Hierarchical Clustering with Membrane Computing

Mónica Cardona[1], M. Angels Colomer[1],
Mario J. Pérez-Jiménez[2], Alba Zaragoza[1]

[1] Department of Mathematics, University of Lleida
Av. Alcalde Rovira Roure, 191. 25198 LLeida, Spain
{colomer,alba,mcardona}@matematica.udl.es

[2] Research Group on Natural Computing
Department of Computer Science and Artificial Intelligence
University of Sevilla
Avda. Reina Mercedes s/n, 41012 Sevilla, Spain
marper@us.es

Summary. In this paper we approach the problem of the 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 constructions is polynomial in the number of individuals and the number of characteristics analyzed.

1 Introduction

Researchers develop a lot of investigation that depend on many factors and this makes their study very complex. In order to simplify and make the problems more tractable it is necessary to group individuals with similar characteristics. The individuals are characterized by a high number of properties so the grouping is not a simple task. The clustering methods appear with the purpose of establishing a methodology with a statistical base in order to obtain the groupings of the individuals according to their degree of similarity.

There are different methods of ranking the groups of individuals. In order to simplify it we can consider two types, the nonhierarchical clustering and the hierarchical clustering. In a nonhierarchical clustering homogenous groups are formed without establishing relations among them; in the hierarchical clustering the individuals are grouped in levels. The inferior levels are contained in the superior levels. The hierarchical clustering is the most 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 the two most similar groups are joined according to a previously established criterion.

Researchers use the clustering to characterize and to order a vast amount of information on the 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, of the fingerprints, artificial intelligence... Recently the 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], in the classification of the ocean color [1], in the classification 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. To have established groups allows us to identify the most similar cluster of a new individual. The characteristics measured of the individuals can be qualitative variables or quantitative variables. In most cases we are only interested in the presence or absence of certain qualitative characteristics. So in this paper we make a hierarchical clustering using dichotomizing variables by means of membrane computing.

In this paper the problem of hierarchical clustering is approached with the framework of cellular computing with membranes. It is interesting because allows us treated some statistics topics with this new models of computation. The amount of used resources is polynomial in the number of individuals and the number of characterizes analyzed without increasing the complexity of the classical clustering algorithms.

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

2 Overview

2.1 Hierarchical Clustering

In order to obtain a hierarchical clustering we need a set of observations or individuals that we define as follows:

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

The hierarchical clustering needs a finite k-set Ω with N elements, $\Omega = \{\omega_1, \ldots, \omega_N\}$. The elements of the set Ω are called individuals or observations and their components in the k-tuple are denoted characteristics or variables. The values of the individuals can be represented in matrix form:

$$P_{Nk} = \begin{pmatrix} \omega_{11} & \omega_{12} & \cdots & \omega_{1k} \\ \omega_{21} & \omega_{22} & \cdots & \omega_{2k} \\ & & \cdots & \\ \omega_{N1} & \omega_{N2} & \cdots & \omega_{Nk} \end{pmatrix}$$

where ω_{ij} is the value of the j-th variable of the individual i.

The objective of any clustering is to group the individuals in similar groups whose members are all close to one another with various dimensions being measured. It will be necessary to establish criteria in order to measure the similarity between individuals and similarity between groups. Evidently, the clustering that is obtained will depend on the similarity function that is chosen. This function is called similarity and it is defined as follows [10].

Definition 2. *A similarity over a finite k–set $\Omega = \{\omega_1, \dots, \omega_N\}$ 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, in particular their values are denoted by 0 and 1. One of the similarities most used for dichotomizing variables is the similarity of Sokal and Michener [2] and it is defined by:

$$\forall \omega_i, \omega_j \in \Omega : \quad s'(\omega_i, \omega_j) = \frac{1}{k} \cdot \sum_{r=1}^{k} (1 - |\omega_{ir} - \omega_{jr}|) \tag{1}$$

where $\omega_i = \{\omega_{i1}, \dots, \omega_{ik}\}$.

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

$$\forall(\omega_i, \omega_j) \in \Omega \times \Omega : \quad s(\omega_i, \omega_j) = \sum_{r=1}^{k} (1 - |\omega_{ir} - \omega_{jr}|) \tag{2}$$

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

In the case of the hierarchical clustering the groupings follow a hierarchy formed by partitions. The partitions are formed in a recursive manner. We start with as many clusters as individuals, in each iteration the partition is obtained joining the two closest clusters. This process is done until we obtain a single set formed by all the individuals. The partitions obtained $P_0, P_1, \dots, P_m$ verify $P_0 \subseteq P_1 \subseteq P_2 \subseteq \dots \subseteq P_m$ with $1 \leq m \leq N - 1$ and the sets that form the partitions are called clusters.

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 to classify. A subset* H *of the parts of* Ω*,* $H \subseteq \mathcal{P}(\Omega)$*, is a hierarchy over* Ω *if it verifies:*

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

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*
$\delta : \mathcal{P}(\Omega) \times \mathcal{P}(\Omega) \to \mathbf{R}$ *is called aggregation index between clusters if it verifies:*

- $\forall h_1, h_2 \in \mathcal{P}(\Omega): \quad \delta(h_1, h_2) \geq 0$
- $\forall h_1, h_2 \in \mathcal{P}(\Omega): \quad \delta(h_1, h_2) = \delta(h_2, h_1)$

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:

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

If a hierarchy has associated an index that measures the homogeneity degree between the individuals belonging to the same cluster it is called indexed hierarchy. We refer to this index by the hierarchical index.

Definition 5. *An indexed hierarchy is a pair* (H, f) *where* H *is a hierarchy and* f *is an application* H *over* $\mathbf{R}^+$ *such that:*

- $f(\{\omega\}) = k \ (\forall \omega \in \Omega)$
- $\forall h' \in H: \quad h \subsetneq h' \Rightarrow f(h) > f(h')$

The hierarchical 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 $f(h) = \delta(h_1, h_2)$.

An algorithm for the construction of an indexed hierarchy

The algorithms that are used to obtain an indexed hierarchy have the same structure, the only differences in them is the way to compute the similarities between clusters [3].

The input of this algorithm is the k-set Ω 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 $L = P_0$

$$L = P_0 = \{S_1 = \{\omega_1\}, S_2 = \{\omega_2\}, \dots, S_N = \{\omega_N\}\}$$

In this moment $\delta(\{\omega_i\}, \{\omega_j\}) = s(\omega_i, \omega_j)$ and $f(\{\omega_i\}) = k$ $(1 \le i < j \le N)$

2 Find the two closest clusters S_i, S_j with $1 \le i < j \le N$, which will form a new class $S_i = S_i \cup S_j$.

3 Remove S_j from L.

4 Compute the aggregation index, by equation (3), between all the pair of clusters in L.

5 Go to step 2 until there is only one set remaining.

Remark: 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 of a Group of Individuals

3.1 Designing a P System

The goal of this paper is to obtain one hierarchical clustering of a k-set Ω, of N different individuals by means of the cellular computing with membranes. We considered each individual $\omega_i \in \Omega$ by a k-tuple of dichotomizing variables, $\Omega \subseteq \{0, 1\}^k$ which is denoted by $\omega_i = (\omega_{i1}, \omega_{i2}, \dots, \omega_{ik})$. The similarity between individuals that we use is the following:

$$s(\omega_i, \omega_j) = \sum_{t=1}^{k} (1 - |\omega_{it} - \omega_{jt}|)$$

This similarity measures the number of equal components between two individuals.

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

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

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

- Working alphabet:

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

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

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

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

- The set R of evolution rules consists of the following rules:
  - Rules in the skin membrane labeled N:

$$\begin{aligned}r_0 = \quad &\{\epsilon_0 \to \epsilon_1\beta_0\} \cup \{\epsilon_i \to \epsilon_{i+1} : 1 \le i \le 3k-2 \wedge i \ne k\} \cup \\ &\{\eta_i \to \eta_{i+1} : 0 \le i \le (N-1)(3k-1)-1\}\end{aligned}$$

$$r_u = \quad \{\beta_{u-1}S_{ij}^{k-u} \to \alpha_{ij(k-u)} :\ 1 \le i < j \le N\} \quad 1 \le u \le k-1$$

$$r'_u = \quad \{\beta_{u-1} \to \beta_u\} \quad 1 \le u \le k-1$$

$$r'_{k-1} = \{\eta_{(N-1)(3k-1)} \to (\sharp, out)\}$$

$$\begin{aligned}r_k = \quad &\{\epsilon_k\gamma_q\alpha_{ijt} \to \epsilon_{k+1}X_{ijt}^{q-2}\gamma_{q-1}(X_{ijt}, out) :\ 2 \le q \le N, \\ &1 \le i < j \le N,\ 1 \le t \le k-1\}\end{aligned}$$

$$r'_k = \quad \{\epsilon_k \to \epsilon_{k+1}\}$$

$$\begin{aligned}r_{k+1} = &\{X_{ijt}S_{ip}S_{jp} \to C_{ip}X_{ijt} :\ 1 \le i < j < p \le N,\ 1 \le t \le k-1\} \cup \\ &\{X_{ijt}S_{ip}S_{pj} \to C_{ip}X_{ijt} :\ 1 \le i < p < j \le N,\ 1 \le t \le k-1\} \cup \\ &\{X_{ijt}S_{pi}S_{pj} \to C_{pi}X_{ijt} :\ 1 \le p < i < j \le N,\ 1 \le t \le k-1\}\end{aligned}$$

$$\begin{aligned}r_{k+2} = &\{X_{ijt}S_{ip} \to X_{ijt} :\ 1 \le i < p < j \le N,\ 1 \le t \le k-1\} \cup \\ &\{X_{ijt}S_{jp} \to X_{ijt} :\ 1 \le i < j < p \le N,\ 1 \le t \le k-1\} \cup \\ &\{X_{ijt}S_{pi} \to X_{ijt} :\ 1 \le p < i < j \le N,\ 1 \le t \le k-1\} \cup \\ &\{X_{ijt}S_{pj} \to X_{ijt} :\ 1 \le p < i < j \le N,\ 1 \le t \le k-1\} \cup \\ &\{X_{ijt}S_{pj} \to X_{ijt} :\ 1 \le i < p < j \le N,\ 1 \le t \le k-1\}\end{aligned}$$

$$\begin{aligned}r_{k+3} = &\{C_{ij} \to S_{ij} :\ 1 \le i < j \le N\} \cup \\ &\{\epsilon_{3k-1}X_{ijt}^{q-2}\gamma_{q-1} \to \epsilon_1\beta_0\gamma_{q-1} :\ 1 \le i < j \le N,\ 1 \le t \le k-1\}\end{aligned}$$

$$r'_{k+3} = \{\epsilon_{3k-1} \to \epsilon_1\beta_0\}$$

– Rules in the membrane labeled i $\{1 \le i \le N-1\}$:

$r_{k+4} = \{a_s e_{js} \to (S_{ij}, out) : \ 1 \le s \le k, \ i+1 \le j \le N\}$

$r_{k+5} = \{b_s d_{js} \to (S_{ij}, out) : \ 1 \le s \le k, \ i+1 \le j \le N\}$

- The partial order relation ρ over R consists of the following priority relations:
  – Priority relation on the membrane labeled i with $1 \le i \le N-1$: $\rho_i = \emptyset$
  – Priority relation on the skin membrane labeled N:
  $\rho_N = \{r_1 > r'_1 > r_2 > r'_2 > \ldots > r_{k-1} > r'_{k-1}\} \cup \{r_k > r'_k\} \cup$
  $\{r_{k+1} > r_{k+2} > r_{k+3} > r'_{k+3}\}$

3.2 An Overview of Computations

At the beginning of a computation the membrane labeled i, with $1 \le i \le N-1$, contains the objects $a_s, \ b_s, \ e_{js}, \ d_{js}$ with $1 \le s \le k$ and $i+1 \le j \le N$. In this membrane the presence or absence of the objects a_s, b_s encode the values of the individual ω_i. If the value of ω_{is} is equal to 1 we have the object a_s and if the value ω_{is} is equal to 0 we have the object b_s.

The objects e_{js}, d_{js} with $a \le s \le k$ and $i < j \le N$ are also in this membrane, and they codify the values of the k components of the individuals ω_j. If the value of the component s is 1, i.e. $\omega_{js} = 1$ then the membrane i contains the object e_{js}, if $\omega_{js} = 0$ then the membrane i contains the object d_{js}.

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 any configuration 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 in $3k-1$ steps the loop, that allows us to unite 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 in the environment the object $\sharp$.

In the initial configuration the only rules that can be applied in membrane labeled i with $1 \le i \le N-1$ are $r_{k+4}, \ r_{k+5}$, that send the objects S_{ij} with $1 \le i < j \le N$ to the skin membrane. The multiplicity of these objects allows us to know the similarity between individuals of the set Ω, that is the number of equal components between these individuals. In this configuration the rule r_0 constructs the object β_0.

¿From this configuration the computation of the P system is formed by loops of $3k-1$ steps. Each one of these loops is formed by two very differentiated stages. The first stage is formed by k steps and begins with the object β_0. In these steps the object S_{ij} with maximum multiplicity is selected encoding the maximum similarity between the clusters i and j. 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. This object represents the clusters that have the highest similarity, t, that can be joined to form a new cluster. Moreover, in this step the object γ_q is transformed in the object γ_{q-1}, encoding the fact that two clusters have been joined.

The second stage is formed by $2k-1$ steps. In the skin membrane there are the objects X_{ijt} meaning that a new cluster i is formed by the union of the previous clusters i, j. The rules $r_{k+1}, r_{k+2}, r_{k+3}$ calculate the similarities between new cluster i and the other clusters, this information is kept in the multiplicity of the objects S_{ip}.

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

The first partition consist of N singletons; in each loop two clusters are joined so it is necessary $N-1$ loops to obtain the last partition that consists of a cluster containing all N individuals. Therefore the loop repeats $N-1$ times and the rule r'_{k-1} is applied finalizing the P system.

3.3 Formal Verification

In this section we are going to show that the P system $\Pi(P_{Nk})$ is non-deterministic, but, in spite of this for any computation we will obtain 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 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 level $l \in \{1, 2, \ldots, N\}$, we denote by $\mathcal{C}_p(l)$ the multiset of objects contained in the membrane labeled l in the configuration $\mathcal{C}_p$.

The following result proves that in the configuration $\mathcal{C}_1$, the multiplicity of the object S_{ij}, $\forall 1 \leq i < j \leq N$, represents the similarity between the individual $\omega_i = (\omega_{i1}, \ldots, \omega_{ik})$ and the individual $\omega_j = (\omega_{j1}, \ldots, \omega_{jk})$.

Proposition 1. *Let $\mathcal{C}$ an arbitrary computation of the P system. If*
$t_{ij}^{(1)} = \max\{t : S_{ij}^t \in \mathcal{C}_1(N)\} \quad \forall i, j, t \quad (1 \le i < j \le N, \quad 1 \le t \le k-1)$
then $t_{ij}^{(1)} = \sum_{s=1}^k (1- \mid \omega_{is} - \omega_{js} \mid)$.

Proof. In the initial configuration we have

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

with $1 \le i \le N-1$.

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

Whenever one of these rules is applied the object S_{ij} goes out to the skin membrane. Then in $\mathcal{C}_1(N)$ the multiplicity of the objects S_{ij} will coincide with the number of equal components between the individuals ω_i and ω_j, i.e. $\mid \omega_{is} - \omega_{js} \mid = 0$. Therefore the multiplicity of the objects S_{ij} is

$$t_{ij}^{(1)} = \sum_{s=1}^{k} (1- \mid \omega_{is} - \omega_{js} \mid)$$

that is, $t_{ij}^{(1)}$ corresponds to the similarity between the individuals ω_i and ω_j. □

From now on we denote the maximum multiplicity of the objects S_{ij} in the step one of the n-th loop of the computation by

$$t_{ij}^{(n)} = \max\{t : S_{ij}^t \in \mathcal{C}_{1+(n-1)(3k-1)}(N)\}$$

In the following proposition we prove that each $3k-1$ steps is constructing the object β_0 so this object is in the skin of all the configurations of the type $1 + n(3k-1)$ with $1 \le n \le N-2$. Moreover we prove in what configuration the object ϵ_j with $1 \le j \le 3k-1$ appears.

The objects β_0 and ϵ_1 determine the moment that the loop starts and the object ϵ_{3k-1} determines when the loop finishes.

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

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

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

Proof. We prove this proposition by induction on n.

- For $n = 0$, it is necessary to verify that $\beta_0 \in \mathcal{C}_1(N)$, and $\epsilon_j \in \mathcal{C}_j(N)$, $\forall\ 1 \leq j \leq 3k-1$.
  In the initial configuration we have $\epsilon_0 \in \mathcal{C}_0(N)$ that allows us to apply one of the rules r_0 in order to obtain $\epsilon_1, \beta_0 \in \mathcal{C}_1(N)$, so *a)* is proved for $n = 0$.
  In the following $k-1$ steps the rules r_0 will be applied transforming the object ϵ_1 until we obtain the object $\epsilon_k \in \mathcal{C}_k(N)$. In this configuration if there are the objects $\alpha_{ijt}, \gamma_q \in \mathcal{C}_k(N)$ the rule r_k will be applied, or the rule r'_k will be applied. In both cases ϵ_k evolves to $\epsilon_{k+1} \in \mathcal{C}_{k+1}(N)$.
  In the 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 for $0 \leq n < N-2$. Then, we will show that $\epsilon_j \in \mathcal{C}_{1+(n+1)(3k-1)+(j-1)}(N)$, $\quad \forall\ 1 \leq j \leq 3k-1$ and $\beta_0 \in \mathcal{C}_{1+(n+1)(3k-1)}(N)$.
  By induction hypothesis $\epsilon_{3k-1} \in \mathcal{C}_{1+n(3k-1)+(3k-1-1)}(N) = \mathcal{C}_{(n+1)(3k-1)}(N)$.
  If in this configuration there is some object X_{ijt} the rules r_{k+3} will be applied and in the other case the rule r'_{k+3} will be applied. In both cases the object ϵ_{3k-1} is transformed in ϵ_1, $\beta_0 \in \mathcal{C}_{1+(n+1)(3k-1)}(N)$. So that *a)* is proved.

  Applying $k-1$ times the rules r_0 we obtain $\forall\ 1 \leq j \leq k$ that the object $\epsilon_j \in \mathcal{C}_{1+(n+1)(3k-1)+(j-1)}(N)$. In the configuration $\mathcal{C}_{1+(n+1)(3k-1)+(k-1)}(N)$ the object ϵ_k is transformed in the object $\epsilon_{k+1} \in \mathcal{C}_{1+(n+1)(3k-1)+k}(N)$ by means of one of the rules r_k or r'_k. Then applying the rules r_0 successively we obtain $\forall\ k+1 \leq j \leq 3k-1$ that the object $\epsilon_j \in \mathcal{C}_{1+(n+1)(3k-1)+(j-1)}(N)$.

□

Remark: According to Proposition 2 we have:

$$\epsilon_k \in \mathcal{C}_{1+n(3k-1)+k-1}(N) = \mathcal{C}_{k+n(3k-1)}(N) \quad \forall n \quad 0 \leq n \leq N-2$$

Corollary 1. *The objects X_{ijt} are sent to the environment at moments of the type $\mathcal{C}_{1+n(3k-1)+k}$ with $0 \leq n \leq N-2$.*

Proof. The only rule that sends some object X_{ijt} to the environment is the rule r_k. In order to be able to apply this rule the object ϵ_k is necessary, that verifies $\epsilon_k \in \mathcal{C}_{1+n(3k-1)+k-1}(N)$ with $0 \leq n \leq N-2$ by Proposition 2.

Therefore the objects X_{ijt} can only be sent to the environment in the following configuration, that is $X_{ijt} \in \mathcal{C}_{1+n(3k-1)+k}(env)$. □

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

Proof. Applying $(N-1)(3k-1)$ times the rule r_0 the object $\eta_0 \in \mathcal{C}_0(N)$ is transformed to $\eta_{(N-1)(3k-1)} \in \mathcal{C}_{(N-1)(3k-1)}(N)$. In this configuration the rule r'_{k-1} sends the halt object $\sharp$ to the environment. □

In the following result we prove that it is only possible to modify the environment in the k-th step of the loop.

Corollary 2. *Let $\mathcal{C}$ be an arbitrary computation of the P system. For each $0 \leq n \leq N-2$ the following assertions hold:*

a) For each 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 each 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. ¿From Proposition 3 the only rule that sends some objects to the environment before the halting configuration is the rule r_k. From Corollary 1 this rule sends the objects X_{ijt} to the environment in the configuration $\mathcal{C}_{1+n(3k-1)+k}$. Therefore, for each r $\forall r \quad 1+n(3k-1) < r < 1+n(3k-1)+k, \quad \mathcal{C}_r(env) = \mathcal{C}_{1+n(3k-1)}(env)$ and $\forall r \quad 1+n(3k-1)+k < r < 1+n(3k-1)+3k-1$ $\mathcal{C}_r(env) = \mathcal{C}_{1+n(3k-1)+k}(env)$ concluding the proof of a) and b). □

In the following results we will prove that in each loop one object X_{ijt} is sent to the environment. If a loop exists that doesn't send any object X_{ijt} to the environment in all the following loops no more objects are sent to the environment. Therefore a configuration always exists from which any object X_{ijt} is not sent to the environment.

Firstly we prove that if in the k-th step of the loop the rule r_k is not possible to be applied then in the following loop it is not possible to apply this rule either. This is because the objects S_{ij} do not exist in the skin membrane.

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

Proof. In order to apply the rule r_k it is necessary to have the objects ϵ_k, γ_q and α_{ijt}. According to Proposition 2 $\epsilon_k \in \mathcal{C}_{1+n(3k-1)+k-1}(N)$ for any n.

With the object γ_q only the rules r_k and r_{k+3} are applied, this object never disappears, so it always remains in the skin membrane.

The object α_{ijt} is produced by means of the rule r_u $(1 \leq u \leq k-1)$. In order to apply this rule it is necessary to have the object β_{u-1} and some object S_{ij}. The object β_{u-1} is produced by means of the rules $r'_1, r'_2, \ldots, r'_{u-1}$.

Therefore if in the configuration $\mathcal{C}_{1+n(3k-1)+k}$ the rule r_k cannot be applied, it is because the object α_{ijt} does not exist, then the objects $S_{ij} \in \mathcal{C}_{1+n(3k-1)}(N)$ do not exist.

¿From the configuration $\mathcal{C}_{1+n(3k-1)+k-1}$ to the configuration $\mathcal{C}_{1+(n+1)(3k-1)+k-1}$ in the skin membrane the only rule that can produce the objects S_{ij} is the rule r_{k+3}. To apply this rule, the objects C_{ij} are necessary, that are produced in the rule r_{k+1} from the objects S_{ij}. As the objects S_{ij} do not exist the rule r_{k+1} cannot be applied. □

The following result proves that if the environment in the k-th step of the loop $n+1$ is equal to the environment in the k-th step of the loop n then the environment is the same until the halting configuration.

Corollary 3. *For each* n $(0 \leq n \leq N-2)$ *if*

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

then for each n' $(n \leq n' \leq N-2)$ *we have*

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

Proof. We prove by induction that

$$\mathcal{C}_{1+(n+j)(3k-1)+k}(env) = \mathcal{C}_{1+(n+j+1)(3k-1)+k}(env) \quad \forall j (0 \leq j \leq N-n-3)$$

- The case base, $j = 0$, corresponds to the hypothesis of the corollary, so $\mathcal{C}_{1+n(3k-1)+k}(env) = \mathcal{C}_{1+(n+1)(3k-1)+k}(env)$.
- We suppose true for the cases $0 \leq j < N-n-3$. Let us show that the result is true for $j+1$.
  By induction hypothesis we have

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

  that is, in the previous configuration to these it has not been able to send any object to the environment, that is the rule r_k has not been possible to apply. By Proposition 4 if in the configuration $\mathcal{C}_{1+(n+j+1)(3k-1)+k-1}$ cannot be applied the rule r_k in the configuration $\mathcal{C}_{1+(n+j+2)(3k-1)+k-1}$ cannot be applied either. Therefore it is not possible to send any object to the environment and $\mathcal{C}_{1+(n+j+1)(3k-1)+k}(env) = \mathcal{C}_{1+(n+j+2)(3k-1)+k}(env)$. □

We are going to prove that a loop always exists from any object X_{ijt} is sent to the environment, so it is not possible to apply the rule r_k.

Corollary 4. *For each computation* $\mathcal{C}$ *there exists an unique object* $\nu_{\mathcal{C}}$ $(1 \leq \nu_{\mathcal{C}} \leq N-2)$ *such that in the configuration* $\mathcal{C}_{1+(\nu_{\mathcal{C}}-1)(3k-1)+k}$ *the rule* r_k *is applicable and in the configuration* $\mathcal{C}_{1+(\nu_{\mathcal{C}})(3k-1)+k}$ *the rule* r_k *is not applicable.*

Proof. By Proposition 4 and by Corollary 3 if in the configuration $\mathcal{C}_{1+(\nu_{\mathcal{C}})(3k-1)+k-1}$ the rule r_k is not applicable, then for each j $(\nu_{\mathcal{C}} \leq j \leq N-2)$ we have $\mathcal{C}_{1+\nu_{\mathcal{C}}(3k-1)+k}(env) = \mathcal{C}_{1+j(3k-1)+k}(env)$.
Therefore, the rule r_k is not applicable in any configuration of the type $\mathcal{C}_{1+j(3k-1)+k-1}$, $\forall j \quad \nu_{\mathcal{C}} \leq j \leq N-2$. □

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

Proposition 5. *Let $\mathcal{C}$ be an arbitrary computation of the P system and let the object $X_{i_nj_nt^{(n)}_{i_nj_n}}$ that is sent to the environment by the rule r_k in the configuration $\mathcal{C}_{(k+1)+n(3k-1)-1}$. Then, we have*

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

Proof. As the rule r_k is applicable in the configuration $\mathcal{C}_{(k+1)+n(3k-1)-1}$ then $\alpha_{i_nj_nt^{(n)}_{i_nj_n}} \in \mathcal{C}_{(k+1)+n(3k-1)-1}$. The object $\alpha_{i_nj_nt^{(n)}_{i_nj_n}}$ 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 another $t' > t^{(n)}_{i_nj_n}$ exists then the rule $r_{k-t'}$ will be applied and so the rule $r_{k-t^{(n)}_{i_nj_n}}$ has not been applied. □

The following result proves that the maximum multiplicity to the objects S_{ij} pertaining to the skin membrane in any loop n is always greater or equal to the multiplicity the objects S_{ij} of the following loop $n+1$.

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

Proof. If $w_n = \max\{t : S^t_{ij} \in \mathcal{C}_{1+n(3k-1)}(N)\}$, in the following configurations $\mathcal{C}_{1+n(3k-1)}$ the rule r_0 is applied successively and the rules with priority $r'_1, r'_2, \ldots, r'_{w_n-1}, r_{w_n}$ until arriving at the configuration $\mathcal{C}_{1+n(3k-1)+w_n}$. The object S_{ij} is not used in the rules $r'_1, r'_2, \ldots, r'_{w_n-1}$. For Proposition 5 in the rule r_{w_n} is used the object S_{ij} that have the maximum multiplicity equal to w_n. By this rule the object S_{ij} is eliminated by the membrane labeled by N, therefore $w_n \geq \max\{t : S^t_{ij} \in \mathcal{C}_{1+n(3k-1)+w_n}(N)\}$.

From this configuration the rule r_0 is applied $k - w_n$ times until we obtain the object ϵ_k. In these configurations the objects S_{ij} do not evolve.

- If $w_n \neq 0$, then in the configuration $\mathcal{C}_{1+n(3k-1)+k-1}$ when the rule r_{k+1} is applied the information of some objects S_{ij} is sent to the object C_{ij} and later this information is transformed in the object S_{ij} by means of the rule r_{k+3}. The rule r_{k+2} deletes some objects S_{ij}, so the multiplicity of these objects never increases, it is only possible to decrease. After that the rule r_0 is applied since to arrive at the configuration $\mathcal{C}_{1+(n+1)(3k-1)}$. So, $w_{n+1} = \max\{t : S^t_{ij} \in \mathcal{C}_{1+(n+1)(3k-1)}(N)\} \leq w_n$.
- If $w_n = 0$, then 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}, so: $w_{n+1} = \max\{t : S^t_{ij} \in \mathcal{C}_{1+(n+1)(3k-1)}(N)\} = 0$. □

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

By the following result we show that if a loop goes out to the environment an object of the type X_{ijt}, then in the following loop the objects $S_{ij}, S_{i'j}, S_{ji'}$,

$i' \notin \{i,j\}$ disappears 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 $\mathcal{C}$ be an arbitrary computation of the P system. Let*

$$X_{i_1 j_1 t^{(1)}_{i_1 j_1}}, X_{i_2 j_2 t^{(2)}_{i_2 j_2}}, \ldots, X_{i_n j_n t^{(n)}_{i_n j_n}} \in \mathcal{C}_{1+n(3k-1)}(env), \text{ with } 1 \leq n \leq \nu_{\mathcal{C}}$$

If $S_{ij} \in \mathcal{C}_{1+n(3k-1)}(N)$ then

a) $(i,j) \notin \{(i_1,j_1),\ldots,(i_n,j_n)\}$.
b) $\{i,j\} \in \{1,\ldots,N\} - \{j_1,\ldots,j_n\}$.

Proof. We prove the result by induction on n.

- For n=1.
  If in the configuration $\mathcal{C}_k$ the rule r_k sends the object $X_{i_1 j_1 t^{(1)}_{i_1 j_1}}$ to the environment, then by Proposition 6 in the configuration $\mathcal{C}_{k-t^{(1)}_{i_1 j_1}}$ with $1 \leq t^{(1)}_{i_1 j_1} < k$ the rule $r_{k-t^{(1)}_{i_1 j_1}}$ has had to apply so the objects $S_{i_1 j_1}$ have disappeared. Therefore the objects $S_{ij} \in \mathcal{C}_{1+(3k-1)}(N)$ verify that $(i,j) \notin \{(i_1,j_1)\}$.
  In the following configurations when we apply the rule r_{k+1} the pairs of objects $(S_{i_1 p}, S_{j_1 p}), (S_{i_1 p}, S_{p j_1}), (S_{p i_1}, S_{p j_1})$ are transformed respectively to the objects $C_{i_1 p}, C_{i_1 p}, C_{p i_1}$, these objects do not have the subscript j_1. After that the rule r_{k+2} is applied in order to eliminate the objects
  $S_{i_1 p}, S_{j_1 p}, S_{p i_1}, S_{p j_1} \in \mathcal{C}_1$ that have not been eliminated in the previous configurations. By these rules all the objects S_{ij} with $\{i,j\} \cap \{i_1,j_1\} \neq \emptyset$ have disappeared.
  After that when we apply the rule r_{k+3} the objects $C_{ij},\ i_1 \in \{i,j\}$ are transformed in the objects $S_{ij},\ i_1 \in \{i,j\}$. Therefore if the objects $S_{ij} \in \mathcal{C}_{1+(3k-1)}$ then $(i,j) \notin \{(i_1 j_1)\},\ \{i,j\} \in \{1,\ldots,N\} - \{j_1\}$.

- Let us suppose the proposition holds for $1 \leq n < \nu_{\mathcal{C}}$. Let us show that the the result is held for $n+1$.
  If the object $X_{i_{n+1} j_{n+1} t^{(n+1)}_{i_{n+1} j_{n+1}}} \in \mathcal{C}_{k+n(3k-1)}(env)$ by Proposition 6 in the configuration $\mathcal{C}_{k-t^{(n+1)}_{i_{n+1} j_{n+1}}+n(3k-1)}$ with $1 \leq t^{(n+1)}_{i_{n+1} j_{n+1}} < k$ the rule $r_{k-t^{(n+1)}_{i_{n+1} j_{n+1}}}$ has had to apply and the objects $S_{i_{n+1} j_{n+1}}$ have disappeared. Therefore the objects $S_{ij} \in \mathcal{C}_{1+(n+1)(3k-1)}(N)$ verify that $(i,j) \notin \{(i_{n+1}, j_{n+1})\}$, by induction hypothesis $(i,j) \notin \{(i_1,j_1),\ldots,(i_n,j_n)\}$ then
  $(i,j) \notin \{(i_1,j_1),\ldots,(i_{n+1},j_{n+1})\}$.
  In the following configurations when we apply the rule r_{k+1} the pairs of objects $(S_{i_{n+1} p}, S_{j_{n+1} p}), (S_{i_{n+1} p}, S_{p j_{n+1}}), (S_{p i_{n+1}}, S_{p j_{n+1}})$ are transformed respectively in the objects $C_{i_{n+1} p}, C_{i_{n+1} p}, C_{p i_{n+1}}$, these objects do not have the subscript j_{n+1}. After that the rule r_{k+2} is applied in order to eliminate the objects

$S_{i_{n+1}p}, S_{j_{n+1}p}, S_{pi_{n+1}}, S_{pj_{n+1}} \in \mathcal{C}_{1+n(3k-1)}$ that have not been eliminated in the previous configurations, with these two rules all the objects S_{ij} with $\{i,j\} \cap \{i_{n+1}, j_{n+1}\} \neq \emptyset$ have disappeared.
When we apply the rule r_{k+3} the objects $C_{ij},\ i_{n+1} \in \{i,j\}$ are transformed in the objects $S_{ij},\ i_{n+1} \in \{i,j\}$.
So if $S_{ij} \in \mathcal{C}_{1+(n+1)(3k-1)}(N)$ then
$(i,j) \notin \{(i_{n+1}j_{n+1})\},\ \{i,j\} \in \{1,\ldots,N\} - \{j_{n+1}\}$.
And by the induction hypothesis $S_{ij} \in \mathcal{C}_{1+(n+1)(3k-1)}(N)$
$(i,j) \notin \{(i_1j_1),\ldots,(i_nj_n)\},\ \{i,j\} \in \{1,\ldots,N\} - \{j_1\ldots j_n\}$ therefore
$(i,j) \notin \{(i_1j_1),\ldots,(i_{n+1}j_{n+1})\},\ \{i,j\} \in \{1,\ldots,N\} - \{j_1\ldots j_{n+1}\}$
concluding the proof of b). □

In the following proposition we study how the multiplicities of the objects S_{ij} changed at the moment that two clusters joined.

Proposition 8. *Let $\mathcal{C}$ be an arbitrary computation of the P system. Let us suppose that $X_{i_1j_1t^{(1)}_{i_1j_1}}, X_{i_2j_2t^{(2)}_{i_2j_2}}, \ldots, X_{i_nj_nt^{(n)}_{i_nj_n}} \in \mathcal{C}_{1+n(3k-1)}(env)$ with $1 \leq n \leq \nu_{\mathcal{C}}$, and $t^{(n)}_{ij} = max\{t : S^t_{ij} \in \mathcal{C}_{1+n(3k-1)}\}$. Then,*

- *If $i_n \notin \{i,j\}$ then $t^{(n+1)}_{ij} = t^{(n)}_{ij}$. That is, the multiplicity of the objects S_{ij} is the same in the configurations $\mathcal{C}_{1+n(3k-1)}$ and $\mathcal{C}_{1+(n+1)(3k-1)}$.*
- *If $1 \leq i_n < j_n < p \leq N$ then $t^{(n+1)}_{i_np} = \min\{t^{(n)}_{i_np}, t^{(n)}_{j_np}\}$. That is, the multiplicity of the objects S_{i_np} corresponds to the minimum multiplicity of the objects S_{i_np}, S_{j_np}.*
- *If $1 \leq i_n < p < j_n \leq N$ then $t^{(n+1)}_{i_np} = \min\{t^{(n)}_{i_np}, t^{(n)}_{pj_n}\}$. That is, the multiplicity of the object S_{i_np} corresponds to the minimum multiplicity of the objects S_{i_np}, S_{pj_n}.*
- *If $1 \leq p < i_n < j_n \leq N$ then $t^{(n+1)}_{pi_n} = \min\{t^{(n)}_{pi_n}, t^{(n)}_{pj_n}\}$. That is, the multiplicity of the object S_{pi_n} corresponds to the minimum multiplicity of the objects S_{pi_n}, S_{pj_n}.*

Proof. For the proof of Proposition 7 when we apply the rule r_{k+1} in the configuration $\mathcal{C}_{k+(n-1)(3k-1)}$ the pairs of objects $(S_{i_np}, S_{j_np}), (S_{i_np}, S_{pj_n}), (S_{pi_n}, S_{pj_n})$ are transformed respectively in the objects $C_{i_np}, C_{i_np}, C_{pi_n}$. Therefore the number of objects $C_{i_np}, C_{i_np}, C_{pi_n}$ are the same respectively of the pairs of the objects $(S_{i_np}, S_{j_np}), (S_{i_np}, S_{pj_n}), (S_{pi_n}, S_{pj_n})$. After that the rule r_{k+2} is applied in order to eliminate the objects $S_{i_np}, S_{j_np}, S_{pi_n}, S_{pj_n} \in \mathcal{C}_{1+(n-1)(3k-1)}$ that have not been eliminated in the previous configurations. So the multiplicity of the objects $C_{i_np}, C_{i_np}, C_{pi_n}$ is respectively equal to $\min\{t^{(n)}_{i_np}, t^{(n)}_{j_np}\}, \min\{t^{(n)}_{i_np}, t^{(n)}_{pj_n}\}$, and $\min\{t^{(n)}_{pi_n}, t^{(n)}_{pj_n}\}$. When we apply the rule r_{k+3} the objects C_{ij} are transformed in the objects S_{ij}. □

Next, we define how the partition of the individuals is formed from the objects sent to the environment.

Definition 6. *Given a computation $\mathcal{C}$ of the P system we denote the succession of partitions of the set of the individuals by $\Delta_0^{\mathcal{C}},\ \Delta_1^{\mathcal{C}},\ \ldots,\Delta_\theta^{\mathcal{C}}$. These partitions are constructed recursively as follows:*

The initial partition is formed by the initial individuals,
$\Delta_0^{\mathcal{C}} = \{B^0_{q_0^1},\ \ldots,\ B^0_{q_0^N}\}$ *with* $q_i^0 = i$ *y* $B_i^0 = \{\omega_i\} \equiv \{i\}$

The partition $\Delta_1^{\mathcal{C}}$ is constructed from the object $X_{i_1 j_1 t^{(1)}_{i_1 j_1}} \in \mathcal{C}_{k+1}(env)$ with $1 \le i_1 < j_1 \le N$ as follows:

As $\{q_0^1, \ldots, q_0^N\} = \{1, \ldots, N\}$ then $\{i_1, j_1\} \subseteq \{q_0^1, \ldots, q_0^N\}$.
If $i_1 = q_0^u$, $j_1 = q_0^s$ with $1 \le u < s \le N$, then the new cluster is
$B^1_{q_1^u} = B^0_{q_0^u} \cup B^0_{q_0^s}$ *with* $q_1^u = q_0^u$ *and*

$B^0_{q_0^s} \notin \Delta_1^{\mathcal{C}}$
$B_l^1 = B_l^0$ *for* $l \in \{q_0^1, \ldots, q_0^N\} - \{q_0^u, q_0^s\}$

Then the new partition obtained is $\Delta_1^{\mathcal{C}} = \{B^1_{q_1^1},\ \ldots,\ B^1_{q_1^{N-1}}\}$

In a recursive manner we obtain the partition $\Delta_{n+1}^{\mathcal{C}}$ as follows:
¿From the objects$X_{i_1 j_1 t^{(1)}_{i_1 j_1}}, \ldots X_{i_n j_n t i_n j_n{}^{(n)}} \in \mathcal{C}_{(k+1)+(n-1)(3k-1)}(env)$ the partition $\Delta_n^{\mathcal{C}} = \{B^n_{q_n^1}, \ldots, B^n_{q_n^{N-n}}\}$ have been obtained and in the configuration $\mathcal{C}_{(k+1)+n(3k-1)}$ the object $X_{i_{n+1} j_{n+1} t^{(n+1)}_{i_{n+1} j_{n+1}}}$ with $1 \le i_{n+1} < j_{n+1} \le N$ is sent to the environment.

By Proposition 7, $\{i_{n+1}, j_{n+1}\} \in \{1, \ldots, N\} - \{j_1, j_2, \ldots, j_n\}$ therefore $i_{n+1} < j_{n+1}$ and $\{i_{n+1}, j_{n+1}\} \subseteq \{q_n^1, \ldots, q_n^{N-n}\}$.
By construction of the partition it is verified $\{q_n^1, \ldots, q_n^{N-n}\} \subset \{1, \ldots, N\}$ and we have $\{j_1, j_2, \ldots, j_n\} \cap \{q_n^1, \ldots, q_n^{N-n}\} = \emptyset$.

Let $i_{n+1} = q_n^u$, $j_{n+1} = q_n^s$ with $1 \le u < s \le N$, then the new cluster is
$B^{n+1}_{q^u_{n+1}} = B^n_{q_n^u} \cup B^n_{q_n^s}$ *with* $q^u_{n+1} = q^u_n$
$B^n_{q_n^s} \notin \Delta^{\mathcal{C}}_{n+1}$
$B_l^{n+1} = B_l^n$ *with* $l \in \{q_n^1, \ldots, q_n^{N-n}\} - \{q_n^u, q_n^s\}$

Then $\Delta^{\mathcal{C}}_{n+1} = \{B^{n+1}_{q^1_{n+1}},\ \ldots,\ B^{n+1}_{q^{N-n-1}_{n+1}}\}$

Theorem 1. *Let $\mathcal{C}$ be an arbitrary computation of the P system. Let us suppose that $S_{ij}^{t_{ij}^{(n)}} \in \mathcal{C}_{1+n(3k-1)}(N)$ and $S_{ij}^{t_{ij}^{(n)}+1} \notin \mathcal{C}_{1+n(3k-1)}(N)$, with $1 \le n \le \nu_{\mathcal{C}}$. Then, $t_{ij}^{(n)}$ is the minimum similarity between any pair of individuals pertaining to $B_i^{n-1} \cup B_j^{n-1}$. That is, $t_{ij}^{(n)}$ corresponds to the aggregation index between the groups B_i^{n-1} and B_j^{n-1}:*

$$t_{ij}^{(n)} = \min\{t' : S^{t'}_{i'j'} \in \mathcal{C}_1(N), \quad S^{t'+1}_{i'j'} \notin \mathcal{C}_1(N) :\ i', j' \in B_i^n \cup B_j^n\}$$

Proof. We prove the theorem by induction on n.

- For $n = 0$.
  By Proposition 1 if $S_{ij}^{t_{ij}^{(0)}} \in \mathcal{C}_1(N)$ then $t_{ij}^{(0)}$ corresponds to the similarity between individuals i, j.

- Let us suppose it is certain for n with $1 \leq n < \nu_{\mathcal{C}}$.

- Let us show that the theorem is held for $n + 1$.
  In the configuration $\mathcal{C}_{(k+1)+(n-1)(3k-1)}$ the partition $\Delta_n^{\mathcal{C}} = \{B_{q_n^1}^n, \ldots, B_{q_n^{N-n}}^n\}$ is obtained and in the configuration $\mathcal{C}_{(k+1)+n(3k-1)}$ the object $X_{i_{n+1}j_{n+1}t_{i_{n+1}j_{n+1}}^{(n+1)}}$ with $1 \leq i_{n+1} < j_{n+1} \leq N$ is sent to the environment, then:

  – For Proposition 8 if $i_{n+1} \notin \{i, j\}$ then $t_{ij}^{(n+1)} = t_{ij}^{(n)}$. By the induction hypothesis $t_{ij}^{(n)}$ is the aggregation index between the groups i, j and for the definition 6, $B_i^{n+1} = B_i^{n+1}$, $B_j^{n+1} = B_j^{n+1}$, then the theorem is true.
  – For Proposition 8 we have $t_{i_{n+1}j}^{(n+1)} = \min\{t_{i_{n+1}j}^{(n)}, t_{j_{n+1}j}^{(n)}\}$ and by induction hypothesis $t_{i_{n+1}j}^{(n)}, t_{j_{n+1}j}^{(n)}$ corresponds to the minimum similarity between any pair of individuals pertaining respectively to $B_{i_{n+1}}^{n-1} \cup B_j^{n-1}, B_{j_{n+1}}^{n-1} \cup B_j^{n-1}$. Therefore $t_{i_{n+1}j}^{(n+1)}$ is the minimum of these two similarities and $B_{i_{n+1}}^n = B_{i_{n+1}}^{n-1} \cup B_{j_{n+1}}^{n-1}$ then it is verified that the minimum similarity between any pair of individuals pertaining to $B_{i_{n+1}}^n \cup B_j^n$. □

The following result proves that if in the configuration $\mathcal{C}_{(k+1)+(n-1)(3k-1)}$ the object $X_{i_nj_nt_{i_nj_n}^{(n)}}$ goes out to the environment, then when we form the partition $\Delta_n^{\mathcal{C}}$ the similarities between two individuals obtained in each set of $\Delta_n^{\mathcal{C}}$ are always greater or equal than $t_{i_nj_n}^{(n)}$.

Proposition 9. *Let $\mathcal{C}$ be an arbitrary computation of the P system. Let us suppose that in the configuration $\mathcal{C}_{(k+1)+(n-1)(3k-1)}$, with $0 \leq n \leq \nu_{\mathcal{C}} - 1$, the object $X_{i_nj_nt_{i_nj_n}^{(n)}}$ is sent to the environment, and the new partition is $\Delta_n^{\mathcal{C}} = \{B_{q_n^1}^n, \ldots, B_{q_n^{N-n}}^n\}$. Then, the hierarchical index of the cluster $B_{i_n}^n$ is $t_{i_nj_n}^{(n)}$, and the hierarchical index of the rest of clusters $\Delta_n^{\mathcal{C}} - \{B_{i_n}^n\}$ is greater or equal to $t_{i_nj_n}^{(n)}$.*

Proof. We prove this result by induction on n.
For Definition 6, $\Delta_0^{\mathcal{C}} = \{\{\omega_1\}, \ldots, \{\omega_N\}\}$ and $f(\{\omega_i\}) = k$ with $1 \leq i \leq N$.

- For $n = 1$.
  $\Delta_1^{\mathcal{C}} = \{B_{q_1^1}^1, \ \ldots, \ B_{q_1^{N-1}}^1\}$ with $B_{i_1} = \{\omega_{i_1}, \omega_{j_1}\}$ and $B_j = \{\omega_j\}$, $\forall j \neq i_1$.
  Then $f(B_j) = k$ with $j \neq i_1$
  and $f(B_{i_1}) = t_{i_1j_1}^{(1)}$ because $s(\omega_{i_1}, \omega_{j_1}) = t_{i_1j_1}^{(1)} \leq k - 1$.

- We suppose that is true for any $1 \leq n \leq \nu_{\mathcal{C}}$. Let us show that the result holds for $n+1$.
  In the configuration $\mathcal{C}_{(k+1)+n(3k-1)}$ the rule r_k sends the object $X_{i_{n+1}j_{n+1}t^{(n+1)}_{i_{n+1}j_{n+1}}}$ to the environment, where $t^{(n+1)}_{i_{n+1}j_{n+1}}$ is the similarity between the clusters i_{n+1} and j_{n+1}. By construction of the P system $t^{(n+1)}_{i_{n+1}j_{n+1}} \leq t_n$ and the partition is $\Delta^{\mathcal{C}}_{n+1} = \{B^{n+1}_{q^1_{n+1}}, \ldots, B^{n+1}_{q^{N-n-1}_{n+1}}\}$ for Definition 6:
  - If $i_{n+1}, j_{n+1} \in B^n_{q^l_n}$ then $B^{n+1}_{q^l_{n+1}} = B^n_{q^l_n}$ and for induction hypothesis $f(B^{n+1}_{q^l_{n+1}}) = t^{(n)}_{i_n j_n} \geq t^{(n+1)}_{i_{n+1}j_{n+1}}$.
  - If $B^{n+1}_{i_{n+1}} = B^n_{i_{n+1}} \cup B^n_{j_{n+1}}$ and $\delta(B^n_{i_{n+1}},\ B^n_{j_{n+1}}) = t^{(n+1)}_{i_{n+1}j_{n+1}}$, then $f(B^{n+1}_{i_{n+1}}) = t^{(n+1)}_{i_{n+1}j_{n+1}}$. □

Proposition 10. *The P system $\Pi(P_{Nk})$ allows us to find a hierarchical clustering.*

Proof. ¿From the partition $\Delta^{\mathcal{C}}_0,\ \Delta^{\mathcal{C}}_1,\ \ldots, \Delta^{\mathcal{C}}_\theta$ according to Definition 6 we can obtain an indexed hierarchy $P_0, P_1, \ldots, P_m$ with $1 \leq m \leq N-1$.
By Proposition 9 all partitions $\Delta^{\mathcal{C}}_n$ have a hierarchical index $t_n = t^{(n)}_{i_n j_n}$ and we denoted by $(\Delta^{\mathcal{C}}_0, t_0), (\Delta^{\mathcal{C}}_1, t_1), \ldots, (\Delta^{\mathcal{C}}_\theta, t_\theta)$ with
$t_0 > t_1 = t_2 = \ldots t_{p_1} > t_{p_1+1} = \ldots = t_{p_2} > \ldots > t_{p_m} = \ldots = t_\theta$.

In order to construct the partition $P_0, P_1, \ldots, P_m$ we do the following steps:

- $P_0 = \Delta_0 = \{\{\omega_1\}, \{\omega_2\}, \ldots, \{\omega_N\}\}$.
- The partitions $\Delta_1, \ldots, \Delta_{p_1}$ have associated a hierarchical index equal to t_1 so $P_1 = \Delta_{p_1}$.
- The partitions $\Delta_{p_1+1}, \ldots, \Delta_{p_2}$ have associated a hierarchical index equal to t_{p_2} so $P_2 = \Delta_{p_2}$.
- And successively until we have one of the following situations:
  - if Δ_θ has a hierarchical index $t_\theta = k-1$ then $P_m = \Delta_\theta = \Omega$.
  - if Δ_θ has a hierarchical index $t_\theta < k-1$ then $P_{m-1} = \Delta_\theta$ and $P_m = \Omega$.

□

4 Conclusions

One of the central issues when we have a set of individuals 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 give 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 of N individuals and of the number of k characteristics. The amount of resources initially required to construct the system is quadratic in N and k.

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

References

1. M.Acoub, F. Badran, S. Thiria. A topological hierarchical clustering: Application to ocean color classification, *Lecture Notes in Computer Science*, **2130** (2001), 492–499.
2. E. Diday, J. Lemaire, J. Pouget, F. Testu. *Éléments d ' analyse de données*, Dunod, Paris, 1982.
3. R.O. Duda, P.E. Hart. *Pattern Classification and Scene Analysis*, John Wiley, New York, 1973.
4. B. Everitt. *Cluster Analysis*, John Wiley & Sons. London, 1980.
5. S.C. Johnson. Hierarchical Clustering Schemes, *Psychometrika*, **32** (1967), 241–254.
6. Gh. Păun. Computing with membranes. *Journal of Computer and System Sciences*, **61**, 1 (2000), 108–143, and *Turku Center for Computer Science-TUCS Report* Nr. 208, 1998.
7. Gh. Păun. *Membrane Computing. An Introduction.* Springer-Verlag, Berlin, 2002.
8. Gh. Păun, G. Rozenberg. A guide to membrane computing. *Theoretical Computer Science*, 287 (2002), 73–100.
9. J. Razmi, M. Zairi, A. Gunasekaran. The application of analytic hierarchy process in classification of material planning and control systems. *International Journal of Services and Operations Management*, **2**, 4 (2006), 352–366.
10. R.R. Sokal, P.H.A. Sneath. *Principles of Numerical Taxonomy*, Freeman & Co. London, 1963.
11. H. Späth, U. Bull. *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*, John Wiley & Sons, New York, 1980.
12. M. Ture, I. Kurt, A.T. Kurum, K. Ozdamar. Comparing classification technique for predicting essential hypertension. *Expert Systems with Applications*, **29** (2005), 583-588.
13. G. Tzanetakis, G. Essl, P. Cook. *Automatic Musical Genre Classification Of Audio Signals*, CiteSee.IST-Copyright Penn State and NEC.
14. X.H. Zhang, J.M. Chen. Application of hierarchical cluster with high-performance liquid chromatography in the classification of genus Clinopidium chinesis. *Zhongguo Zhong Yao Za Zhi*, **28**, 9 (2003), 812–816.
15. ISI web page: `http://esi-topics.com/erf/october2003.html`