieter Katoen, editors, *Euro Summer School on Trends in Computer Science*, volume 2090 of *Lecture Notes in Computer Science*, pages 183–231. Springer, 2001.

[15] A. Parma and R. Segala. Logical characterizations of bisimulations for discrete probabilistic systems. In Springer-Verlag, editor, *FOSSACS 2007, LNCS*, volume 4423, pages 287–301, 2007.

[16] A. Tarski. A lattice-teoretichal fixpoint theorem and its applications. *Pacific J. Math.*, 5:285–309, 1955.

[17] Yixiang Chen and Jie Zhou. Fuzzy interval-valued processes algebra. In S.Barry Cooper, Benedict Lowe, and Leen Torenvliet, editors, *CiE 2005 Abstracts*, ILLC Publications, pages 44–57, 2005.

[18] L. A. Zadeh. Commonsense reasoning based on fuzzy logic. In *WSC '86: Proceedings of the 18th conference on Winter simulation*, pages 445–447, New York, NY, USA, 1986. ACM Press.

[19] Eviatar Tron and Michael Margaliot. Mathematical modeling of observed natural behavior: a fuzzy logic approach. *Fuzzy Sets and Systems*, 2004.

[20] Bernd Möller, Wolfgang Graf, and Song Ha Nguyen. Modeling the life cycle of a structure using fuzzy processes. *Computer-Aided Civil and Infrastructure Engineering*, 2004.

ISBN: 978-989-95079-6-8