Computing and Informatics, Vol. 27, 2008, 497–513

HIERARCHICAL CLUSTERING WITH MEMBRANE COMPUTING

Mónica Cardona, M. Angels Colomer, Alba Zaragoza

Department of Mathematics
University of Lleida
Av. Alcalde Rovira Roure, 191. 25198 Lleida, Spain
e-mail: {colomer, alba, mcardona}@matematica.udl.es

Mario J. Pérez-Jiménez

Research Group on Matural Computing
Department of Computer Science and Artificial Intelligence
University of Sevilla
Avda. Reina Mercedes s/n, 41012 Sevilla, Spain
e-mail: marper@us.es

Revised manuscript received 3 December 2007

Abstract. In this paper we approach the problem of hierarchical clustering through membrane computing. A specific P system with external output is designed for each Boolean matrix associated with a finite set of individuals. The computation of the system allows us to obtain one of the possible classifications in a non-deterministic way. The amount of resources required in the construction is polynomial in the number of individuals and of characteristics analyzed.

Keywords: P systems, hierarchical clustering

1 INTRODUCTION

Many scientific investigations depend on many factors and this makes such investigations very complex. In order to simplify the problems and make them more

tractable it is necessary to group individuals with similar characteristics. The individuals are characterized by a high number of properties so the clustering according to their degree of similarity is not a simple task.

There are different methods of ranking the groups of individuals. We mention two types only here, the non-hierarchical and the hierarchical clustering. In non-hierarchical clustering homogenous groups are formed without establishing relations among them; in hierarchical clustering the individuals are grouped in levels. The inferior levels are contained in the superior levels. Hierarchical clustering is mostly used and it is dealt with in this paper.

Hierarchical clustering refers to the formation of a recursive clustering of the individuals by means of the partitions $P_0, P_1, \ldots, P_m$ of the set of N individuals with $1 \leq m \leq N-1$. The partition P_0 consists of N groups, each one of them formed by a single individual. The groups that form this partition join progressively until arriving at the last partition, P_m, that consists of a single group formed by all the individuals. In each step two most similar groups are joined according to a previously established criterion.

Researchers use clustering to characterize and to order a vast amount of information about variability of population of individuals. These populations are grouped in more or less homogenous clusters based on their properties. This methodology has been applied in fields as diverse as medicine, biology, classification of words and of fingerprints, artificial intelligence, etc. Recently clustering has been applied to the classification of musical genre [13], to predict essential hypertension [12], in the classification of material planning and control systems [9], of the ocean color [1], of the plants gens [14].

The different groups obtained by means of the classification are characterized by different levels of the measured variables. These values allow us to give common properties of the individuals belonging to the same group. The fact of having established groups allows us to identify the most similar cluster of a new individual. The characteristics measured of the individuals can be qualitative or quantitative variables. In most cases we are only interested in the presence or absence of certain qualitative characteristics. Thus, in this paper we consider the hierarchical clustering using dichotomizing variables, and this problem is approached within the framework of cellular computing with membranes. This approach is interesting because it allows us to treat some statistical topics within this new model of computation. The amount of used resources is polynomial in the number of individuals and the number of characteristics analyzed.

In the following, we assume that the reader is familiar with the basic notions of membrane computing; for details, refer to [6, 7, 8, 15].

2 HIERARCHICAL CLUSTERING

In order to obtain a hierarchical clustering we firstly need a set of individuals or observations.

Definition 1. A k-set over a metric space (E, d), with $d(E \times E) \subseteq \mathbb{N}$ is a subset of E^k.

The elements of a finite k-set, $\Omega = \{\omega_1, \ldots, \omega_N\}$, are called individuals. The components of the individual ω_i (called characteristics or variables) is denoted by $\omega_{i1}, \ldots, \omega_{ik}$. Thus, the set of the individuals can be represented by the matrix $P_{Nk} = (\omega_{ij})_{1 \leq i \leq N, 1 \leq j \leq k}$.

The objective of clustering is to gather the individuals in similar groups whose members are all close to each other with various dimensions being measured. It will be necessary to establish criteria in order to measure the similarity between individuals and between groups. Obviously, the clustering that is obtained will depend on the similarity function that is chosen [10].

Definition 2. A similarity over a finite k–set, $\Omega = \{\omega_1, \ldots, \omega_N\}$, is a function s of $\Omega \times \Omega$ in $\mathbb{R}^+$ that verifies

- s is symmetric, that is $\forall(\omega_i, \omega_j) \in \Omega \times \Omega : \quad s(\omega_i, \omega_j) = s(\omega_j, \omega_i)$
- $\forall \omega_i, \omega_j \in \Omega$ with $i \neq j : \quad s(\omega_i, \omega_i) = s(\omega_j, \omega_j) \geq s(\omega_i, \omega_j)$.

In this paper we work with dichotomizing variables; their values are denoted by 0 and 1. One of the similarities most used for dichotomizing variables is that proposed by Sokal and Michener [2] and it is defined by:

$$s'(\omega_i, \omega_j) = \frac{1}{k} \cdot \sum_{r=1}^{k} (1 - |\omega_{ir} - \omega_{jr}|), \text{ for each } (\omega_i, \omega_j) \in \Omega \times \Omega. \tag{1}$$

In this paper the similarity that we use is a modification of the previous one. It represents the number of coincidences in the number of total characteristics and it is defined as follows:

$$s(\omega_i, \omega_j) = \sum_{r=1}^{k} (1 - |\omega_{ir} - \omega_{jr}|), \text{ for each } (\omega_i, \omega_j) \in \Omega \times \Omega. \tag{2}$$

We use this similarity because it is easier to implement with P systems and the result obtained is the same as those obtained with the similarity of Sokal and Michener.

In the case of the hierarchical clustering the groupings follow a hierarchy formed by partitions $P_0, P_1, \ldots, P_m$ that are called *clusterings*, and verify

$$P_0 \subseteq P_1 \subseteq P_2 \subseteq \ldots \subseteq P_m$$

with $1 \leq m \leq N - 1$. The sets that belong to the partitions are called *clusters*. The clusterings are constructed in a recursive manner. P_0 is formed by as many clusters as individuals. The following partitions are obtained by joining the two closest clusters belonging in the previous one. This process is done until we obtain a partition, P_m, with a single set formed by all the individuals.

Next we define the necessary mathematical concepts in the hierarchical clustering [11].

Definition 3. Let $\Omega = \{\omega_1, \ldots, \omega_N\}$ the k-set of N individuals. A subset H of the power set $\mathcal{P}(\Omega)$, is a *hierarchy* over Ω if it verifies:

- $\Omega \in H$
- $\{\omega\} \in H \quad (\forall \omega \in \Omega)$
- If $h \cap h' \neq \emptyset \Rightarrow h \subseteq h'$ *or* $h' \subseteq h$ $(\forall h, h' \in H)$
- $\bigcup \{h' \mid h' \in H,\ h' \subsetneq h\} \in \{h, \emptyset\}$ $(\forall h \in H)$.

The elements of H are called clusters. If $h_1, \ldots, h_p \in H$ with $\Omega = h_1 \cup \ldots \cup h_p$, then the set $\{h_1, \ldots, h_p\}$ is a clustering.

In order to construct a hierarchy it is necessary to have a similarity between individuals and another function that measures the similarity between clusters. The second function is called the aggregation index.

Definition 4. A symmetrical and nonnegative application δ from $\mathcal{P}(\Omega) \times \mathcal{P}(\Omega)$ to $\mathbf{R}$ is called an aggregation index between clusters if:

- $\delta(h_1, h_2) \geq 0$, for each $h_1, h_2 \in \mathcal{P}(\Omega)$.
- $\delta(h_1, h_2) = \delta(h_2, h_1)$, for each $h_1, h_2 \in \mathcal{P}(\Omega)$.

There are several aggregation indices [4] that depend on the similarity s chosen. In this paper we use the aggregation index based on the minimum [5], defined by:

$$\delta(h_1, h_2) = \min \{s(\omega_i, \omega_j) \mid \omega_i \in h_1,\ \omega_j \in h_2\} \tag{3}$$

A hierarchy has associated an index that measures the homogeneity degree between the individuals belonging to the same cluster, and it is called hierarchical index. This index is always obtained by means of the aggregation index. In this paper we define the hierarchical index of a new cluster h obtained from the union of two clusters $h = h_1 \cup h_2$, by means of $f(h) = \delta(h_1, h_2)$.

2.1 An Algorithm for the Construction of a Hierarchy

The algorithms that are used to obtain a hierarchy have the same structure, the only differences are the way to compute the similarities between individuals and the aggregation index between clusters [3].

In this paper we consider an algorithm whose input is a finite k-set Ω, the similarity s, and the aggregation index δ. The output is an indexed hierarchy (H, f).

1. Place each individual of Ω in its own cluster (singleton), creating the list of clusters $L = P_0 = \{S_1 = \{\omega_1\}, S_2 = \{\omega_2\}, \ldots, S_N = \{\omega_N\}\}$. In this moment $\delta(S_i, S_j) = s(\omega_i, \omega_j)$, and $f(S_i) = k$ $(1 \leq i < j \leq N)$.

2. Find the two closest clusters S_i, S_j $(1 \leq i < j \leq N)$, which will form a new class $S_i = S_i \cup S_j$.
3. Remove S_j from L.
4. Compute the aggregation index between all the pair of clusters in L by using equation (3).
5. Go to step 2 until there is only one set remaining.

Remark 1. If at step 2 there are more than one possibility, then one of them is chosen at random. So, the hierarchy obtained is not unique.

3 HIERARCHICAL CLUSTERING BY MEANS OF MEMBRANE COMPUTING

3.1 Designing a P System

The goal of this paper is to obtain one hierarchical clustering of a finite k-set Ω, of N different individuals by using P systems. Each individual $\omega_i \in \Omega \subseteq \{0,1\}^k$ is denoted by $\omega_i = (\omega_{i1}, \omega_{i2}, \ldots, \omega_{ik})$, and we consider the similarity between individuals defined by (2).

Let $P_{Nk} = (\omega_{ij})_{1 \leq i \leq N, 1 \leq j \leq k}$ be the matrix associated with the N individuals to classify. We define the P system of degree N with external output,

$$\Pi(P_{Nk}) = (\Gamma(P_{Nk}), \mu(P_{Nk}), \mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_{N-1}, \mathcal{M}_N, R, \rho)$$

associated with the matrix P_{Nk}, as follows:

- Working alphabet:

$$\begin{aligned}\Gamma(P_{Nk}) = \; & \{e_{js}, d_{js} : \; 1 \leq j \leq N, 1 \leq s \leq k\} \cup \{a_s, b_s : \; 1 \leq s \leq k\} \cup \\ & \{S_{ij}, C_{ij} : \; 1 \leq i < j \leq N\} \cup \{\beta_i : 0 \leq i \leq k-2\} \cup \\ & \{\alpha_{ijt}, X_{ijt} : \; 1 \leq i < j \leq N, \; 1 \leq t \leq k-1\} \cup \{\gamma_i : \; 1 \leq i \leq N\} \cup \\ & \{\epsilon_i : \; 0 \leq i \leq 3k-2\} \cup \{\eta_i : \; 0 \leq i \leq (N-1)(3k-1)\} \cup \{\sharp\}\end{aligned}$$

- Membrane structure: $\mu(P_{Nk}) = [_N \; [_1 \;]_1 \; [_2 \;]_2 \; \ldots \; [_{N-1} \;]_{N-1} \;]_N$.
- Initial multisets:

$$\begin{aligned}\mathcal{M}_i = \; & \left\{a_s^{(N-i)\omega_{is}} : \; 1 \leq s \leq k \; \wedge \; 1 \leq i \leq N-1\right\} \cup \\ & \left\{b_s^{(N-i)(1-\omega_{is})} : \; 1 \leq s \leq k \; \wedge \; 1 \leq i \leq N-1\right\} \cup \\ & \left\{e_{js}^{\omega_{js}} : \; 1 \leq s \leq k \; \wedge \; i \leq j \leq N\right\} \cup \\ & \left\{d_{js}^{(1-\omega_{js})} : \; 1 \leq s \leq k \; \wedge \; i \leq j \leq N\right\} \; ; \quad 1 \leq i \leq N-1\end{aligned}$$

$$\mathcal{M}_N = \{\gamma_N, \epsilon_0, \eta_0\} \, ;$$

- The set R consists of the following rules:

  - Rules in the skin membrane:

$$
\begin{aligned}
r_0 = \quad & \{\epsilon_0 \to \epsilon_1\beta_0\} \cup \{\epsilon_i \to \epsilon_{i+1} : 1 \leq i \leq 3k-2 \wedge i \neq k\} \cup \\
& \{\eta_i \to \eta_{i+1} : 0 \leq i \leq (N-1)(3k-1)-1\} \\
r_u = \quad & \left\{\beta_{u-1} S_{ij}^{k-u} \to \alpha_{ij(k-u)} : \ 1 \leq i < j \leq N\right\} \quad 1 \leq u \leq k-1 \\
r'_u = \quad & \{\beta_{u-1} \to \beta_u\} \quad 1 \leq u \leq k-1 \\
r'_{k-1} = \quad & \left\{\eta_{(N-1)(3k-1)} \to (\sharp, out)\right\} \\
r_k = \quad & \{\epsilon_k \gamma_q \alpha_{ijt} \to \epsilon_{k+1} X_{ijt}^{q-2} \gamma_{q-1}(X_{ijt}, out) : \ 2 \leq q \leq N, \\
& 1 \leq i < j \leq N, \ 1 \leq t \leq k-1\} \\
r'_k = \quad & \left\{\epsilon_k \to \epsilon_{k+1}\right\} \\
r_{k+1} = \quad & \{X_{ijt} S_{ip} S_{jp} \to C_{ip} X_{ijt} : \ 1 \leq i < j < p \leq N, \ 1 \leq t \leq k-1\} \cup \\
& \{X_{ijt} S_{ip} S_{pj} \to C_{ip} X_{ijt} : \ 1 \leq i < p < j \leq N, \ 1 \leq t \leq k-1\} \cup \\
& \{X_{ijt} S_{pi} S_{pj} \to C_{pi} X_{ijt} : \ 1 \leq p < i < j \leq N, \ 1 \leq t \leq k-1\} \\
r_{k+2} = \quad & \{X_{ijt} S_{ip} \to X_{ijt} : \ 1 \leq i < p < j \leq N, \ 1 \leq t \leq k-1\} \cup \\
& \{X_{ijt} S_{jp} \to X_{ijt} : \ 1 \leq i < j < p \leq N, \ 1 \leq t \leq k-1\} \cup \\
& \{X_{ijt} S_{pi} \to X_{ijt} : \ 1 \leq p < i < j \leq N, \ 1 \leq t \leq k-1\} \cup \\
& \{X_{ijt} S_{pj} \to X_{ijt} : \ 1 \leq p < i < j \leq N, \ 1 \leq t \leq k-1\} \cup \\
& \{X_{ijt} S_{pj} \to X_{ijt} : \ 1 \leq i < p < j \leq N, \ 1 \leq t \leq k-1\} \\
r_{k+3} = \quad & \{C_{ij} \to S_{ij} : \ 1 \leq i < j \leq N\} \cup \\
& \left\{\epsilon_{3k-1} X_{ijt}^{q-2} \gamma_{q-1} \to \epsilon_1 \beta_0 \gamma_{q-1} : \ 1 \leq i < j \leq N, \ 1 \leq t \leq k-1\right\} \\
r'_{k+3} = \quad & \left\{\epsilon_{3k-1} \to \epsilon_1 \beta_0\right\}
\end{aligned}
$$

  - Rules in the membrane labelled by i $\{1 \leq i \leq N-1\}$:
    $r_{k+4} = \{a_s e_{js} \to (S_{ij}, out) : \ 1 \leq s \leq k, \ i+1 \leq j \leq N\}$
    $r_{k+5} = \{b_s d_{js} \to (S_{ij}, out) : \ 1 \leq s \leq k, \ i+1 \leq j \leq N\}$

- The partial order relation ρ over R consists of the following:

  - Priority relation in the membrane labelled by i $(1 \leq i \leq N-1)$: $\rho_i = \emptyset$
  - Priority relation in the skin membrane:

$$
\begin{aligned}
\rho_N = \quad & \left\{r_1 > r'_1 > r_2 > r'_2 > \ldots > r_{k-1} > r'_{k-1}\right\} \cup \left\{r_k > r'_k\right\} \cup \\
& \left\{r_{k+1} > r_{k+2} > r_{k+3} > r'_{k+3}\right\}.
\end{aligned}
$$

3.2 An Overview of Computations

At the beginning of a computation the membrane labelled by i $(1 \leq i \leq N-1)$ contains the objects $a_s,\ b_s,\ e_{js},\ d_{js}$ $(1 \leq s \leq k$ and $i+1 \leq j \leq N)$. In this membrane the presence or absence of the objects a_s, b_s (or e_{js}, d_{js}) encodes the values (0/1) of the individual ω_i (or components of the individuals ω_j).

Initially, the skin membrane contains the objects γ_N, ϵ_0 and η_0. The evolution of the object γ_N allows us to know the number of clusters in all configurations of the

P system: when the object γ_i appears, then the individuals are grouped in i clusters. We use the object ϵ_0 in order to synchronize the loop in $3k-1$ steps, and this allows us to join two clusters with maximum similarity. The object η_0 is a counter used to stop the P system in the configuration $(3k-1)(N-1)$ sending the object $\sharp$ to the environment .

In the initial configuration the only rules that can be applied in membrane i $(1 \leq i \leq N-1)$ are the rules of the type r_{k+4}, r_{k+5} that send the objects $S_{ij}(1 \leq i < j \leq N)$ to the skin membrane. The multiplicity of these objects allows us to know the similarity between individuals of the set Ω. In the skin membrane, the rule r_0 produces the objects ϵ_1, β_0.

After this configuration the computation of the P system is structured into loops with $3k-1$ steps, each one formed by two stages. The first one takes k steps and it begins with the object β_0. In these steps the object S_{ij} with maximum multiplicity, t, is selected. In the k-th step of the loop the rule r_k creates the objects X_{ijt} in the skin membrane and sends a copy to the environment, and the object γ_q is transformed in the object γ_{q-1}, encoding the fact that two clusters have been joined.

The second stage lasts $2k-1$ steps. In the skin membrane there are the objects X_{ijt} meaning that a new cluster labelled by i is constructed by the union of the previous clusters i, j. The rules $r_{k+1}, r_{k+2}, r_{k+3}$ *compute* the similarities between new cluster i and the other clusters, and this information is encoded by the multiplicity of the objects S_{ip} $(1 \leq p \leq N,\ p \neq i)$.

In the $(3k-1)$-th step of the loop, the rule r_{k+3} transforms the object ϵ_{3k-1} in β_0 and ϵ_1 that allow us to go to the top of the loop.

The first partition consist of N singletons, and in each loop two clusters are joined. So, $N-1$ loops are necessary to obtain the last partition (a cluster containing all N individuals). For that, the loop is repeated $N-1$ times and the rule r'_{k-1} is applied in the last step of any computation.

3.3 Formal Verification

In this section we show that the P system $\Pi(P_{Nk})$ is non-deterministic, and any computation we will provide a solution of the clustering problem.

First of all, let us list the necessary resources to construct the P system $\Pi(P_{Nk})$ from the matrix P_{Nk}.

- Size of the alphabet: $\Theta(N^2 \cdot k)$.
- Sum of the sizes of initial multisets: $\Theta(N \cdot k)$.
- Maximum of rules' lengths: $\Theta(N)$.
- Number of rules: $\Theta(k \cdot N^3)$.
- Number of priority relations: $\Theta(k^2 \cdot N^6)$.
- Cost of time: $\Theta(N \cdot k)$.

Bearing in mind the recursive description of the rules and that the amount of resources is polynomial in N, k, it is possible to construct the system $\Pi(P_{Nk})$ from

the matrix P_{Nk} by means of a deterministic Turing machine working in polynomial time.

Given a computation $\mathcal{C}$ of the P system $\Pi(P_{Nk})$, for each $p \in \mathbb{N}$ we denote by $\mathcal{C}_p$ the configuration of the P system obtained after the execution of p steps. For each membrane $l \in \{1, 2, \ldots, N\}$, we denote by $\mathcal{C}_p(l)$ the multiset of objects contained in the membrane labelled by l in $\mathcal{C}_p$.

In what follows, $\mathcal{C}$ will denote an arbitrary computation of the P system $\Pi(P_{Nk})$.

First, we show that in the configuration $\mathcal{C}_1$, the multiplicity of the object S_{ij} $(1 \leq i < j \leq N)$ represents the similarity between the individuals ω_i and ω_j.

Proposition 1. For every i, j, t $(1 \leq i < j \leq N,\ 1 \leq t \leq k-1)$, we denote $t_{ij}^{(1)} = \max\left\{t : S_{ij}^t \in \mathcal{C}_1(N)\right\}$. Then, $t_{ij}^{(1)} = \sum_{s=1}^{k}(1 - \mid \omega_{is} - \omega_{js} \mid)$.

Proof. For every i $(1 \leq i \leq N-1)$ we have:

$$\mathcal{C}_0(i) = \left\{a_s^{(N-i)\omega_{is}}, b_s^{(N-i)(1-\omega_{is})}, e_{js}^{\omega_{js}}, d_{js}^{(1-\omega_{js})} \mid\ i \leq j \leq N, \omega_{is} \in \{0,1\}\right\}.$$

Then, the only rules that can be applied are r_{k+4} and r_{k+5}. The rule r_{k+4} (or r_{k+5}) is only possible to apply when the component s of the individuals ω_i and ω_j is equal to 1 (or to 0).

Whenever one of these rules is applied, the object S_{ij} is sent to the skin membrane. Then, the multiplicity of the objects S_{ij} in $\mathcal{C}_1(N)$ will coincide with the number of equal components between the individuals ω_i and ω_j. That is, $t_{ij}^{(1)} = \sum_{s=1}^{k}(1 - \mid \omega_{is} - \omega_{js} \mid)$. □

From now on, we denote by $t_{ij}^{(n)} = \max\left\{t : S_{ij}^t \in \mathcal{C}_{1+(n-1)(3k-1)}(N)\right\}$ the maximum multiplicity of the objects S_{ij} in the first step of the n-th loop of the computation.

Proposition 2. For each n $(0 \leq n \leq N-2)$, we have:

a) $\beta_0 \in \mathcal{C}_{1+n(3k-1)}(N)$.

b) If $1 \leq j \leq 3k-1$, then $\epsilon_j \in \mathcal{C}_{1+n(3k-1)+(j-1)}(N)$.

Proof. We prove this proposition by induction on n.

- As $\epsilon_0 \in \mathcal{C}_0(N)$ we can apply one of the rules r_0 producing the objects $\epsilon_1, \beta_0 \in \mathcal{C}_1(N)$. In the following $k-1$ steps the rules r_0 will be applied producing the object $\epsilon_k \in \mathcal{C}_k(N)$. If $\alpha_{ijt}, \gamma_q \in \mathcal{C}_k(N)$, then the rule r_k (or the rule r'_k) will be applied. In both cases we obtain that $\epsilon_{k+1} \in \mathcal{C}_{k+1}(N)$. In successive configurations the rule r_0 transforms the objects $\epsilon_j \in \mathcal{C}_j(N)$, $(k+1 \leq j \leq 3k-2)$ until we obtain the object $\epsilon_{3k-1} \in \mathcal{C}_{3k-1}(N)$.
- Let us suppose the hypothesis holds for $0 \leq n < N-2$. Then, $\epsilon_{3k-1} \in \mathcal{C}_{1+n(3k-1)+(3k-1-1)}(N) = \mathcal{C}_{(n+1)(3k-1)}(N)$. If there is some object X_{ijt} in that configuration, then the rules from r_{k+3} will be applied, and in the opposite case,

the rule r'_{k+3} will be applied. In both cases the object ϵ_{3k-1} is transformed in ϵ_1, $\beta_0 \in \mathcal{C}_{1+(n+1)(3k-1)}(N)$. Applying $k-1$ times the rules from r_0 we obtain that $\epsilon_j \in \mathcal{C}_{1+(n+1)(3k-1)+(j-1)}(N)$ $(1 \leq j \leq k)$. In $\mathcal{C}_{1+(n+1)(3k-1)+(k-1)}(N)$ the object ϵ_k produces $\epsilon_{k+1} \in \mathcal{C}_{1+(n+1)(3k-1)+k}(N)$ by means of one rule from r_k or r'_k. Then, applying the rules from r_0 successively we obtain that $\epsilon_j \in \mathcal{C}_{1+(n+1)(3k-1)+(j-1)}(N)$ $(k+1 \leq j \leq 3k-1)$. □

Corollary 1. The objects X_{ijt} only can be sent to the environment at the moments $1+n(3k-1)+k$ $(0 \leq n \leq N-2)$.

Proof. The rule r_k is the only one that sends objects X_{ijt} to the environment, and from Proposition 2 we have $\epsilon_k \in \mathcal{C}_{1+n(3k-1)+k-1}(N)$ $(0 \leq n \leq N-2)$. □

Corollary 2. There exists n $(0 \leq n \leq N-2)$ and there are objects X_{ijt} such that $X_{ijt} \in \mathcal{C}_{1+n(3k-1)+k}(N)$.

Proposition 3. The configuration $\mathcal{C}_{(N-1)(3k-1)}$ sends the halting object ♯ to the environment.

Proof. Applying $(N-1)(3k-1)$ times the rules from r_0, the object $\eta_0 \in \mathcal{C}_0(N)$ produces $\eta_{(N-1)(3k-1)} \in \mathcal{C}_{(N-1)(3k-1)}(N)$. In this configuration the rule r'_{k-1} sends the halting object ♯ to the environment. □

Next, we show that it is only possible to modify the environment in the k^{th} step of the loop.

Corollary 3. For every n $(0 \leq n \leq N-2)$ the following assertions hold:

a) For every r, $(1+n(3k-1) < r < 1+n(3k-1)+k)$ we have:

$$\mathcal{C}_r(env) = \mathcal{C}_{1+n(3k-1)}(env).$$

b) For every r, $(1+n(3k-1)+k < r < 1+n(3k-1)+3k-1)$ we have:

$$\mathcal{C}_r(env) = \mathcal{C}_{1+n(3k-1)+k}(env).$$

Proof. The rule r_k is the only one sending objects to the environment before the halting configuration. From Corollary 1 we have $X_{ijt} \in \mathcal{C}_{1+n(3k-1)+k}(env)$. Thus, for every r $(1+n(3k-1) < r < 1+n(3k-1)+k)$ we have $\mathcal{C}_r(env) = \mathcal{C}_{1+n(3k-1)}(env)$, and for every r $(1+n(3k-1)+k < r < 1+n(3k-1)+3k-1$ we have $\mathcal{C}_r(env) = \mathcal{C}_{1+n(3k-1)+k}(env)$. □

In the following, we will show that in each loop one object X_{ijt} is sent to the environment and eventually there exists a loop which sends no object X_{ijt} to the environment. In this case, in next loops no further objects are sent to the environment.

Firstly we prove that if in the k-th step of the loop the rule r_k is not applicable, then also it is not applicable in the next loop.

Proposition 4. For every n $(0 \leq n \leq N-2)$ if the rule r_k cannot be applied in $\mathcal{C}_{1+n(3k-1)+k-1}$, then it cannot be applied in $\mathcal{C}_{1+(n+1)(3k-1)+k-1}$.

Proof. According to Proposition 2, for each n $(0 \leq n \leq N-2)$ we have $\epsilon_k \in \mathcal{C}_{1+n(3k-1)+k-1}(N)$. The objects γ_q always appear in the skin membrane. As the rule r_k is not applicable in $\mathcal{C}_{1+n(3k-1)+k-1}$, we have $\alpha_{ijt} \notin \mathcal{C}_{1+n(3k-1)+k-1}(N)$.

Having in mind that objects β_u appear in $\mathcal{C}_{1+n(3k-1)+u}(N)$ $(1 \leq u \leq k-1)$ we deduce that $S_{ij} \notin \mathcal{C}_{1+n(3k-1)+u}(N)$. Hence the rule r_k cannot be applied to the configuration $\mathcal{C}_{1+n(3k-1)+k}$.

From step $1+n(3k-1)+k-1$ to step $1+(n+1)(3k-1)+k-1$, any object S_{ij} is produced. Then, the rule r_k is not applicable to $\mathcal{C}_{1+(n+1)(3k-1)+k-1}$. □

Corollary 4. Let n be such that $0 \leq n \leq N-2$. If $\mathcal{C}_{1+n(3k-1)+k}(env)$ is equal to $\mathcal{C}_{1+(n+1)(3k-1)+k}(env)$, then for each n' $(n \leq n' \leq N-2)$ we have

$$\mathcal{C}_{1+n(3k-1)+k}(env) = \mathcal{C}_{1+n'(3k-1)+k}(env).$$

Corollary 5. For each computation $\mathcal{C}$ there exists an unique $\nu_{\mathcal{C}}$ $(1 \leq \nu_{\mathcal{C}} \leq N-2)$ such that the rule r_k is applicable to $\mathcal{C}_{1+(\nu_{\mathcal{C}}-1)(3k-1)+k-1}$, but it is not applicable to $\mathcal{C}_{1+\nu_{\mathcal{C}}(3k-1)+k-1}$.

Proof. It follows from Corollary 2, Proposition 4, and Corollary 4. □

The following result allows us to give a meaning to the value t associated with the object X_{ijt}.

Proposition 5. Let $X_{i_nj_nt^{(n)}_{i_nj_n}}$ be the object that is sent to the environment by applying the rule r_k to the configuration $\mathcal{C}_{(k+1)+n(3k-1)-1}$. Then, we have

$$t^{(n)}_{i_nj_n} = \max\left\{t \mid \ S^t_{ij} \in \mathcal{C}_{1+n(3k-1)}(N),\ 1 \leq i < j \leq N\right\}.$$

Proof. As the rule r_k is applicable to $\mathcal{C}_{(k+1)+n(3k-1)-1}$, we have the object $\alpha_{i_nj_nt^{(n)}_{i_nj_n}}$ belongs to $\mathcal{C}_{(k+1)+n(3k-1)-1}$. That object is obtained from the application of one of the rules $r_{k-t^{(n)}_{i_nj_n}}$ over the object $S^{t^{(n)}_{i_nj_n}}_{ij}$, where $t^{(n)}_{i_nj_n}$ is the maximum of the multiplicities of the objects S_{ij}. If there exists $t' > t^{(n)}_{i_nj_n}$ such that $S^{t'}_{ij} \in \mathcal{C}_{1+n(3k-1)}(N)$, then rules from $r_{k-t'}$ have been applied. So, rules from $r_{k-t^{(n)}_{i_nj_n}}$ will not be applicable. □

Next, we show that the maximum multiplicity of the objects S_{ij} belonging to the skin membrane in any loop n is always greater than or equal to the multiplicity of the objects S_{ij} of the following loop $n+1$.

Proposition 6. Let $w_n = \max\left\{t : S^t_{ij} \in \mathcal{C}_{1+n(3k-1)}(N), 1 \leq i < j \leq N\right\}$, with $0 \leq n \leq N-2$. Then $w_n \geq w_{n+1}$, for each n.

Proof. From step $1+n(3k-1)+1$ to step $1+n(3k-1)+w_n$, the rules $r_0, r'_1, \ldots, r'_{w_n-1}$, r_{w_n} are applied in the skin membrane. So, the objects S_{ij} do not evolve, and $\epsilon_k \in \mathcal{C}_{1+n(3k-1)+w_n}(N)$. From Proposition 5 we deduce that

$$w_n \geq \max\left\{t : S_{ij}^t \in \mathcal{C}_{1+n(3k-1)+w_n}(N)\right\}.$$

- If $w_n \neq 0$, then the multiplicity of the objects S_{ij} only can decrease by applying the rules r_{k+1}, r_{k+2} and r_{k+3}. After that, the rule r_0 is applied until reaching the configuration $\mathcal{C}_{1+(n+1)(3k-1)}$. Hence, $w_{n+1} = \max\left\{t : S_{ij}^t \in \mathcal{C}_{1+(n+1)(3k-1)}(N)\right\} \leq w_n$.
- If $w_n = 0$, the objects S_{ij} do not belong to the skin membrane and by Proposition 4 it is not possible to produce any object S_{ij}.

  Hence, $w_{n+1} = \max\left\{t : S_{ij}^t \in \mathcal{C}_{1+(n+1)(3k-1)}(N)\right\} = 0$. □

Remark 2. According to Proposition 6 we obtain $t_1 \geq t_2 \geq \ldots \geq t_n$.

Next, we show that if a loop sends an object X_{ijt} to the environment, then in the next loop the objects $S_{ij}, S_{i'j}, S_{ji'}$ $(i' \notin \{i,j\})$ disappear from the skin membrane. That is, at the moment that two clusters $\{i,j\}$ are joined a new class i is formed and all the objects $S_{i'j'}$ that have subscript j disappear.

Proposition 7. Let $X_{i_1j_1t_{i_1j_1}^{(1)}}, X_{i_2j_2t_{i_2j_2}^{(2)}}, \ldots, X_{i_nj_nt_{i_nj_n}^{(n)}} \in \mathcal{C}_{1+n(3k-1)}(env)$, with $1 \leq n \leq \nu_{\mathcal{C}}$. If $S_{ij} \in \mathcal{C}_{1+n(3k-1)}(N)$, then $(i,j) \notin \{(i_1,j_1), \ldots, (i_n,j_n)\}$, and $\{i,j\} \in \{1,\ldots,N\} - \{j_1,\ldots,j_n\}$.

Proof. We prove the result by induction on n.

- Let us suppose that the rule r_k sends the object $X_{i_1j_1t_{i_1j_1}^{(1)}}$ to the environment in step k. From Proposition 6 we deduce that if $S_{ij} \in \mathcal{C}_{1+(3k-1)}(N)$, then $(i,j) \notin \{(i_1,j_1)\}$. In the next steps, applying the rules r_{k+1}, r_{k+2} and r_{k+3} we obtain that the objects $S_{ij} \in \mathcal{C}_{1+(3k-1)}$ verify $(i,j) \notin \{(i_1j_1)\}$, $\{i,j\} \in \{1,\ldots,N\} - \{j_1\}$.
- Let us suppose the proposition holds for $1 \leq n < \nu_{\mathcal{C}}$. From Proposition 6, if $X_{i_{n+1}j_{n+1}t_{i_{n+1}j_{n+1}}^{(n+1)}} \in \mathcal{C}_{k+n(3k-1)}(env)$ and $S_{ij} \in \mathcal{C}_{1+(n+1)(3k-1)}(N)$, then $(i,j) \notin \{(i_{n+1},j_{n+1})\}$. So, we have $(i,j) \notin \{(i_1,j_1),\ldots,(i_n,j_n)\}$ by the induction hypothesis. Then, $(i,j) \notin \{(i_1,j_1),\ldots,(i_{n+1},j_{n+1})\}$. In the next steps, applying the rules r_{k+1}, r_{k+2} and r_{k+3} we obtain that the objects $S_{ij} \in \mathcal{C}_{1+(n+1)(3k-1)}(N)$ verify $(i,j) \notin \{(i_{n+1}j_{n+1})\}$, $\{i,j\} \in \{1,\ldots,N\} - \{j_{n+1}\}$. The proof concludes by using the induction hypothesis. □

In the following proposition we study how the multiplicities of the objects S_{ij} change when two clusters are joined.

Proposition 8. Let us suppose that $X_{i_1j_1t_{i_1j_1}^{(1)}}, \ldots, X_{i_nj_nt_{i_nj_n}^{(n)}} \in \mathcal{C}_{1+n(3k-1)}(env)$ with $1 \leq n \leq \nu_{\mathcal{C}}$, and $t_{ij}^{(n)} = max\left\{t : S_{ij}^t \in \mathcal{C}_{1+n(3k-1)}(N)\right\}$. Then,

- If $i_n \notin \{i, j\}$, then $t_{ij}^{(n+1)} = t_{ij}^{(n)}$.
- If $1 \leq i_n < j_n < p \leq N$, then $t_{i_n p}^{(n+1)} = \min\left\{t_{i_n p}^{(n)}, t_{j_n p}^{(n)}\right\}$.
- If $1 \leq i_n < p < j_n \leq N$, then $t_{i_n p}^{(n+1)} = \min\left\{t_{i_n p}^{(n)}, t_{p j_n}^{(n)}\right\}$.
- If $1 \leq p < i_n < j_n \leq N$, then $t_{p i_n}^{(n+1)} = \min\left\{t_{p i_n}^{(n)}, t_{p j_n}^{(n)}\right\}$.

Proof. The pairs $(S_{i_n p}, S_{j_n p}), (S_{i_n p}, S_{p j_n}), (S_{p i_n}, S_{p j_n})$ are transformed in the objects $C_{i_n p}, C_{i_n p}, C_{p i_n}$, respectively, by applying the rule r_{k+1} to $\mathcal{C}_{k+(n-1)(3k-1)}$. After that, the rule r_{k+2} removes the objects $S_{i_n p}, S_{j_n p}, S_{p i_n}, S_{p j_n}$. Thus, the multiplicity of the objects $C_{i_n p}, C_{i_n p}, C_{p i_n}$ is equal to $\min\left\{t_{i_n p}^{(n)}, t_{j_n p}^{(n)}\right\}, \min\left\{t_{i_n p}^{(n)}, t_{p j_n}^{(n)}\right\}$, and $\min\left\{t_{p i_n}^{(n)}, t_{p j_n}^{(n)}\right\}$, respectively. Finally, we note that the objects S_{ij} are produced from the objects C_{ij} by applying the rule r_{k+3}. □

Let us consider a $\mathcal{C}$ a computation of the P system $\Pi(P_{Nk})$. We recursively construct a sequence of partitions $\Delta_0^{\mathcal{C}},\ \Delta_1^{\mathcal{C}},\ \ldots, \Delta_{\nu_{\mathcal{C}}}^{\mathcal{C}}$ of the set of the individuals, as follows:

- $\Delta_0^{\mathcal{C}} = \left\{B_{q_0^1}^0,\ \ldots,\ B_{q_0^N}^0\right\}$ with $q_0^i = i$ and $B_i^0 = \{\omega_i\} \equiv \{i\}$ $(1 \leq i \leq N)$.
- The partition $\Delta_1^{\mathcal{C}}$ is constructed from the object $X_{i_1 j_1 t_{i_1 j_1}^{(1)}} \in \mathcal{C}_{k+1}(env)$ with $1 \leq i_1 < j_1 \leq N$, as follows:
  - We have $\{i_1, j_1\} \subseteq \left\{q_0^1, \ldots, q_0^N\right\}$. If $i_1 = q_0^u$ and $j_1 = q_0^s$, with $1 \leq u < s \leq N$, then the new cluster is $B_{q_1^u}^1 = B_{q_0^u}^0 \cup B_{q_0^s}^0$ with $q_1^u = q_0^u$, $B_{q_0^s}^0 \notin \Delta_1^{\mathcal{C}}$, and $B_l^1 = B_l^0$ for $l \in \left\{q_0^1, \ldots, q_0^N\right\} - \{q_0^u, q_0^s\}$.

  Then $\Delta_1^{\mathcal{C}} = \left\{B_{q_1^1}^1, \ldots, B_{q_1^{N-1}}^1\right\}$.
- We construct $\Delta_{n+1}^{\mathcal{C}}$ from $\Delta_n^{\mathcal{C}} = \left\{B_{q_n^1}^n, \ldots, B_{q_n^{N-n}}^n\right\}$ and $X_{i_{n+1} j_{n+1} t_{i_{n+1} j_{n+1}}^{(n+1)}}$, as follows:
  - From Proposition 7, we deduce that $\{i_{n+1}, j_{n+1}\} \subseteq \left\{q_n^1, \ldots, q_n^{N-n}\right\}$. If $i_{n+1} = q_n^u$ and $j_{n+1} = q_n^s$, with $1 \leq u < s \leq N$, then the new cluster is $B_{q_{n+1}^u}^{n+1} = B_{q_n^u}^n \cup B_{q_n^s}^n$ with $q_{n+1}^u = q_n^u$, $B_{q_n^s}^n \notin \Delta_{n+1}^{\mathcal{C}}$, and $B_l^{n+1} = B_l^n$ for $l \in \left\{q_n^1, \ldots, q_n^{N-n}\right\} - \{q_n^u, q_n^s\}$.

  Then $\Delta_{n+1}^{\mathcal{C}} = \left\{B_{q_{n+1}^1}^{n+1}, \ldots, B_{q_{n+1}^{N-n-1}}^{n+1}\right\}$.

  Next, we show that $t_{ij}^{(n)}$ is the aggregation index between clusters.

Theorem 1. For every i, j, n $(1 \leq i < j \leq N,\ 1 \leq n \leq \nu_{\mathcal{C}})$, $t_{ij}^{(n)}$ is the minimum similarity between any pair of individuals belonging to $B_i^{n-1} \cup B_j^{n-1}$. That is,

$$t_{ij}^{(n)} = \min\left\{t_{i'j'}^{(1)} :\ i', j' \in B_i^{n-1} \cup B_j^{n-1}\right\} = \delta(B_i^{n-1}, B_j^{n-1}).$$

Proof. By induction on n.

- From Proposition 1, if $S_{ij}^{t_{ij}^{(1)}} \in \mathcal{C}_1(N)$, then $t_{ij}^{(1)}$ corresponds to the similarity between individuals i, j.
- Let us suppose that the result holds for n with $1 \leq n < \nu_{\mathcal{C}}$. Let $X_{i_n j_n t_{i_n j_n}^{(n)}}$ $(1 \leq i_n < j_n \leq N)$ be the object sent to the environment in the configuration $\mathcal{C}_{1+n(3k-1)+k}$, and let $\Delta_{n-1}^{\mathcal{C}} = \left\{B_{q_{n-1}^1}^{n-1}, \ldots, B_{q_{n-1}^{N-n+1}}^{n-1}\right\}$.
  - If $i_n \notin \{i, j\}$, then from Proposition 8 we have $t_{ij}^{(n+1)} = t_{ij}^{(n)}$. The result follows from the construction of $\Delta_n^{\mathcal{C}}$ and the induction hypothesis.
  - If $1 \leq i < i_n < j_n$, then from Proposition 8 we deduce that $t_{ii_n}^{(n+1)} = \min\left\{t_{ii_n}^{(n)}, t_{ij_n}^{(n)}\right\}$. The result follows from the construction of $\Delta_n^{\mathcal{C}}$ and the induction hypothesis.
  - The proof is similar in the remaining cases. □

Proposition 9. For each n $(0 \leq n \leq \nu_{\mathcal{C}} - 1)$, let us suppose that the partition $\Delta_n^{\mathcal{C}} = \left\{B_{q_n^1}^n, \ldots, B_{q_n^{N-n}}^n\right\}$ is constructed from the object $X_{i_n j_n t_{i_n j_n}^{(n)}}$. If we denote by f the hierarchical index function, then $f(B_{i_n}^n) = t_{i_n j_n}^{(n)}$, and for each $B \in \Delta_n^{\mathcal{C}} - \left\{B_{i_n}^n\right\}$, we have $f(B) \geq t_{i_n j_n}^{(n)}$.

Proof. By induction on n.

- Let us recall that $\Delta_0^{\mathcal{C}} = \{\{\omega_1\}, \ldots, \{\omega_N\}\}$ and $f(\{\omega_i\}) = k$ $(1 \leq i \leq N)$.
- From construction, $\Delta_1^{\mathcal{C}} = \left\{B_{q_1^1}^1,\ \ldots,\ B_{q_1^{N-1}}^1\right\}$ with $B_{i_1}^1 = \{\omega_{i_1}, \omega_{j_1}\}$ and $B_j^1 = \{\omega_j\}$ $(\forall j \neq i_1)$. Then $f(B_j^1) = k$ with $j \neq i_1$ and $f(B_{i_1}^1) = t_{i_1 j_1}^{(1)}$ (because $s(\omega_{i_1}, \omega_{j_1}) = t_{i_1 j_1}^{(1)} \leq k - 1$).
- Let us suppose that the result holds for n $(1 \leq n < \nu_{\mathcal{C}})$. We have $t_{i_{n+1} j_{n+1}}^{(n+1)} \leq t_{i_n j_n}^{(n)}$. Let $\Delta_{n+1}^{\mathcal{C}} = \left\{B_{q_{n+1}^1}^{n+1}, \ldots, B_{q_{n+1}^{N-n-1}}^{n+1}\right\}$.
  - If $i_{n+1}, j_{n+1} \in B_{q_n^l}^n$, then $B_{q_{n+1}^l}^{n+1} = B_{q_n^l}^n$. From the induction hypothesis we deduce that $f(B_{q_{n+1}^l}^{n+1}) = t_{i_n j_n}^{(n)} \geq t_{i_{n+1} j_{n+1}}^{(n+1)}$.
  - If $B_{i_{n+1}}^{n+1} = B_{i_{n+1}}^n \cup B_{j_{n+1}}^n$ and $\delta(B_{i_{n+1}}^n,\ B_{j_{n+1}}^n) = t_{i_{n+1} j_{n+1}}^{(n+1)}$, then $f(B_{i_{n+1}}^{n+1}) = t_{i_{n+1} j_{n+1}}^{(n+1)}$. □

Proposition 10. The P system $\Pi(P_{Nk})$ allows us to construct a hierarchical clustering associated with any computation of the P system.

Proof. Let $\mathcal{C}$ be a computation of the P system $\Pi(P_{Nk})$. Let $\Delta_0^{\mathcal{C}},\ \Delta_1^{\mathcal{C}},\ \ldots, \Delta_{\nu_{\mathcal{C}}}^{\mathcal{C}}$ be the partition associated with it. By Proposition 9 all the clusters of the partition $\Delta_n^{\mathcal{C}}$ have a hierarchical index greater than or equal to $t_{in_{Jn}}^{(n)}$ (denoted by t_n in advance).

We construct the partition of the hierarchy $P_0, P_1, \ldots, P_m$ as follows:

- $P_0 = \Delta_0 = \{\{\omega_1\}, \{\omega_2\}, \ldots, \{\omega_N\}\}$.
- If the partitions $\Delta_1, \Delta_2, \ldots, \Delta_{p_1}$ have associated the same hierarchical index, then $P_1 = \Delta_{p_1}$.
- If the partitions $\Delta_{p_1+1}, \Delta_{p_1+2}, \ldots, \Delta_{p_2}$ have associated the same hierarchical index, then $P_2 = \Delta_{p_2}$.
- We continue in this way until we have one of the following situations:
  - if $\Delta_{\nu_{\mathcal{C}}}$ has a hierarchical index $t_{\nu_{\mathcal{C}}} = k - 1$, then $P_m = \Delta_{\nu_{\mathcal{C}}} = \Omega$.
  - if $\Delta_{\nu_{\mathcal{C}}}$ has a hierarchical index $t_{\nu_{\mathcal{C}}} < k - 1$, then $P_{m-1} = \Delta_{\nu_{\mathcal{C}}}$ and $P_m = \Omega$.

□

4 CONCLUSIONS

One of the central issues when we have a set of individuals, each of them characterized by a k-tuple, is to obtain a cluster that allows us to group similar individuals.

In this paper we propose a non-deterministic P system with external output to obtain a hierarchical clustering. This P system gives one of the possible solutions to the problem. We present an efficient (semi-uniform) solution to the problem of clustering in the framework of the cellular computing with membranes. The solution is semi-uniform because for each matrix formed by the values of the individuals, a specific P system with external output is designed. The solution is efficient, because it is polynomial in order of the number N of individuals and of the number k of characteristics. The amount of resources initially required to construct the system is polynomial in N and k.

The mechanisms of the formal verification of P systems are often a very hard task. Therefore, to have new examples where this task is accomplished is always interesting, in order to find systematic processes of formal verification in a model of computation oriented to machines, like the P systems. The paper provides such a new example of formal verification of P systems designed to solve a problem, following a specific methodology valid in cases as that considered in the paper.

Acknowledgement

The authors acknowledge the support of the project TIN2006-13425 of the Ministerio de Educación y Ciencia of Spain, cofinanced by FEDER funds, and the support of the project of excellence TIC-581 of the Junta de Andalucía.

REFERENCES

[1] Acoub, M.—Badran, F.—Thiria, S.: A Topological Hierarchical Clustering: Application to Ocean Color Classification. Lecture Notes in Computer Science, Vol. 2130, 2001, pp. 492–499.

[2] Diday, E.—Lemaire, J.—Pouget, J.—Testu, F.: Éléments d'Analyse de Données. Dunod, Paris, 1982.

[3] Duda, R. O.—Hart, P. E.: Pattern Classification and Scene Analysis. John Wiley, New York, 1973.

[4] Everitt, B.: Cluster Analysis. John Wiley & Sons. London, 1980.

[5] Johnson, S. C.: Hierarchical Clustering Schemes. Psychometrika, Vol. 32, 1967, pp. 241–254.

[6] Păun, Gh.: Computing with Membranes. Journal of Computer and System Sciences, Vol. 61, No. 1, pp. 108–143, and Turku Center for Computer Science-TUCS Report, No. 208, 1998.

[7] Păun Gh.: Membrane Computing. An Introduction. Springer-Verlag, Berlin, 2002.

[8] Păun, Gh.—Rozenberg, G.: A Guide to Membrane Computing. Theoretical Computer Science, Vol. 287, 2002, pp. 73–100.

[9] Razmi, J.—Zairi, M.—Gunasekaran, A.: The Application of Analytic Hierarchy Process in Classification of Material Planning and Control Systems. International Journal of Services and Operations Management, Vol. 2, 2006, No. 4, pp. 352–366.

[10] Sokal, R. R.—Sneath, P. H. A.: Principles of Numerical Taxonomy. Freeman & Co., London, 1963.

[11] Späth, H.—Bull, U.: Cluster Analysis Algorithms for Data Reduction and Classification of Objects. John Wiley & Sons, New York, 1980.

[12] Ture, M.—Kurt, I.—Kurum, A. T.—Ozdamar, K.: Comparing Classification Technique for Predicting Essential Hypertension. Expert Systems with Applications, Vol. 29, 2005, pp. 583–588.

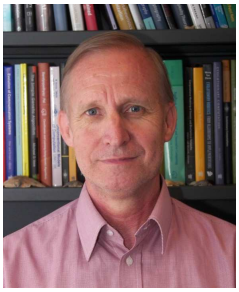
[13] Tzanetakis, G.—Cook, P.: Automatic Musical Genre Classification of Audio Signals. IEEE Transactions on Speech and Audio Processing, Vol. 10, July 2002, No. 5.

[14] Zhang, X. H.—Chen, J. M.: Application of Hierarchical Cluster with High-Performance Liquid Chromatography in the Classification of Genus Clinopidium Chinesis. Zhongguo Zhong Yao Za Zhi, Vol. 28, 2003, No. 9, pp. 812–816.

[15] ISI Web Page: `http://esi-topics.com/erf/october2003.html`.

Mónica Cardona received her degree in mathematics from the Barcelona University. She is associated professor of applied mathematics at the University of Lleida. Her main research areas are in stochastic process and membrane computing. She has published 3 scientific papers.

M. Angels Colomer received her doctor degree in mathematical sciences by the Polytechnic University of Catalonia. In the past,she was associated professor at the Polytechnic University of Catalonia. She is titular Professor of statistic and operative research at the University of Lleida. Her main research areas are in stochastic process and membrane computing. She has published seven books in statistics and control quality, two book chapters of applications of stochastic models, and 15 scientific papers.



Mario J. Pérez-Jiménez received his degree in mathematics from the Barcelona University and doctor degree in mathematics from the Sevilla University. In the past, he was associated professor at the University of Barcelona. Currently, he is titular professor of computer science and artificial intelligence at the University of Sevilla, where is the head of the Research Group on Natural Computing. His main research areas are computational complexity theory, unconventional models of computation, natural computing, membrane computing, bioinformatics, and computational modelling for systems biology. He has published twelve books in computer science and mathematics, and over 100 scientific papers in international journals (collaborating with many researchers worldwide). He has served as reviewer for various prestigious ISI–journals (e.g. BioSystems, Discrete Applied Mathematics, Informatica Lithuan, International Journal of Foundations of Computer Science, International Journal of Computer Mathematics, Neurocomputing, Pattern Recognition Letters, SoftComputing, Theoretical Computer Science, Theory of Computing Systems). He is a member of editorial boards of the journals Soft Computing, and International Journal of Computers, Communications and Control. He is the main researcher in various European, Spanish and Andalusian research grants. He has been and independent expert to the evaluation of NEST (New and Emergent Science and Technology) proposals under the Sixth Framework Programme of the European Community, and he is an European Science Foundation peer reviewer.

Alba Zaragoza received her degree in mathematics from the Autonomous University of Barcelona and doctor degree from Lleida University. She is an associate professor of applied mathematics at the University of Lleida. Her main research areas are in stochastic processes and membrane computing. She has published a book on control quality and 4 scientific papers.