An approach to membrane computing under inexactitude

J. Casasnovas, J. Miró, M. Moyà, and F. Rosselló

Department of Mathematics and Computer Science
Research Institute of Health Science (IUNICS)
University of the Balearic Islands
E-07122 Palma de Mallorca, Spain
{jaume.casasnovas,joe.miro,manuel.moya,cesc.rossello}@uib.es

Abstract. In this paper we introduce a fuzzy version of symport/antiport membrane systems. Our fuzzy membrane systems handle possibly inexact copies of reactives and their rules are endowed with threshold functions that determine whether a rule can be applied or not to a given set of objects, depending of the degree of accuracy of these objects to the reactives specified in the rule. We prove that these fuzzy membrane systems generate exactly the recursively enumerable finite-valued fuzzy subsets of $\mathbb{N}$.

Keywords: Membrane computing; P-systems; Fuzzy sets; Universality; Biochemistry.

1 Introduction

Membrane computing is a formal computational paradigm, invented in 1998 by Gh. Păun [9], that rewrites multisets of objects within a spatial structure inspired by the membrane structure of living cells and according to evolution rules that are reminiscent of the processes that take place inside cells. Most approaches to membrane computing developed so far have been exact: the objects used in the computations are exact copies of the reactives involved in the biochemical reactions modelled by the rules, and every application of a given rule always yields exact copies of the objects it is assumed to produce. But, in everyday's practice, one finds that cells do not behave in this way. Biochemical reactions may deal with inexact, mutated copies of the reactives involved in them, and errors may happen when a biochemical reaction takes place. These errors can be due, for instance, to the inexactitude of the chemical compounds used or to unnoticed changes in the surrounding conditions.

The inexactitude underlying cell processes made Gh. Păun ask in his very first list of open problems in membrane computing [10], dated October 2000, for the development of "approximate" mathematical approaches. A first answer to this question was given by A. Obtułowicz and Gh. Păun himself by extending the classical model to a probabilistic one [7]. Actually, these authors discussed several ways of introducing probabilities in membrane computing: at the level of objects (each object lies in a membrane with a certain probability), at the level of rules (at each moment, each rule is fired with a certain probability), and at the level of targets (outputs of applications of rules are moved to each possible membrane with a certain probability).

Beyond this probabilistic approach, Gh. Păun has asked more specifically for the development of a rough set version of membrane computing in a later list of open problems [12] and Obtułowicz [6] has discussed several possible rough set based mathematical models of uncertainty that could be used in membrane computing. In the Concluding Remarks section of their aforementioned paper [7], A. Obtułowicz and Gh. Păun proposed the use of fuzzy

set approaches to introduce uncertainty effects in membrane computing. But, although the probabilistic approaches developed in that paper can be easily generalized to the possibilistic setting, other fuzzy approaches to the membership of objects to membranes may have some drawbacks. For instance, the classical rules of fuzzy logic —for the usual operations between fuzzy sets valued in $[0, 1]$ and involutive complement operation c [3]— entail that, for every membrane (including the environment) m_0, the maximum of the membership values of a given object to all membranes other than m_0 must be the image under c of its membership value to m_0. Then, it is straightforward to prove that this rule implies that, for every object, there exists some $\alpha \in [0, 1]$ such that its membership value to each membrane is $\alpha \wedge c(\alpha)$ for all membranes but one, and $\alpha \vee c(\alpha)$ for the remaining membrane: almost a crisp situation.

Nevertheless, there is one uncertainty aspect that cannot be handled by means of probabilistic methods and that is suitable to being attacked using a fuzzy set approach: the inexactitude of the reactives involved in computations. I.e., the fact that the actual objects used in computations, as well as the actual output of the latter, need not be exact copies of the reactives that are assumed to be used in the computations or to be produced by them but only approximate copies of these reactives. In this paper we present a first approach to the use of fuzzy methods to handle this kind of uncertainty. For simplicity, we consider only symport/antiport systems [8] —membrane systems whose rules only move reactives through membranes— but it is straightforward to extend our approach to other models of membrane computing. Our fuzzy symport/antiport systems also move objects through membranes, but now these objects can be inexact copies of reactives and each rule is endowed with threshold functions that determine whether it can be applied or not to a given set of objects, depending of the degree of approximation of these objects to the reactives specified in the rule. Then we prove that these fuzzy membrane systems are universal in the sense that they generate exactly the recursively enumerable finite-valued fuzzy subsets of $\mathbb{N}$.

2 Preliminaries

In this section we recall some concepts on fuzzy sets and multisets and on symport/antiport membrane systems, and we take the opportunity to establish some notations and conventions, some of them not the standard ones, that we shall use.

2.1 Fuzzy sets

Any subset Y of a set X can be identified with its *membership*, or *characteristic*, *mapping* $\chi_Y : X \to \{0, 1\}$, defined by $\chi_Y(x) = 1$ if $x \in Y$ and $\chi_Y(x) = 0$ if $x \notin Y$. Fuzzy subsets generalize this interpretation of subsets as membership mappings by allowing membership values other than 0 and 1. Thus, a *fuzzy subset* of a set X is a mapping from X to the unit interval $[0, 1]$. More in general, given any subset I of $[0, 1]$, an *I-fuzzy subset* of a set X is a mapping from X to I. Whenever we speak about *I*-fuzzy subsets (or multisets, see below), we shall assume that $0, 1 \in I$. A fuzzy subset of a set X is *finite-valued* when its image is a finite subset of $[0, 1]$, i.e., when it is *I*-fuzzy for some finite subset I of $[0, 1]$.

For every fuzzy subset $\varphi : X \to [0, 1]$, its *t-level*, for every $t \in [0, 1]$, is

$$\varphi_t = \{x \in X \mid \varphi(x) \geq t\}.$$

Notice that $\varphi_0 = X$ and, for every $t, t' \in [0, 1]$, if $t \leq t'$, then $\varphi_t \supseteq \varphi_{t'}$. For every fuzzy subsets $\varphi, \varphi' : X \to [0, 1]$, $\varphi = \varphi'$ if and only if $\varphi_t = \varphi'_t$ for every $t \in [0, 1]$.

If φ is I-fuzzy and $t_0, t_1 \in I$ are such that the open interval $]t_0, t_1[$ does not contain any element of I, then $\varphi_t = \varphi_{t_1}$ for every $t \in]t_0, t_1]$. Thus, if φ, φ' are I-fuzzy subsets of X, then $\varphi = \varphi'$ if and only if $\varphi_t = \varphi'_t$ for every $t \in I$. This allows us, when dealing with I-fuzzy subsets, to consider only their t-levels for $t \in I$.

A (crisp) *multiset*, or *bag*, over a set V is simply a mapping $d : V \to \mathbb{N}$. The usual interpretation of a multiset $d : V \to \mathbb{N}$ is that it describes a set consisting of $d(v)$ "exact" copies of each $v \in V$, without specifying which element of the set is a copy of which element of V. A natural generalization of this interpretation of multisets leads to a first definition of *fuzzy multiset*, or *fuzzy bag*, over a set V as a mapping $F : V \times [0, 1] \to \mathbb{N}$. Such a fuzzy multiset can be understood as describing a set consisting, for each $v \in V$ and $t \in [0, 1]$, of $F(v, t)$ "possibly inexact" copies of v with degree of similarity t to it. In other words, we understand that a fuzzy multiset F over V describes a set endowed with a family $(F^{(v)})_{v \in V}$ of fuzzy subsets that contains $F(v, t)$ elements x such that $F^{(v)}(x) = t$, for every $v \in V$ and $t \in [0, 1]$.

It will be convenient for our purposes to take a slightly modified definition of fuzzy multiset. The first modification affects both crisp and fuzzy multisets. Sometimes we shall need to represent the fact that the set described by a multiset (resp., a fuzzy multiset) F over V contains an arbitrarily large number of copies of some elements v of V (resp., with some degree of similarity t). We shall do it by writing $F(v) = \infty$ (resp., $F(v, t) = \infty$). Thus, our (crisp and fuzzy) multisets will actually take values in $\mathbb{N} \cup \{\infty\}$. To simplify the notations, from now on we shall denote this set $\mathbb{N} \cup \{\infty\}$ by $\mathbb{N}_\infty$.

On the other hand, we impose two limitations on the interpretation of a fuzzy multiset as a set that allow us to modify its definition; cf. [1]. First, we shall assume that if an element of the set described by a fuzzy multiset over V is an inexact copy of $v \in V$ with degree of similarity $t > 0$, then it cannot be an inexact copy of any other element in V with any non-zero degree of similarity. And second, we shall also assume that the set described by a fuzzy multiset over V does not contain any element that is not a copy of some $v \in V$ with some non-zero degree of similarity, or, rather, we shall not take into consideration these elements. These conditions entail that, for every $v \in V$, the value $F(v, 0)$ must be equal to $\sum_{w \in V - \{v\}} \sum_{t \in]0,1]} F(w, t)$ and in particular that the restriction of F to $V \times \{0\}$ is determined by the restriction of F to $V \times]0, 1]$.

These restraints allow us to define in this paper a *fuzzy multiset* over a set V as a mapping

$$F : V \times]0, 1] \to \mathbb{N}_\infty.$$

Not having to care about the images under fuzzy multisets of the elements of the form $(v, 0)$ will greatly simplify some of the definitions and results that will be introduced in the main body of this paper.

For every $I \subseteq [0, 1]$, we shall denote $I - \{0\}$ by I^+. Consistently with our definition of fuzzy multiset over a set V, given any $I \subseteq [0, 1]$, an I-*fuzzy multiset over V* will be a mapping $F : V \times I^+ \to \mathbb{N}_\infty$. Every such I-fuzzy multiset can be understood as defined on the whole $V \times]0, 1]$ by extending it by means of $F(v, t) = 0$ for every $v \in V$ and $t \in]0, 1] - I^+$.

A fuzzy multiset $F : V \times]0, 1] \to \mathbb{N}_\infty$ over a *finite* set V is *finite-valued* when $F(v, t) = 0$ for all ordered pairs $(v, t) \in V \times]0, 1]$ except a finite number of them. This is equivalent to say that the image of F is finite and the preimage under F of every $n \in \mathbb{N}_\infty - \{0\}$ is a finite set. If $I \subseteq [0, 1]$ is a finite set such that $F(v, t) \neq 0$ for some $v \in V$ implies $t \in I$, then we shall identify such a fuzzy multiset with the I-fuzzy multiset $F : V \times I^+ \to \mathbb{N}_\infty$ obtained as its restriction to $V \times I^+$.

2.2 Crisp symport/antiport membrane systems

In this subsection we explain in detail the basic model of membrane systems with symport/antiport rules, with notations that are not the usual ones but that will be helpful in the generalization of this model to the fuzzy setting. The interested reader can look up Chapter 4 of Gh. Păun's textbook on molecular computing [11] and the references cited therein for more information on this model of computation.

Given an alphabet V, we denote by V^* the set of words over V. Given a word $w \in V^*$, we denote by $|w|$ the length of w and, given a letter $a \in V$, by $|w|_a$ the number of occurrences of a in w.

A *membrane structure* μ is a finite rooted tree whose nodes are called *membranes*. We shall always denote by M the set of membranes of a membrane structure and, in practice, we shall assume that these membranes are injectively labelled by natural numbers, in such a way that the root's label is 1. The edges of μ are oriented pointing to the root.

This tree represents a hierarchical structure of nested membranes, with the edges representing the relation "being directly inside": an edge going from a membrane m to a membrane m' means that m is directly included in m'. The tree's root 1 is then called the *skin membrane*, because it surrounds the whole membrane system, and the tree's leaves are called *elementary membranes*, because no further membrane lies inside them.

We expand every membrane structure μ by adding a new node to it labelled *env* and an arc going from 1 to *env*; let $\overline{\mu}$ denote the resulting tree and $\overline{M}$ its set of nodes $M \cup \{env\}$. This new node *env* is called the *environment*, because it surrounds the skin membrane. In this way, *env* becomes the root of $\overline{\mu}$. Although, formally, *env* is not a membrane, when we generically talk about membranes, we shall include it unless we explicitly state otherwise.

For every $m \in M$, we shall denote by $\varepsilon(m)$ the target node of the arc in $\overline{\mu}$ whose source node is m, i.e., the membrane which m is directly included into. Notice that $\varepsilon(1) = env$, and that if $m \neq 1$, then $\varepsilon(m) \in M$.

We understand that every $m \in \overline{M}$ defines a *region* K_m. For an elementary membrane, it would represent the space enclosed by it, and for any other membrane it would represent the space comprised between this membrane and those directly included in it. The *env* node also defines a region K_{env}, which would represent the space outside the skin membrane: the *environment*, indeed. The rules in the membrane systems considered in this paper will move in a controlled way objects from regions K_m to regions $K_{\varepsilon(m)}$ and, the other way round, from regions $K_{\varepsilon(m)}$ to regions K_m.

At each moment, every such region contains a set of objects that are copies of elements of a certain finite set of *reactives*. If V stands for this set of reactives, the content of all regions K_m, with $m \in \overline{M}$, at any moment is represented by means of an $\overline{M}$-indexed family $(F_m)_{m \in \overline{M}}$ of multisets over V

$$F_m : V \to \mathbb{N}_\infty, \quad m \in \overline{M}.$$

We shall call such an $\overline{M}$-indexed family of multisets $(F_m)_{m \in \overline{M}}$ a *configuration*.

Now, a *symport/antiport membrane system*, a *P-system* for short, is a structure

$$\Pi = (V, V_{out}, \mu, m_{out}, (S_m)_{m \in \overline{M}}, (\mathcal{R}_m)_{m \in M})$$

where:

- V is the set of *reactives* used by the membrane system; it is a finite set.

- $V_{out} \subseteq V$ is the set of *output reactives*; these are the only objects that matter at the end of a computation.[1]
- μ is a membrane structure, with set of membranes M.
- $m_{out} \in M$ is the *output membrane*; the results of the computations are read in the region defined by this membrane.
- $(S_m)_{m \in \overline{M}}$ is a configuration, called *initial*, that describes the initial content of each region K_m. We assume that, for each $v \in V$, $S_{env}(v)$ is either 0 or ∞, and that $S_m(v) \neq \infty$ for every $m \in M$ and $v \in V$.
- For every $m \in M$, $\mathcal{R}_m$ is a finite set of *evolution rules* associated to the membrane m: notice that the environment has no rule associated to it. These rules represent changes of place of objects between K_m and regions adjacent to it. Every rule in $\mathcal{R}_m$ has the form

$$R = (\underline{a}, in; \underline{b}, out),$$

where $\underline{a}, \underline{b} \in V^*$ represent the multisets of reactives that enter (in) or exit (out) the region defined by the membrane m under the action of this rule.

When $\underline{b}$ or $\underline{a}$ is the empty word λ, the rule R is said to be a *symport rule*, and it is simply written $(\underline{a}, in)$ or $(\underline{b}, out)$, respectively. When $\underline{a}, \underline{b} \neq \lambda$, the rule R is said to be an *antiport rule*.

Let $m_0 \in M$ be a membrane and $\varepsilon(m_0)$ the membrane in $\overline{\mu}$ directly outside it. An evolution rule $R = (\underline{a}, in; \underline{b}, out) \in \mathcal{R}_{m_0}$ *can be applied* to a configuration $(F_m)_{m \in \overline{M}}$ when, for every $v \in V$,

$$F_{\varepsilon(m_0)}(v) \geq |\underline{a}|_v \text{ and } F_{m_0}(v) \geq |\underline{b}|_v.$$

And when it can be applied, its *application* produces a new configuration $(F'_m)_{m \in \overline{M}}$, which we call the *result* of this application, that is obtained as follows:

- $F'_m = F_m$ if $m \neq m_0, \varepsilon(m_0)$;
- $F'_{m_0}(v) = F_{m_0}(v) - |\underline{b}|_v + |\underline{a}|_v$ for every $v \in V$;
- $F'_{\varepsilon(m_0)}(v) = F_{\varepsilon(m_0)}(v) + |\underline{b}|_v - |\underline{a}|_v$ for every $v \in V$.

This represents that $(F'_m)_{m \in \overline{M}}$ is obtained from $(F_m)_{m \in \overline{M}}$ by moving, for every $v \in V$, $|\underline{a}|_v$ copies of v from $K_{\varepsilon(m_0)}$ to K_{m_0} and $|\underline{b}|_v$ copies of v from K_{m_0} to $K_{\varepsilon(m_0)}$.

A *transition* for a P-system Π consists of a maximal simultaneous application of evolution rules. These rules are chosen non-deterministically in such a way that no further rule in $\mathcal{R}_m$, for any m, can be triggered simultaneously to them. Formally, a transition consists of the simultaneous application to a configuration $(F_m)_{m \in \overline{M}}$ of a family of rules

$$(R_{m,1}, \ldots, R_{m,r_m})_{m \in M},$$

with $R_{m,i} = (a_{m,i}, in; b_{m,i}, out) \in \mathcal{R}_m$, for every $m \in M$ and $i = 1, \ldots, r_m$. This family of rules must satisfy the following two conditions:

(1) For every elementary membrane m and for every $v \in V$,

$$\sum_{i=1}^{r_m} |\underline{b_{m,i}}|_v \leq F_m(v);$$

---

[1] In the original definition of symport/antiport membrane systems, no set of output reactives is distinguished, i.e., $V = V_{out}$. But, the specification of a set of output reactives will simplify a proof in the fuzzy setting, and it does not increase the computational power of these systems: see Remark 1 at the end of this section.

for every non-elementary membrane $m \in M$, say with $m = \varepsilon(m_1) = \ldots = \varepsilon(m_l)$, and for every $v \in V$,

$$\sum_{j=1}^{l} \sum_{i=1}^{r_{m_j}} |\underline{a_{m_j,i}}|_v + \sum_{i=1}^{r_m} |\underline{b_{m,i}}|_v \leq F_m(v);$$

and, finally,

$$\sum_{i=1}^{r_1} |\underline{a_{1,i}}|_v \leq F_{env}(v).$$

(2) No further rule can be added to any $(R_{m,1}, \ldots, R_{m,r_m})$ in such a way that the resulting family of rules still satisfies the application condition (1).

These conditions globally impose that there are enough reactives in all regions to allow the simultaneous application of all rules $R_{m,i}$, but that there are not enough reactives to allow the application of any further rule.

Now, the simultaneous application of these rules to a configuration $(F_m)_{m \in \overline{M}}$ produces a new configuration $(\widehat{F}_m)_{m \in \overline{M}}$ that is obtained as follows:

– For every elementary membrane m and for every $v \in V$,

$$\widehat{F}_m(v) = F_m(v) + \sum_{i=1}^{r_m} |\underline{a_{m,i}}|_v - \sum_{i=1}^{r_m} |\underline{b_{m,i}}|_v.$$

– For every non-elementary membrane $m \in M$, say with $m = \varepsilon(m_1) = \ldots = \varepsilon(m_l)$, and for every $v \in V$,

$$\widehat{F}_m(v) = F_m(v) + \sum_{i=1}^{r_m} |\underline{a_{m,i}}|_v + \sum_{j=1}^{l} \sum_{i=1}^{r_{m_j}} |\underline{b_{m_j,i}}|_v - \sum_{j=1}^{l} \sum_{i=1}^{r_{m_j}} |\underline{a_{m_j,i}}|_v - \sum_{i=1}^{r_m} |\underline{b_{m,i}}|_v.$$

– And, for every $v \in V$,

$$\widehat{F}_{env}(v) = F_{env}(v) + \sum_{i=1}^{r_1} |\underline{b_{1,i}}|_v - \sum_{i=1}^{r_1} |\underline{a_{1,i}}|_v.$$

We shall forbid the existence in $\mathcal{R}_1$ of any symport rule of the form $(\underline{a}, in)$ with $\underline{a} \in V^*$ such that $S_{env}(v) = \infty$ if $|\underline{a}|_v > 0$, because any such rule could be applied an infinite number of times in any transition.

A finite sequence of transitions between configurations of a P-system Π, starting with the initial configuration, is called a *computation* with respect to Π. A computation C *halts* when it reaches a *halting configuration* $(H(C)_m)_{m \in \overline{M}}$ where no rule can be applied. The *output* of such a halting computation C is the final number of output reactives contained in the region defined by the output membrane:

$$Out_{\Pi,C} = \sum_{v \in V_{out}} H(C)_{m_{out}}(v).$$

A computation that does not halt does not yield any output.

The set $Gen(\Pi) \subseteq \mathbb{N}$ *generated* by Π is the set of all outputs $Out_{\Pi,C}$ of halting computations C with respect to Π.

We have now the following result; see [8,11,4].

Theorem 1. *A subset of $\mathbb{N}$ is recursively enumerable if and only if it is generated by some P-system.*

Moreover, every recursively enumerable subset of $\mathbb{N}$ can be generated by a P-system that satisfies the following conditions:

- *its membrane structure has only two nodes, and the output membrane is the elementary one;*
- *it has only symport rules;*
- *all reactives used by the P-system are output reactives;*
- *the rules associated to the output membrane are (α, in), $(\#, in)$ and $(\#, out)$ for some specific reactives α and $\#$, and α is the only reactive that may enter the output membrane in any halting computation;*
- *in the initial configuration, both the skin membrane and the output membrane do not contain any copy of this reactive α.* $\square$

Remark 1. Notice that last theorem establishes that every recursively enumerable subset of $\mathbb{N}$ is generated by a P-system all whose reactives are considered as output reactives. Since, by Church-Turing Thesis, a P-system with a set of output reactives specified will generate a recursively enumerable subset of $\mathbb{N}$, we deduce that the specification of a set of output reactives does not increase the computational power of the model, as we claimed when we defined our P-systems.

Besides, we also have that every recursively enumerable subset of $\mathbb{N}$ is generated by a P-system with only one output reactive: the only reactive α that may enter the output membrane in any halting computation in the P-system given by the last theorem.

3 The fuzzy model

We assume henceforth the existence of a *universe X* containing all objects we use in computations.

Roughly described, a fuzzy P-system will be a structure similar to a crisp P-system, supported on a membrane structure that defines regions whose contents evolve following rules that specify the transport of reactives through membranes. But the details will be quite different.

To begin with, we shall use *reactives* as "ideal definitions" of chemical compounds, and hence they are fuzzy subsets of X: for every reactive $v : X \to [0, 1]$, we understand that $v(x) = t$ denotes that the object $x \in X$ is a copy of $v \in V$ with a degree t of exactitude. So, $v(x) = 1$ means that x is an *exact copy* of the reactive v, and $v(x) = 0$ means that x cannot represent in any way the reactive v.

Actually, every reactive will be, for the purposes of each fuzzy P-system, a finite-valued fuzzy subset of X: this represents that, in any fuzzy P-system, only a finite set of values of accuracy of objects to reactives will be taken into account. This can be seen as translating nature's discreteness, or that the accuracy of an object to a reactive cannot be measured exactly, but only up to some threshold. As we shall explain in the Conclusion, this finite-valuedness assumption does not decrease the computational power of our fuzzy P-systems: if we allowed the reactives to take values in the whole $[0, 1]$, the set of natural numbers generated by a fuzzy P-system would still be finite-valued.

We shall say that an object $x \in X$ is *similar* to a reactive $v \in V$ when $v(x) > 0$. To simplify the definition of an application of a rule, and as it was already hinted in §2.1, we shall assume

in this paper that each object in X is similar to at most one reactive, and it will be clear from the definition of the application of a rule that in each fuzzy P-system we shall not care about objects that are not similar to some reactive among those used in it.

As in the crisp case, fuzzy P-systems will be supported by a membrane structure and each membrane in it will define a region. But, the reactives being fuzzy sets, the content of these regions at each moment will be formally described by means of an $\overline{M}$-indexed family of *fuzzy multisets* over a set V of reactives. These fuzzy multisets specify, for every $v \in V$ and for every value $t \in]0,1]$, how many objects in each region K_m are copies of the reactive v with degree of accuracy t.

Since each fuzzy P-system will involve only a finite set of reactives V, and, for the purposes of each specific P-system, we consider each reactive as a finite-valued fuzzy subset of X, all possible values of accuracy of objects to reactives used in a given fuzzy P-system form a finite subset of $[0,1]$. Thus, we shall specify in the description of a fuzzy P-system a finite subset I of $[0,1]$ that will contain all these images as well as all other elements in $[0,1]$ needed in that description. Then, a *configuration* for this fuzzy P-system, with set of membranes M and set of reactives V, will be a family of I-valued fuzzy multisets $(F_m)_{m \in \overline{M}}$ over V,

$$F_m : V \times I^+ \to \mathbb{N}_\infty, \qquad m \in \overline{M}.$$

Each such mapping F_m specifies, for every $v \in V$ and for every $t \in I^+$, how many objects there exist in the region K_m such that $v(x) = t$ at the moment described by the configuration. We impose several conditions on these configurations. First, every F_m with $m \in M$ is such that $F_m(v,t) < \infty$ for every $v \in V$ and for every $t \in I^+$: this translates the fact that the regions defined by the membranes other than the environment can only contain at any time a finite set of objects. Second, we allow the environment to contain at every moment an unlimited supply of copies of some reactives v, and then with all possible degrees of accuracy $t \in I^+$: we shall represent it by writing $F_{env}(v,t) = \infty$ for every $t \in I^+$.

Now, a *fuzzy symport/antiport membrane system*, a *fuzzy P-system* for short, is a structure

$$\Pi = (V, V_0, \mu, m_{out}, I, (S_m)_{m \in \overline{M}}, (\mathcal{R}_m)_{m \in M}),$$

where:

- V is the finite set of *reactives* used by the membrane system.
- $V_{out} \subseteq V$ is the set of *output reactives*.
- μ is a membrane structure, with set of membranes M.
- $m_{out} \in M$ is the *output membrane*.
- I is a finite subset of $[0,1]$ containing 0 and 1.
- $(S_m)_{m \in \overline{M}}$ is a family of I-valued fuzzy multisets over V, called the *initial configuration*, which describes the initial content of all regions K_m. We impose that, for each $v \in V$, either $S_{env}(v,t) = 0$ for every $t \in I^+$ or $S_{env}(v,t) = \infty$ for every $t \in I^+$. This translates the assumption that, for every reactive v, it either happens that the environment does not contain any object similar to it or that it contains an unbounded homogeneous supply of copies of it.
- For every $m \in M$, $\mathcal{R}_m$ is a finite set of *evolution rules* associated to m. Each evolution rule in $\mathcal{R}_m$ has the form
$$R = ((\underline{a}, in; \underline{b}, out), \tau_{in}, \tau_{out}),$$
  where:

— $(\underline{a}, in; \underline{b}, out)$ is a crisp symport/antiport rule; we shall say that a reactive v is *incoming* (resp., *outgoing*) for this rule R when $|\underline{a}|_v > 0$ (resp., $|\underline{b}|_v > 0$).

— $\tau_{in}, \tau_{out} : V \to I$ are *threshold* functions that determine, for every incoming or outgoing reactive for R, respectively, the degree of accuracy of an object to this reactive that is necessary for this object to be considered as this reactive to the effect of triggering an application of this rule.

We impose on these threshold functions that $\tau_{in}(v) > 0$ for every incoming reactive and $\tau_{out}(v) > 0$ for every outgoing reactive: objects that are not similar to an incoming or outgoing reactive can never play its role in the application of a rule. Moreover, and for simplicity, we do not impose any threshold condition on reactives that are not incoming or outgoing: if $|\underline{a}|_v = 0$, then $\tau_{in}(v) = 0$, and if $|\underline{b}|_v = 0$, then $\tau_{out}(v) = 0$.

As in the crisp case, when $\underline{b}$ or $\underline{a}$ is the empty word λ, R is said to be a *symport rule*, and we shall simply write it as $((\underline{a}, in), \tau)$ or $((\underline{b}, out), \tau)$, respectively: in the rules of the first type, τ represents τ_{in}, and in those of the second type, it represents τ_{out}. When $\underline{a}, \underline{b} \neq \lambda$, R is said to be an *antiport rule*.

Also as in the crisp case, and for the very same reason as then, we forbid the existence in $\mathcal{R}_1$ of symport rules of the form $((\underline{a}, in), \tau)$ with $\underline{a} \in V^*$ such that $S_{env}(v, -) = \infty$ if $|\underline{a}|_v > 0$.

Let $m_0 \in M$ be a membrane and $\varepsilon(m_0)$ the membrane in $\overline{\mu}$ directly outside it. An evolution rule

$$R = ((\underline{a}, in; \underline{b}, out), \tau_{in}, \tau_{out})$$

in $\mathcal{R}_{m_0}$ *can be triggered* in a configuration $(F_m)_{m \in \overline{M}}$ when, for every $v \in V$,

$$\sum_{t \geq \tau_{in}(v)} F_{\varepsilon(m_0)}(v, t) \geq |\underline{a}|_v \text{ and } \sum_{t \geq \tau_{out}(v)} F_{m_0}(v, t) \geq |\underline{b}|_v.$$

This means that there are more copies of every incoming or outgoing reactive in the regions $K_{\varepsilon(m_0)}$ and K_{m_0}, respectively, within the degree of accuracy required by the threshold functions, than the specified quantities.

When a rule

$$R = ((\underline{a}, in; \underline{b}, out), \tau_{in}, \tau_{out}) \in \mathcal{R}_{m_0}$$

can be triggered in a configuration $(F_m)_{m \in \overline{M}}$, an *application* of it modifies this configuration into a new configuration $(F'_m)_{m \in \overline{M}}$, which we call the *result* of this specific application. This new configuration is obtained as follows:

(1) For every reactive $v \in V$, we choose $|\underline{a}|_v$ objects in $K_{\varepsilon(m_0)}$ with degree of accuracy to v at least $\tau_{in}(v)$. Formally, to do it, for every $v \in V$, we take a mapping $enter_v^R : I^+ \to \mathbb{N}$ such that:

— If $t < \tau_{in}(v)$, then $enter_v^R(t) = 0$.

— If $t \geq \tau_{in}(v)$, then $0 \leq enter_v^R(t) \leq F_{\varepsilon(m_0)}(v, t)$.

— $\sum_{t \in I^+} enter_v^R(t) = |\underline{a}|_v$.

Notice in particular that if $|\underline{a}|_v = 0$, then $enter_v^R(t) = 0$ for every $t \in I^+$.

This corresponds to choosing, for every $t \geq \tau_{in}(v)$, a certain number $enter_v^R(t)$ of objects x in $K_{\varepsilon(m_0)}$ such that $v(x) = t$ and in such a way that the total amount of these objects is $|\underline{a}|_v$. These objects, or, rather, the number of them within each degree $t \geq \tau_{in}(v)$ of accuracy to v, are chosen in a non-deterministic way: taking a different mapping $enter_v^R$ would correspond to a different application of the rule and hence it could lead to a different result.

(2) In a similar way, for every $v \in V$, we choose $|\underline{b}|_v$ objects in K_{m_0} with degree of accuracy to v at least $\tau_{out}(v)$. As before, we do it by taking, for every $v \in V$, a mapping $exit_v^R : I^+ \to \mathbb{N}$ such that:

— If $t < \tau_{out}(v)$, then $exit_v^R(t) = 0$.

— If $t \geq \tau_{out}(v)$, then $0 \leq exit_v^R(t) \leq F_{m_0}(v, t)$.

— $\sum_{t \in I^+} exit_v^R(t) = |\underline{b}|_v$.

We have again that if $|\underline{b}|_v = 0$, then $exit_v^R(t) = 0$ for every $t \in I^+$.

(3) For every reactive $v \in V$, we move from K_{m_0} to $K_{\varepsilon(m_0)}$ the $|\underline{b}|_v$ possibly inexact copies of it that have been chosen by means of the mapping $exit_v^R$, and we move from $K_{\varepsilon(m_0)}$ to K_{m_0} the $|\underline{a}|_v$ possibly inexact copies of it that have been chosen by means of $enter_v^R$. This leads to a new configuration $(F'_m)_{m \in \overline{M}}$ defined as follows:

– $F'_m = F_m$ if $m \neq m_0, \varepsilon(m_0)$.

– $F'_{m_0}(v, t) = F_{m_0}(v, t) - exit_v^R(t) + enter_v^R(t)$ for every $v \in V$ and $t \in I^+$.

– $F'_{\varepsilon(m_0)}(v, t) = F_{\varepsilon(m_0)}(v, t) + exit_v^R(t) - enter_v^R(t)$ for every $v \in V$ and $t \in I^+$.

Consequently, if $F_{env}(v, t) = \infty$, then $F'_{env}(v, t) = \infty$, too, and if $F_{env}(v, t) \neq \infty$, then $F'_{env}(v, t) \neq \infty$ either.

This new configuration $(F'_m)_{m \in \overline{M}}$ is the *result* of this application of R. Let us point out again that a given rule may admit several applications to a given configuration, yielding different results, depending on the mappings taken in steps (1) and (2). This does not happen in the crisp case.

Now, a *transition* for a fuzzy P-system Π consists of a maximal simultaneous application of rules in the same sense as in the crisp case: the triggering condition must be satisfied simultaneously for all rules, and then all steps (1) and (2) corresponding to rules being applied in one transition are performed simultaneously, and finally all steps (3) are performed together. The rules applied in a given transition are chosen non-deterministically but so that no further rule in $\mathcal{R}_m$ for any m can be triggered simultaneously to them. In particular, a given rule can be triggered several times in the same transition, provided enough copies of the corresponding incoming and outgoing reactives are available within the required degree of exactitude.

Formally, a transition consists of the simultaneous application to a configuration $(F_m)_{m \in \overline{M}}$ of a family of rules

$$(R_{m,1}, \ldots, R_{m,r_m})_{m \in M},$$

with

$$R_{m,i} = ((\underline{a_{m,i}}, in; \underline{b_{m,i}}, out), \tau_{in}^{m,i}, \tau_{out}^{m,i}) \in \mathcal{R}_m, \quad m \in M, \ i = 1, \ldots, r_m.$$

These rules must satisfy that:

(a) For every $v \in V$ and for every $t \in I^+$,

– for every elementary membrane m,

$$\sum_{i \text{ s.t. } \tau_{out}^{m,i}(v) \geq t} |\underline{b_{m,i}}|_v \leq \sum_{t' \geq t} F_m(v, t');$$

– for every non-elementary membrane $m \in M$ with $m = \varepsilon(m_1) = \ldots = \varepsilon(m_l)$,

$$\sum_{i \text{ s.t. } \tau_{out}^{m,i}(v) \geq t} |\underline{b_{m,i}}|_v + \sum_{j=1}^{l} \sum_{i \text{ s.t. } \tau_{in}^{m_j,i}(v) \geq t} |\underline{a_{m_j,i}}|_v \leq \sum_{t' \geq t} F_m(v, t');$$

– finally, as far as *env* goes,

$$\sum_{i \,\text{s.t.}\, \tau_{in}^{1,i}(v) \geq t} |\underline{a_{1,i}}|_v \leq \sum_{t' \geq t} F_{env}(v, t').$$

(b) No further rule can be added to any $(R_{m,1}, \ldots, R_{m,r_m})$ so that the resulting family of rules still satisfies condition (a).

And then the simultaneous application of these rules to a configuration $(F_m)_{m \in \overline{M}}$ produces a new configuration $(\widehat{F}_m)_{m \in \overline{M}}$ that is obtained as follows:

(c) For every $m \in M$, for every rule $R_{m,i}$, $i = 1, \ldots, r_m$, and for every $v \in V$, we take mappings $enter_v^{m,i}, exit_v^{m,i} : I^+ \to \mathbb{N}$ in such a way that:
 – $\sum_{t \in I^+} enter_v^{m,i}(t) = |\underline{a_{m,i}}|_v$ and $\sum_{t \in I^+} exit_v^{m,i}(t) = |\underline{b_{m,i}}|_v$.
 – If $t < \tau_{in}^{m,i}(v)$, then $enter_v^{m,i}(t) = 0$, and if $t < \tau_{out}^{m,i}(v)$, then $exit_v^{m,i}(t) = 0$.
 – For every $t \in I^+$,
   • for every elementary membrane m,

$$\sum_{i=1}^{r_m} exit_v^{m,i}(t) \leq F_m(v, t);$$

   • for every non-elementary membrane $m \in M$ with $m = \varepsilon(m_1) = \ldots = \varepsilon(m_l)$,

$$\sum_{i=1}^{r_m} exit_v^{m,i}(t) + \sum_{j=1}^{l} \sum_{i=1}^{r_{m_j}} enter_v^{m_j,i}(t) \leq F_m(v, t);$$

   • $\sum_{i=1}^{r_1} enter_v^{1,i}(t) \leq F_{env}(v, t).$

(d) The new configuration $(\widehat{F}_m)_{m \in \overline{M}}$ produced by this application is obtained as follows: for every $v \in V$ and $t \in I^+$,
  – for every elementary membrane m,

$$\widehat{F}_m(v, t) = F_m(v, t) + \sum_{i=1}^{r_m} enter_v^{m,i}(t) - \sum_{i=1}^{r_m} exit_v^{m,i}(t);$$

  – for every non-elementary membrane $m \in M$ with $m = \varepsilon(m_1) = \ldots = \varepsilon(m_l)$,

$$\widehat{F}_m(v, t) = F_m(v, t) + \sum_{i=1}^{r_m} enter_v^{m,i}(t) + \sum_{j=1}^{l} \sum_{i=1}^{r_{m_j}} exit_v^{m_j,i}(t) \\ - \sum_{i=1}^{r_m} exit_v^{m,i}(t) - \sum_{j=1}^{l} \sum_{i=1}^{r_{m_j}} enter_v^{m_j,i}(t);$$

  – Finally,

$$\widehat{F}_{env}(v, t) = F_{env}(v, t) + \sum_{i=1}^{r_1} exit_v^{1,i}(t) - \sum_{i=1}^{r_1} enter_v^{1,i}(t).$$

A finite sequence of transitions between configurations of a fuzzy P-system Π, starting with the initial configuration, is called a *computation* with respect to Π. A computation *halts* when it reaches a *halting configuration* where no rule can be triggered.

Given a halting computation C with halting configuration $(H(C)_m)_{m \in \overline{M}}$, the (crisp) multiset over I^+ *associated* to it is

$$H_C : I^+ \to \mathbb{N}$$
$$t \mapsto \sum_{v \in V_0} H(C)_{m_{out}}(v, t)$$

Thus, for every $t \in I^+$, $H_C(t)$ is the number of objects in the output region that, at the end of the computation, are copies of some output reactive with degree of exactitude t.

Then, the *output* of a halting computation C will be the fuzzy subset of $\mathbb{N}$

$$Out_{\Pi,C} : \mathbb{N} \to I$$
$$n \mapsto \bigvee \{t \mid H_C(t) = n\}$$

In words, $Out_{\Pi,C}(n)$ is the greatest degree of exactitude t in I for which, at the end of the computation C, there exist n objects in the output region that are copies of some output reactive with degree of exactitude t.

Finally, the fuzzy set of natural numbers *generated* by a fuzzy membrane system Π is the join of all the outputs of halting computations with respect to Π. This is the mapping $Gen_\Pi : \mathbb{N} \to I$ defined by

$$Gen_\Pi(n) = \bigvee_{C \text{ halting}} Out_{\Pi,C}(n), \qquad n \in \mathbb{N}.$$

Thus,

$$Gen_\Pi(n) = \bigvee \left\{ \bigvee \{t \in I^+ \mid H_C(t) = n\} \mid C \text{ halting} \right\}$$
$$= \bigvee \{t \in I^+ \mid H_C(t) = n \text{ for some halting computation } C\}.$$

Notice that, I being finite, this supremum is actually a maximum, and that if $H_C(t) \neq n$ for every halting computation C, then $Gen_\Pi(n) = \bigvee \emptyset = 0$.

The following lemma is a direct consequence of the last description of $Gen_\Pi(n)$ and the finiteness of I.

Lemma 1. *For every fuzzy P-system Π, for every $n \in \mathbb{N}$ and for every $t_0 \in I^+$, $Gen_\Pi(n) \geq t_0$ if and only if there exists some halting computation C and some $t \geq t_0$ such that $H_C(t) = n$.*
□

Crisp P-systems can be seen as special cases of their fuzzy version. Indeed, every crisp P-system

$$\Pi = (V, V_{out}, \mu, m_{out}, (S_m)_{m \in \overline{M}}, (\mathcal{R}_m)_{m \in M})$$

defines a fuzzy P-system

$$\Pi^f = (V, V_{out}, \mu, m_{out}, \{0, 1\}, (S_m^f)_{m \in \overline{M}}, (\mathcal{R}_m^f)_{m \in M})$$

where each fuzzy multiset $S_m^f : V \times \{1\} \to \mathbb{N}_\infty$ is defined from the corresponding crisp multiset $S_m : V \to \mathbb{N}_\infty$ in the natural way: for every $v \in V$, $S_m^f(v, 1) = S_m(v)$. As far as the rules goes, each $\mathcal{R}_m^f$ consists of the rules in $\mathcal{R}_m$ with threshold mappings τ_{in} and τ_{out} that send, respectively, every incoming and every outgoing reactive of the rule to 1 and all other reactives to 0.

Proposition 1. *Let Π be a crisp P-system and Π^f the fuzzy P-system defined by it. Then $Gen_{\Pi^f} = \chi_{Gen_\Pi}$, i.e., $Gen_{\Pi^f}(n) = 1$ if $n \in Gen_\Pi$ and $Gen_{\Pi^f}(n) = 0$ otherwise.*

Proof. Since the configurations for Π^f are $\{0,1\}$-valued, all objects in the membranes and the environment in the initial configuration of Π^f are exact copies of the reactives in V and only exact copies of reactives can enter the skin membrane from the environment. Therefore, at any moment of any computation with respect to Π^f every region defined by a membrane only contains exact copies of reactives. Furthermore, a rule $R \in \mathcal{R}_m$ can be applied to a configuration $(F_m)_{m \in \overline{M}}$ of Π if and only if the corresponding rule R^f in $\mathcal{R}_m^f$ can be triggered in the corresponding configuration $(F_m^f)_{m \in \overline{M}}$ of Π^f (the one defined by $F_m^f(v,1) = F_m(v)$ for every $m \in \overline{M}$ and $v \in V$), and the result of the (unique) application in Π^f of R^f to $(F_m^f)_{m \in \overline{M}}$ is also the configuration corresponding to the result of the application of R in Π to $(F_m)_{m \in \overline{M}}$.

By the formal definitions of transition in crisp and fuzzy P-systems, this argument also entails that every transition for Π defines, in a bijective way, a transition for Π^f, which produces the configuration for Π^f corresponding to the configuration produced by the transition for Π. Thus, every computation C with respect to Π defines, also in a bijective way, a computation with respect to Π^f, which we shall denote by C^f, in such a way that C is halting if and only if C^f is halting and, if they both are halting, $H_{C^f}(1) = Out_{\Pi,C}$.

Therefore, for every halting computation C with respect to Π,

$$Out_{\Pi^f, C^f}(n) = \begin{cases} 1 & \text{if } n = Out_{\Pi,C} \\ 0 & \text{otherwise} \end{cases}$$

and thus

$$\begin{aligned} Gen_{\Pi^f}(n) &= \begin{cases} 1 & \text{if there exists some halting computation } C^f \text{ w.r.t. } \Pi^f \\ & \text{such that } Out_{\Pi^f,C^f}(n) = 1 \\ 0 & \text{otherwise} \end{cases} \\ &= \begin{cases} 1 & \text{if there exists some halting computation } C \text{ w.r.t. } \Pi \\ & \text{such that } Out_{\Pi,C} = n \\ 0 & \text{otherwise} \end{cases} \\ &= \begin{cases} 1 & \text{if } n \in Gen_\Pi \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

as we claimed.

This proposition remains true if we enlarge the set $\{0,1\}$ in the definition of Π^f to any finite subset I of $[0,1]$ containing 0 and 1 and then we set, for every $m \in M$, $S_m^f(v,t) = S_m(v)$ if $t = 1$ and $S_m^f(v,t) = 0$ otherwise, and $S_{env}^f(v,t) = S_{env}(v)$ for every $t \in I^+$, but we still endow all rules with threshold functions that take value 1 on every incoming or outgoing reactive. In this case, at any moment of any computation with respect to Π^f every region defined by a membrane (other than the environment) will only contain exact copies of reactives and hence the proof of the last proposition is still valid.

4 Universality

A *fuzzy language* over an alphabet Σ is a fuzzy subset $L : \Sigma^* \to [0,1]$ of Σ^*. Such a fuzzy language is *recursively enumerable* when all its levels

$$L_t = \{w \in \Sigma^* \mid L(w) \geq t\}, \quad t \in [0,1],$$

are recursively enumerable in the usual sense; cf. [2]. Notice that, since $L_0 = \Sigma^*$, it is enough to consider in this definition the t-levels with $t > 0$. Moreover, arguing as at the beginning of §2.1, we can see that if L is I-valued, then it is enough to consider its t-levels with $t \in I^+$.

Now, in parallel to the definition of a recursively enumerable subset of $\mathbb{N}$ as the set of lengths of some recursively enumerable language, we shall say that a fuzzy subset $F : \mathbb{N} \to [0,1]$ of $\mathbb{N}$ is *recursively enumerable* when there exists some recursively enumerable fuzzy language $L : \Sigma^* \to [0,1]$, over some alphabet Σ, such that, for every $n \in \mathbb{N}$,

$$F(n) = \bigvee \{ L(w) \mid w \in \Sigma^*, \ |w| = n \}.$$

Now we have the following characterization of recursively enumerable fuzzy subsets of $\mathbb{N}$ in terms of levels, which is the one we shall use henceforth.

Proposition 2. *An I-valued fuzzy subset $F : \mathbb{N} \to I$ is recursively enumerable if and only if F_t is a recursively enumerable subset of $\mathbb{N}$, for every $t \in I^+$.*

Proof. Let $L : \Sigma^* \to I$ be a recursively enumerable fuzzy language such that, for every $n \in \mathbb{N}$,

$$F(n) = \bigvee \{ L(w) \mid w \in \Sigma^*, \ |w| = n \}.$$

Then, for every $t \in I^+$,

$$
\begin{aligned}
F_t &= \{ n \in \mathbb{N} \mid \bigvee \{ L(w) \mid w \in \Sigma^*, \ |w| = n \} \geq t \} \\
&= \{ n \mid \text{there exists some } w \in \Sigma^* \text{ with } |w| = n \text{ such that } L(w) \geq t \} \\
&= \{ |w| \mid w \in L_t \}
\end{aligned}
$$

is the set of lengths of a recursively enumerable language, and hence recursively enumerable itself.

Conversely, let $F : \mathbb{N} \to I$ be a fuzzy subset of $\mathbb{N}$ such that each F_t is recursively enumerable, and consider the fuzzy language over a singleton $\{a\}$

$$
\begin{aligned}
L : \{a\}^* &\to I \\
a^n &\mapsto F(n)
\end{aligned}
$$

It is clear that $L_t = \{ a^n \mid n \in F_t \}$ and hence, if every F_t with $t \in I^+$ is recursively enumerable, the same happens for each L_t with $t \in I^+$. Therefore, L is a recursively enumerable fuzzy language. And it is also clear that, for every n,

$$F(n) = L(a^n) = \bigvee \{ L(w) \mid w \in \{a\}^*, \ |w| = n \},$$

which finally implies that F is recursively enumerable, too.

Our goal now is to prove that a finite-valued fuzzy subset of $\mathbb{N}$ is recursively enumerable if and only if it is generated by a fuzzy P-system. We begin with the easy implication in this equivalence.

Theorem 2. *Every fuzzy subset of $\mathbb{N}$ generated by a fuzzy P-system is recursively enumerable.*

Proof. Let

$$\Pi = (V, V_{out}, \mu, m_{out}, I, (S_m)_{m \in \overline{M}}, (\mathcal{R}_m)_{m \in M})$$

be a fuzzy P-system. For every $t \in I^+$, let $\Pi^{(t)}$ be the crisp P-system

$$\Pi^{(t)} = (V \times I^+, V_{out} \times \{t\}, \mu, m_{out}, (S_m)_{m \in \overline{M}}, (\mathcal{R}_m^c)_{m \in M}),$$

where each $S_m : V \times I^+ \to \mathbb{N}$ is now understood as a multiset over $V \times I^+$ and, for every $m \in M$, the set of rules $\mathcal{R}_m^c$ contains, for each $((\underline{a}, in; \underline{b}, out), \tau_{in}, \tau_{out}) \in \mathcal{R}_m$, say with $\underline{a} = a_1 \ldots a_p$ and $\underline{b} = b_1 \ldots b_q$, each possible rule of the form

$$\Big((a_1, t_{i_1}) \ldots (a_p, t_{i_p}), in; (b_1, t_{j_1}) \ldots (b_q, t_{j_q}), out\Big)$$

with $t_{i_1}, \ldots, t_{i_p}, t_{j_1}, \ldots, t_{j_q} \in I^+$ such that $t_{i_k} \geq \tau_{in}(a_k)$ for every $k = 1, \ldots, p$ and $t_{j_l} \geq \tau_{out}(b_l)$ for every $l = 1, \ldots, q$, and it only contains rules obtained in this way. Notice thus that these membrane systems $\Pi^{(t)}$ only differ in their sets of output reactives, and hence they have exactly the same halting computations, but any such halting computation may produce in each $\Pi^{(t)}$ a different output.

Now, we can identify the configurations for Π with the configurations for each $\Pi^{(t)}$, by simply understanding a fuzzy multiset over V as a multiset over $V \times I^+$. We can also identify each application of a rule $R \in \mathcal{R}_m$ to a configuration for Π with the application of some rule contributed by R in $\mathcal{R}_m^c$ (a different rule for each application) to the corresponding configuration for each $\Pi^{(t)}$. From the explicit description of transitions for crisp and fuzzy P-systems, we deduce that we can actually identify each transition for Π from a configuration $(F_m)_{m \in \overline{M}}$ to a configuration $(\widehat{F}_m)_{m \in \overline{M}}$ with a transition for each $\Pi^{(t)}$ from the configuration corresponding to $(F_m)_{m \in \overline{M}}$ to the configuration corresponding to $(\widehat{F}_m)_{m \in \overline{M}}$. This finally entails that every halting computation with respect to any $\Pi^{(t)}$ corresponds to a halting computation with respect to Π.

Now, let us fix an arbitrary $t_0 \in I^+$; we want to prove that the t_0-level $(Gen_\Pi)_{t_0}$ is recursively enumerable. For every halting computation C with respect to Π, let $H_C : I^+ \to \mathbb{N}$ be the multiset on I^+ associated to it, and let $h_C^{(t_0)} \in \mathbb{N}$ be the output of the corresponding halting computation with respect to $\Pi^{(t_0)}$. Since the output set of reactives of $\Pi^{(t_0)}$ is $V_{out} \times \{t_0\}$, it is clear that $h_C^{(t_0)} = H_C(t_0)$.

Now, by Lemma 1, we have that $Gen_\Pi(n) \geq t_0$ if and only if $h_C^{(t)} = n$ for some halting computation C and some $t \geq t_0$. Thus, $n \in (Gen_\Pi)_{t_0}$ if and only if $n \in Gen_{\Pi^{(t)}}$ for some $t \geq t_0$, i.e.

$$(Gen_\Pi)_{t_0} = \bigcup_{t \geq t_0} Gen_{\Pi^{(t)}}.$$

Since every $Gen_{\Pi^{(t)}}$ is a recursively enumerable subset of $\mathbb{N}$ and there are only a finite number of them, this implies that $(Gen_\Pi)_{t_0}$ is recursively enumerable, too. And since t_0 was arbitrary, this shows that Gen_Π is a recursively enumerable fuzzy subset of $\mathbb{N}$.

The converse implication is given by the following result.

Theorem 3. *Every recursively enumerable finite-valued fuzzy subset of $\mathbb{N}$ is generated by a fuzzy P-system.*

Proof. Let $F : \mathbb{N} \to I$ be a recursively enumerable I-valued fuzzy subset of $\mathbb{N}$, with I finite and containing 0 and 1. Each level F_t, with $t \in I^+$, is recursively enumerable, and therefore, by Theorem 1, it is generated by a P-system

$$\Pi^{(t)} = (V^{(t)}, V_{out}^{(t)}, \mu, m_{out}, (S_m^{(t)})_{m \in \overline{M}}, (\mathcal{R}_m^{(t)})_{m \in M})$$

that satisfies the conditions listed in that theorem: $V_{out}^{(t)} = V^{(t)}$; it only has symport rules; it has only two membranes, the skin membrane 1 and the output membrane $m_{out} = 2$; the only rules associated to the output membrane are $(\alpha^{(t)}, in)$, $(\#^{(t)}, in)$ and $(\#^{(t)}, out)$ for some specific reactives $\alpha^{(t)}$ and $\#^{(t)}$; $\alpha^{(t)}$ is the only reactive that may enter the output membrane in any halting computation; and $S_1^{(t)}(\alpha^{(t)}) = S_2^{(t)}(\alpha^{(t)}) = 0$. We shall assume that the sets of reactives $V^{(t)}$ are pairwise disjoint.

Now, consider the fuzzy P-system

$$\Pi = (V, V_{out}, \mu', m_{out}, I, (S_m)_{m \in \overline{M}}, (\mathcal{R}_m)_{m \in M})$$

defined as follows:

- $V = \bigsqcup_{t \in I^+} V^{(t)}$.
- $V_{out} = V$.
- μ' is a linear tree obtained from μ by adding a third membrane, labelled 3, as the new elementary membrane.
- $m_{out} = 2$.
- For every $m = 1, 2$, for every $v \in V$, and for every $t, t' \in I^+$,

$$S_m(v, t') = \begin{cases} S_m^{(t)}(v) & \text{if } v \in V^{(t)} \text{ and } t' = 1 \\ 0 & \text{otherwise} \end{cases}$$

  And set $S_3(v, t) = 0$ for every $v \in V$ and $t \in I^+$.
  Thus, membranes 1 and 2 in Π contain the sum of their contents in all $\Pi^{(t)}$, with all objects being exact copies of the corresponding reactives, while the new membrane 3 is empty at the beginning. Notice in particular that, for every $t, t' \in I^+$ and for every $i = 1, 2, 3$, $S_i(\alpha^{(t)}, t') = 0$.
  Finally, as far as S_{env} goes, if $v \in V^{(t)}$, then $S_{env}(v, t') = S_{env}^{(t)}(v)$ for every $t' \in I^+$. I.e., the environment contains an unbounded number of copies of v in $\Pi^{(t)}$ exactly when the environment in Π contains an unbounded number of copies of this reactive with any non-zero degree of exactitude.
- For every $m = 1, 2$, and for every rule $(\underline{a}, in)$ or $(\underline{a}, out)$ in some $\mathcal{R}_m^{(t)}$, the set $\mathcal{R}_m$ contains a corresponding rule

$$((\underline{a}, in), \tau) \text{ or } ((\underline{a}, out), \tau)$$

  where the threshold mapping τ is defined as follows: for every $v \in V$, if $|\underline{a}|_v = 0$, then $\tau(v) = 0$, and if $|\underline{a}|_v > 0$, then

$$\tau(v) = \begin{cases} t \text{ if } v = \alpha^{(t)} \text{ for some } t \in I^+ \\ 1 \text{ otherwise} \end{cases}$$

  On the other hand, if for every $t \in I^+ - \{1\}$ we denote by $s(t)$ the least element in I greater than t (which exists because I is finite), then $\mathcal{R}_3$ contains, for every $t \in I^+ - \{1\}$, a rule

$$((\alpha^{(t)}, in), \tau)$$

  with $\tau(\alpha^{(t)}) = s(t)$.
  And $\mathcal{R}_m$, $m = 1, 2, 3$, do not contain any rule other than these ones.

To simplify the notations, set

$$V_a = \{\alpha^{(t)} \mid t \in I^+\}.$$

Thus, at the beginning, all membranes in Π other than the environment only contain exact copies of non-output reactives. Furthermore, Π contains rules of two types. There are rules induced from rules in some $\Pi^{(t)}$ that move exact copies of reactives in $V - V_a$ as well as copies of reactives $\alpha^{(t)}$ with degree of similarity at least t, in the same way as the corresponding rule moved them in $\Pi^{(t)}$. And rules that remove from the output membrane all copies of reactives $\alpha^{(t)}$ with degree of similarity greater than the corresponding t and bury them in the elementary membrane.

In particular, no non-exact copy of a reactive in $V - V_a$ may ever enter the skin membrane from the environment, and the only objects similar to some $\alpha^{(t)}$ that enter it must have degree of accuracy at least t. Therefore, all objects that, at some moment of a computation with respect to Π, are contained in some membrane other than the environment, are either exact copies of reactives in $V - V_a$ or similar to some $\alpha^{(t)}$ with degree of exactitude at least t. Moreover, no copy of a reactive $\alpha^{(t)}$ with degree of similarity greater than t may remain in the output membrane when a computation halts.

Now, the fact that each rule involves only reactives in some $V^{(t)}$ and the form of the rules in each $\mathcal{R}_2^{(t)}$ imply that in a given configuration for Π, the application conditions for a rule coming from $\Pi^{(t)}$, a rule coming from $\Pi^{(t')}$ with $t \neq t'$ and any one of the new rules in $\mathcal{R}_3$ are independent of each other.

Using these remarks, one can easily see that every transition with respect to Π consists of the application in parallel of families of rules coming from rules that form transitions with respect to some P-systems $\Pi^{(t)}$ plus the application of all rules in $\mathcal{R}_3$ necessary to remove from the output membrane all copies of reactives $\alpha^{(t)}$ with degree of exactitude greater than t. Therefore, every halting computation with respect to Π corresponds to a family of halting computations $(C_t)_{t \in I^+}$ performed in parallel, every C_t with respect to the corresponding $\Pi^{(t)}$. These computations C_t are uniquely determined by C, and they may halt at different moments: C halts when all computations C_t halt and no copy of any $\alpha^{(t)}$ with degree of exactitude greater than t remains in the output membrane.

Then, the output component $H(C)_2$ of the halting configuration of a halting computation C corresponding to a family of halting computations $(C_t)_{t \in I^+}$ with respect to the P-systems $(\Pi^{(t)})_{t \in I^+}$ satisfies that, for every $t, t' \in I^+$,

$$H(C)_2(v,t) = \begin{cases} (H_{C_t})_2(\alpha^{(t)}) & \text{if } v = \alpha^{(t)} \\ 0 & \text{otherwise} \end{cases}$$

Hence, the mapping H_C associated to this halting computation C is given by

$$H_C(t) = (H_{C_t})_2(\alpha^{(t)}) = Out_{\Pi^{(t)}, C_t}.$$

Now, by Lemma 1, for every $t_0 \in I^+$, $Gen_\Pi(n) \geq t_0$ if and only if there exists some halting computation C with respect to Π, corresponding to some family of halting computations $(C_t)_{t \in I^+}$, and some $t' \geq t_0$ such that $H_C(t') = n$. Since, moreover, every halting computation with respect to any $\Pi^{(t')}$ will be part of some halting computation with respect to Π, this condition is equivalent to the existence of some $t' \geq t_0$ and some halting computation $C_{t'}$ with respect to $\Pi^{(t')}$ such that $Out_{\Pi^{(t')}, C_{t'}} = n$, i.e., to the fact that $n \in Gen_{\Pi^{(t')}}$ for some $t' \geq t_0$. In all, this shows that

$$(Gen_\Pi)_{t_0} = \bigcup_{t' \geq t_0} Gen(\Pi^{(t')}) = \bigcup_{t' \geq t_0} F_{t'}.$$

And finally, since the t-levels of F are decreasing in t, this entails that

$$(Gen_\Pi)_{t_0} = F_{t_0}.$$

Thus, Gen_Π and F have exactly the same levels, and therefore they are the same fuzzy subset of $\mathbb{N}$.

5 Conclusion

In this paper we have introduced a fuzzy version of the symport/antiport model of membrane systems that uses inexact copies of reactives in the transitions. Then, we have proved that this fuzzy model of computation is universal, in the sense that it generates all recursively enumerable finite-valued fuzzy subsets of $\mathbb{N}$. This means a first step towards the use of fuzzy methods to answer a question posed by Gh. Păun in the last problem of his first list of open problems in membrane computing [10]: "What about 'approximate' computing, whatever this can mean?"

The key ingredients in our model are the use of fuzzy multisets in configurations, the endowment of evolution rules with threshold mappings that determine the degree of exactitude of objects to reactives in order to be affected by the rules, and an appropriate way of evaluating the content of the output membrane at the end of a halting computation. These ingredients could also be used *mutatis mutandis* to define the fuzzy version of any other membrane computing model: in this paper we only considered the symport/antiport model for simplicity.

When proving the universality of our fuzzy P-systems we have not addressed any minimality question like the least number of membranes or the least number of output reactives that are necessary to generate all recursively enumerable finite-valued fuzzy subsets of $\mathbb{N}$: we leave these as open problems. Nevertheless, let us mention here that we have specified a set of output reactives in our P-systems with the only purpose of simplifying the proof of Theorem 2. Nowhere else in this paper it is needed and, as one would expect, that theorem can also be proved without distinguishing output reactives, but at the prize of using more involved crisp P-systems: this proof will appear elsewhere [5].

We would like to point out here that the finite-valuedness of the fuzzy subsets of $\mathbb{N}$ generated by our fuzzy P-systems is not due to the specification beforehand of the finite set of possible values I, but rather to the finiteness of the sets of rules and the initial configuration. This effect also appears for instance in fuzzy grammars where only a finite number of rules have a non-zero weight.

Indeed, assume for the moment that such an I is not specified and that in all fuzzy subsets and multisets used in the definition of a fuzzy P-system Π, as well as in the description of how it works, all sets I and I^+ are replaced by $[0,1]$ and $]0,1]$, respectively.

Since the ensemble of evolution rules $\bigcup_{m \in M} \mathcal{R}_m$ in Π is finite and each rule only involves a finite set of reactives, it is clear that the set of the images of all threshold functions of all rules is a finite subset of $[0,1]$. Moreover, similarly to the crisp case, we would assume that the initial content of regions other than the environment is finite (in our definition it is entailed by the fact that I is finite), and thus we would impose that the initial configuration is given by finite-valued fuzzy multisets S_m for every $m \in M$ (but not for the environment). Hence, the set of possible exactitude values of objects to reactives in the initial contents of the regions K_m described by these multisets form a finite subset of $[0,1]$. Let the union of these two finite sets be $J = \{t_0, t_1, \ldots, t_m\}$ with $t_0 < t_1 < \ldots < t_m$, and assume for simplicity that $t_0 = 0$,

$t_m = 1$. To the effect of triggering a rule, any two copies of the same reactive with degrees of accuracy t, t' in some interval $[t_l, t_{l+1}[$, $l = 0, \ldots, m-1$, are indistinguishable, and any object with degree of accuracy $t \in]t_l, t_{l+1}[$ to some reactive must come from the environment.

Additionally, we still must impose that the environment contains an unbounded homogeneous supply of some reactives. In order not to distinguish any degree of similarity, we would impose it by assuming that $S_{env}(v, t)$ is either 0 for every $t \in]0, 1]$ or ∞ for every $t \in]0, 1]$.

This would entail that, for every $t, t' \in]t_l, t_{l+1}[$, $l = 0, \ldots, m-1$, and for every computation C with respect to Π, there exists a computation C' such that $H_{C'}(t') = H_C(t)$. Indeed, C' has the same ordered sequence of families of rules as C, but for every v and every application of a rule R, $enter_v^R(t')$ and $exit_v^R(t')$ in the application in C' take the values of $enter_v^R(t)$ and $exit_v^R(t)$ in the corresponding application in C, and $enter_v^R(t)$ and $exit_v^R(t)$ in the application in C' take the values of $enter_v^R(t')$ and $exit_v^R(t')$ in the corresponding application in C.

Therefore, if some $t \in]t_l, t_{l+1}[$ is contained in

$$\{t \in]0, 1] \mid H_C(t) = n \text{ for some halting computation } C\}$$

for some $n \in \mathbb{N}$, then the whole interval $[t_l, t_{l+1}[$ is contained in this set. This entails that the supremum of this set, which would define $Gen_\Pi(n)$, belongs to J. Thus, after all, the fuzzy set Gen_Π is still finite-valued, and moreover its set of values is contained in J.

Besides, $S_{env}(v, t) = \infty$ for some $v \in V$ and every $t \in]0, 1]$ entails that the universe in non-countably infinite, and that a non-countably infinite number of computations may exist.

These observations, and the obvious fact that working with fuzzy sets and multisets that are explicitly specified as finite-valued greatly simplifies all notations, definitions and proofs, motivated us to restrict ourselves from the very beginning to the finite set J, or rather a finite extension I of it, as the set of values of any fuzzy set related to the fuzzy P-system Π.

To end this paper, we would like to point out that, although formally correct, our specific approach has a drawback from the fuzzy mathematics point of view. The association to a multiset $H : I^+ \to \mathbb{N}$ of the fuzzy subset of $\mathbb{N}$

$$\mathcal{C}(H) : \mathbb{N} \to I$$
$$n \mapsto \bigvee \{t \mid H(t) = n\}$$

that underlies our definition of the output of a halting computation with respect to a fuzzy P-system is not additive in any natural sense, and in particular it cannot be considered a fuzzy cardinality; see [1]. We have tried to use some specific simple fuzzy cardinalities in this step, and we have obtained that the resulting fuzzy P-systems did not generate all finite-valued recursively enumerable fuzzy subsets of $\mathbb{N}$, but we have not ruled out the possibility of using some other, cunningly chosen, fuzzy cardinality. Our current research agenda includes this problem, as well as the problem of getting rid of the assumption used in this paper that an object can only be similar to one reactive.

References

1. Casasnovas, J., Rosselló, F.: Scalar and fuzzy cardinalities of crisp and fuzzy multisets, submitted.

2. Gerla, G.: Fuzzy grammars and recursively enumerable fuzzy languages, *Inform. Sci.*, **60**, 1992, 137–143.

3. Klir, G.J., Yuan, B.: *Fuzzy sets and fuzzy logic. Theory and applications*, Prentice-Hall, 1995.

4. Martín-Vide, C., Păun, A., Păun, Gh.: On the power of P systems with symport rules, *J. Universal Computer Science*, **8**, 2002, 317–331.

5. Moyà, M.: *PhD Thesis*, forthcoming.

6. Obtułowicz, A.: Mathematical models of uncertainty with a regard to membrane systems, in: *Proceedings of the First Brainstorming Week on Membrane Computing* (M. Cavaliere, C. Martín-Vide, Gh. Păun, Eds.), 2003, 241–246. Available at `http://psystems.disco.unimib.it/download/rep26.pdf`

7. Obtułowicz, A., Păun, Gh.: (In search of) probabilistic P systems, *BioSystems* **70**, 2003, 107–121.

8. Păun, A., Păun, Gh.: The power of communication: P systems with symport/antiport, *New Generation Computers*, **20**, 2002, 295–305.

9. Păun, Gh.: Computing with membranes, *J. of Comp. and Syst. Sci.*, **61**, 2000, 108–143.

10. Păun, Gh.:, Computing with membranes (P Systems): Twenty six research topics, available at `http://psystems.disco.unimib.it/download/probl.pdf`.

11. Păun, Gh.: *Membrane Computing. An Introduction*, Springer-Verlag, 2002.

12. Păun, Gh.: Problems circulated before the Brainstorming Week on Membrane Computing, available at `http://psystems.disco.unimib.it/download/rep26.pdf`.