Density-Based Clustering by P System with Active Membranes on Commodity Recommendation in E-commerce Websites

JIE SUN, XIYULIU
School of Management Science and Engineering
Shandong Normal University
Jinan, Shandong
CHINA
sjiezz@163.com, sdxyliu@163.com

Abstract: - With the Popularity of shopping online in people's daily economic life, the commodity recommendation mechanism in e-commerce platform is presented to help customers quickly and accurately find the suitable product. Using the possibility of changing membrane structure, a variant of P system with active membranes is proposed to solve commodity recommendation problems. In this paper, the commodity recommendation problem is transformed into a density-based clustering problem firstly. Then it specifies the procedure of realizing this problem and a P system with a sequence of new rules is designed. The computation complexity of DBSCAN clustering algorithm in this system is $O(n \log n)$, while the original DBSCAN clustering algorithm is $O(n^2)$ without spatial query. This new model of P system can reduce the computation complexity of clustering process and improve the efficiency to solve the problems of commodity recommendation. Through example verification, this new model of P system is proved to be feasible and effective to achieve this practical issue.

Key-Words: - Membrane Computing; P System; Active Membranes; DBSCAN; Commodity Recommendation

1 Introduction

With the Popularity of the Internet, e-commerce systems has more involved with people's daily economic life. Compared with traditional business, the advantages of e-commerce are not only reflected in the abundant and cheap goods, but reflected in the personalization. Along with the rapid development of e-commerce and the growth of online products, its structure has become more complex, users are often lost in a large number of goods in the information space, and could not find their own needs. For this reason, various recommendation systems came into being. The most useful technology of e-commerce recommendation is the data mining technology. Data mining technology is mainly used to cluster the information of consumers and commodities, which will help to provide personalized recommendations to target consumers. As an important part of data mining, clustering is the process of grouping the data into classes or clusters [1, 2].

Membrane computing is a branch of natural computing which abstracts computing models from the architecture and the functioning of living cells, as well as from the organization of cells in tissues, organs. It was presented by Paun in 1998 [3], so now all classes of computing devices considered in membrane computing are generically called P system. It has been applied in broad fields such as biological modeling, combinatorial problems and NPC problems on the strength of its great parallelism and non-determinacy. Its characteristics can improve the efficiency of clustering process so it is able to solve clustering problems [4].

A particularly interesting class of P systems is the P system with active membranes. In a P system with active membranes not only the objects evolve but also the membrane structure. In this paper, we combine the density-based clustering algorithm with commodity recommendation in e-commerce websites, with the help of a P system with active membranes which is adequate for clustering problems. It has great significance to the application of membrane computing in a typical real-world case of clustering problems.

2 Density-Based Clustering Problem

2.1 Commodity recommendation in e-commerce websites

Under the e-commerce mode, the companies hope to pinpoint the potential demand of target consumers

and implement one-to-one marketing on the basis of the actual situation of each consumer. And consumers hope that e-commerce platform can be able to take the initiative to recommend products, something they may be of interest, with no need for spending a lot of time and effort to search for the information and commodities. For this reason, the commodity recommendation mechanism in e-commerce platform is presented to help customers quickly and accurately find the product and promote the competitiveness of enterprises. In 1992, Goldberg in Xerox Palo Alto Research Center first explicitly put forward the idea of personalized information recommendation. At the present stage, the recommendation algorithms frequently used include the following several kinds: Content-based Recommendation (CBR), Collaborative Filtering Recommendation (CFR) and Hybrid Recommendat--ion [5].

E-commerce recommendation system is an intelligent information system. It predicts the commodities preference of target user by collecting and analyzing the historical behavior data of previous customers and takes the initiative to recommend relevant commodity information for target users. Specifically speaking, we group the commodities with high similarities into a cluster by clustering algorithm. Consequently, when a customer purchases one product in a cluster, the commodity recommendation mechanism will automatically recommend other products in this cluster.

2.2 Transform commodity recommendation into Density-Based Clustering Problem

In this section, we use DBSCAN clustering algorithm, one of the popular density-based clustering method, to solve our problem. We refer to [6] and [7] for a more precise knowled--ge of DBSCAN, and here we only simply describe the definition of it. DBSCAN (Density-Based Spatial Clustering of Applicati--on with Noise) algorithm is a kind of spatial clustering algorithm based on density. After inputting parameters ε and *Min*, the algorithm grows regions with sufficiently high density into clusters and discovers clusters of arbitrary shape in spatial databases with noise effectively.

In order to apply DBSCAN clustering algorithm to commodity recommendation associated with historical behaviour data of previous customers, we need to treat every commodity as an independent entity.

Definition 1. We define a k-set Ω of commodities, with $\Omega = \{p_1, p_2, \dots, p_i, \dots, p_n\}$ $(i = 1,2, \dots, n)$, as the dataset to classify.

Where $p_i = \{p_{i1}, p_{i2}, \dots, p_{ij}, \dots, p_{im}\}$. p_i stands for the commodity i, and p_{ij} stands for the amount that customer j purchases the commodity i and its value is denoted by $\{0,1,2, \dots, n\}$.

Definition 2. A similarity over a k-set $\Omega = \{p_1, p_2, \dots, p_i, \dots, p_n\}$ is a function S that verifies:

- S is symmetric, that is
  $\forall(p_i, p_j) \in \Omega \times \Omega$: $S(p_i, p_j) = S(p_j, p_i)$
- $\forall(p_i, p_j) \in \Omega \times \Omega$: $S(p_i, p_j) \in [0,1]$
- $\forall p_i, p_j \in \Omega$ with $i \neq j$:
  $S(p_i, p_i) = S(p_j, p_j) \geq S(p_i, p_j)$

In this paper the similarity between two commodities i and j, defined as $S(p_i, p_j)$, is calculated by the following equation:

$$S(p_i, p_j) = (|p_i| \cap |p_j|)/(|p_i| \cup |p_j|) \quad (i \neq j; i, j = 1,2, \dots, n) \tag{1}$$

Where $|p_i| \cap |p_j|$ characterizes the times that commodity i and commodity j are both purchased by common customers (supposing $\alpha = |p_i| \cap |p_j|$). $|p_i| \cup |p_j|$ characterizes the sum of times that clients purchase either commodity i or commodity j (supposing $\beta = |p_i| \cap |p_j|$). More details in Table 1. $S(p_i, p_j)$ is more precise to indicate the similarity between two commodities i and j than merely using the amount of common customers to be the measurement of similarity. Because β in this equation can improve the precision of similarity. The same α doesn't stand for the same similarity. $S(p_i, p_j)$ gets larger while β begins to decline on the premise of the same α. More specifically, when the amount that commodity i and commodity j are both purchased by common customers, accounting for the greater proportion in the total amount of customers' purchase, commodity i and commodity j will possess higher similarity and be more probable to be clustered together. As shown in Table 2

Table 1: An example of Commodity Property Table

Commodity	1	2	...	j	...	m	*Sum*
$\|p_1\|$	$\|p_{11}\|$	$\|p_{12}\|$	...	$\|p_{1j}\|$	...	$\|p_{1m}\|$	
$\|p_2\|$	$\|p_{21}\|$	$\|p_{22}\|$	...	$\|p_{2j}\|$	...	$\|p_{2m}\|$	
$\|p_{1i}\| \cap \|p_{2i}\|$	$\|p_{11}\| \cap \|p_{21}\|$	$\|p_{12}\| \cap \|p_{22}\|$	...	...	...	$\|p_{1m}\| \cap \|p_{2m}\|$	$\|p_1\| \cap \|p_2\|$
$\|p_{1i}\| \cup \|p_{2i}\|$	$\|p_{11}\| \cup \|p_{21}\|$	$\|p_{12}\| \cup \|p_{22}\|$	...	...	...	$\|p_{1m}\| \cup \|p_{2m}\|$	$\|p_1\| \cup \|p_2\|$

Table 2: The characteristics of commodities

$\|p_i\| \cap \|p_j\|$ $\|p_i\| \cup \|p_j\|$ $S(p_i, p_j)$ w_{ij}	p_1	p_2	p_3	...	p_n
p_1		$\|p_1\| \cap \|p_2\|$	$\|p_1\| \cap \|p_3\|$	...	$\|p_1\| \cap \|p_n\|$
		$\|p_1\| \cup \|p_2\|$	$\|p_1\| \cup \|p_3\|$		$\|p_1\| \cup \|p_n\|$
	$S(p_1, p_1)$	$S(p_1, p_2)$	$S(p_1, p_3)$		$S(p_1, p_n)$
	w_{11}	w_{12}	w_{13}		w_{1n}
p_2	$\|p_2\| \cap \|p_1\|$		$\|p_2\| \cap \|p_3\|$	...	$\|p_2\| \cap \|p_n\|$
	$\|p_2\| \cup \|p_1\|$		$\|p_2\| \cup \|p_3\|$		$\|p_2\| \cup \|p_n\|$
	$S(p_2, p_1)$	$S(p_2, p_2)$	$S(p_2, p_3)$		$S(p_2, p_n)$
	w_{21}	w_{22}	w_{22}		w_{2n}
p_3	$\|p_3\| \cap \|p_1\|$	$\|p_3\| \cap \|p_2\|$		...	$\|p_2\| \cap \|p_n\|$
	$\|p_3\| \cup \|p_1\|$	$\|p_3\| \cup \|p_2\|$			$\|p_2\| \cup \|p_n\|$
	$S(p_3, p_1)$	$S(p_3, p_2)$	$S(p_3, p_3)$		$S(p_2, p_n)$
	w_{31}	w_{32}	w_{33}		w_{3n}
...	...	...	...	...	...
p_n	$\|p_n\| \cap \|p_1\|$	$\|p_n\| \cap \|p_2\|$	$\|p_n\| \cap \|p_3\|$	...	
	$\|p_n\| \cup \|p_1\|$	$\|p_n\| \cup \|p_2\|$	$\|p_n\| \cup \|p_3\|$		
	$S(p_n, p_1)$	$S(p_n, p_2)$	$S(p_n, p_3)$		$S(p_n, p_n)$
	w_{n1}	w_{n2}	w_{n3}		w_{nn}

With respect to DBSCAN clustering algorithm, the main characteristic of the clusters is that within each cluster we have a typical density of points which is considerably higher than outside of the cluster. Furthermore, the density within the areas of noise is lower than the density in any of the clusters. As we assume objects as points in a 2D scheme, the key idea of DBSCAN is that for each point of a cluster the neighbourhood of a given radius has to contain at least a minimum number of points. And the neighbourhood is determined by the choice of a distance function for two points p and q, denoted by $dist(p,q)$. To find a cluster, DBSCAN starts with an arbitrary point p and retrieves all points density-reachable from p with respect to ε and Min. If p is a core object, this procedure yields a cluster with respect to ε and Min. If p is a border object, no objects are density-reachable from p and p is assigned to noise temporarily. Then DBSCAN handles the next point.

In consideration of the clustering process, a commodity recommendation system can be regarde--d as a weighted and undirected graph $G = (V, E)(v_i \subseteq V, e_{i,j} \subseteq E, 1 \le i \le n, 1 \le j \le m)$,with commodities and correlation being as nodes and

edges, and the dissimilarity between commodity i and j is viewed as the weight w_{ij} relative to edge e_{ij}. The graph G can be a complete graph as we assume each commodity can be purchased with other commodities by common customers. The value of $S(p_i, p_j)$ expresses the similarities between two commodities, so it is inversely proportional to w_{ij}. Since higher similarities between commodities indicate lower dissimilarities between commodities which reflect shorter distance between nodes in certain graph $G = (V, E)$, we define w_{ij} as the reciprocal value of $S(p_i, p_j)$.

$$w_{ij} = 1/S(p_i, p_j) \quad (2)$$

When $S(p_i, p_j)$ is equivalent to 0, w_{ij} is confined to infinity labelled as ∞. While $S(p_i, p_j)$ equals to 1, $w_{ij} = 1$ which is the minimum value of the weights of edges between different commodities. Particularly, when i is equivalent to j, we presume $w_{ij} = 0$, as p_i and p_j are the same one commodity.

According to the conversion above and the characteristics of the P system with active membranes, we use DBSCAN clustering algorithm to divide the commodities in e-commerce websites into several clusters and realize commodity recommendation in e-commerce websites. Therefore online commodities grouped in the same cluster will be recommended to each other by system themselves. This paper specifies the procedure of solving this problem. And it is as follows.

Inputs: ε, *Min* and Commodity Property Table of n commodities.

Outputs: the number of clusters, the commodities in each cluster.

Procedure:

1. Calculate the similarities between all commodities by equation (1);
2. Compare the ε -neighborhood of each commodity with *Min*;
3. Determine if each commodity is a core object. If it is, then go to step *3*; if not, it is defined to be a noise;
4. Create new clusters with core objects;
5. Collect directly density-reachable objects from one core object and determine if the directly density-reachable object is core object. If it is, combine the two clusters into one;
6. Repeat step *4* until there is no change in the clusters.

3 P Systems with Active Membranes for Commodity Recommendation

3.1 P Systems with Active Membranes

Firstly it is necessary to introduce some basic prerequisites of the P systems. The P system is a distributed and parallel model. For a P system, there are three main ingredients: membrane structure, multisets of objects and evolution rules. A membrane structure consists of several membranes arranged hierarchically inside a main membrane, which is called skin membrane. A membrane is an elementary membrane if there are no membranes in it or it is a non-elementary membrane. Regions defined by a membrane structure can contain objects, corresponding to chemical substances present in the compartments of a cell. These objects can be described by symbols or by strings of symbols, in such a way that multisets of objects are placed in the regions of the membrane structure. The rules can process both objects and membranes in a non-characteristic and parallel way, and then the configuration of P system is changed. The basic membrane structure is shown in Fig.1.

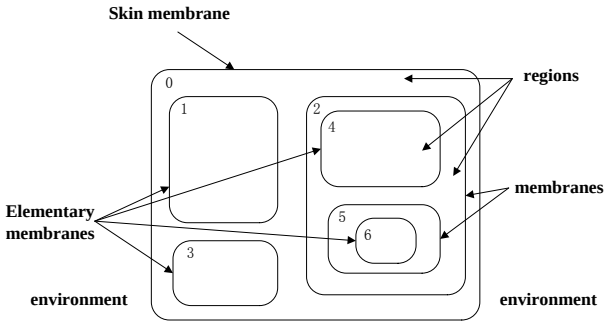

Fig.1 The basic membrane structure

From a biological point of view, a possible weakness of general cell-like P system is that the membrane structure is static and does not evolve during the computation. For this reason, the P system with active membranes was introduced in [4, 10], having rules which directly involve the membranes where the objects evolve and also making the membranes themselves evolve. Using the possibility to change membrane structure, it can create an exponential working space in linear time, which can then be used in a parallel computation for solving computationally hard problems. In the literature, this new model of P system has been successfully used to design solutions to some well-known NP-complete problems [11-14], and the Common Algorithmic Problem [15]. Recently, it has also been proposed to solve clustering problems, as

its characteristics can improve the efficiency of clustering process [9, 16].

In this paper, A P system with active membranes (and electrical charges) is a construct:

$$\Pi = (O, T, H, E, \mu, w_1, \ldots, w_n, R) \qquad (3)$$

Where:

- $n \geq 1$ is the initial degree of the system;
- O is the alphabet of objects;
- T is the output of the P system;
- H is a finite set of labels for membranes, $H = \{1,2,\cdots,n\}$;
- E is the set of electrical charges, $E = \{+,-,0\}$;
- μ is a membrane structure, consisting of n membranes, labelled (not necessarily in a one-to-one manner) with elements of H;
- w_i describes the multisets of objects in membrane i;
- R is finite set of developmental rules with the following forms:

I. $[a \to b]_h^e$,
where $h \in H, e \in E, a \in O, b \in O^*$
(Object evolution rules. An object a is evolved into b in a membrane h with the electric charge e, and the electric charge and the membranes are not modified during the process).

II. $a[\,]_h^{e_1} \to [b]_h^{e_2}$,
where $h \in H, e_1, e_2 \in E, a, b \in O$
(*Send-in* communication rules. An object a is introduced into the membrane h with the electric charge e_1, possibly modified to b during the process; and the electric charge of the membrane can be modified, but not the label of membrane).

III. $[a]_h^{e_1} \to [\,]_h^{e_2} b$,
where $h \in H, e_1, e_2 \in E, a, b \in O$
(*Send-out* communication rules. An object a is sent out of the membrane h with the electric charge e_1, possibly modified to b during the process; and the electric charge of the membrane can be modified, but not the label of membrane).

IV. $[a]_h^e \to b$,
where $h \in H, e \in E, a, b \in O$
(Dissolving rules. An object a is evolved into b with the electric charge e of membrane h, at the same time the surrounding membrane is dissolved. And the remaining objects in the former membranes are left free in the region immediately above it).

V. $[a]_h^{e_1} \to [b]_h^{e_1}[c]_h^{e_1}$,
where $h \in H, e_1, e_2, e_3 \in E, a, b, c \in O$
(Division rules for elementary membranes. The membrane h is divided into two membranes with possibly different labels and different electric charges; the object a specified in the rule is replaced by possibly new objects b, c respectively in the two new membranes; and the remaining objects are duplicated in the process).

VI. $[\,]_{h_1}^{e_1}[\,]_{h_2}^{e_1} \to [\,]_{h_3}^{e_1}$,
where $h_1, h_2, h_3 \in H, e_1, e_2, e_3 \in E$
(Merging rules for elementary membranes. The membranes h_1, h_2 are merged into a single membrane h_3 with possibly different electric charges; the objects of the former membranes are put together in the new membrane).

VII. $[a]_{h_1}^{e_1}[b]_{h_2}^{e_1} \to [c]_{h_3}^{e_1}$,
where $h_1, h_2, h_3 \in H, e_1, e_2, e_3 \in E$,
$a, b, c \in O$
(Fusion rule for elementary membranes. With the object a in membrane h_1 and object b in membrane h_2, the two membranes are merged into a single membrane h_3 with possibly new objects c; and the remaining objects of the former membranes are put together in the new membrane).

These rules are applied according to the following principles [17]:

1. All rules are applied in parallel: Any object and membrane (including all copies) which can evolve by a rule of any form should evolve; an object or a membrane can be used by only one rule when there are priority relations among rules, else it is chosen non-deterministically.
2. All objects and membranes not specified in a rule and which do not evolve are passed unchanged to the next step.
3. Before a rule of type (IV), (V), (VI) or (VII) is applied to a membrane, the rules of type (I) are applied first to its objects and then the resulting objects are further copied or moved in accordance with rule of type (IV), (V), (VI) or (VII).

4. Note the difference between merging rules and fusion rules. In a merging rule, all membranes corresponding to the polarizati--on condition of the rules can be used by this rule. In contrast, in a fusion rule, only when there are objects a and b respectively in membrane h_1 and h_2 and two membranes correspond to the polarization condition of the rules, fusion rules can be executed.
5. All rules are applied during one step in a bottom-up manner: first, they are applied to elementary membranes, then to their parent membranes and up towards the skin membrane.
6. The rules of type (IV), (V), (VI) or (VII) do not apply to the skin membrane.

The membrane structure of the P system at a given time, together with all multisets of objects associated with the regions of this membrane structure is the configuration of the system at that time. The $(\mu, w_1, \ldots, w_n)$ is the initial configuration. We can pass from a configuration to another one by using the rules from R according to the principles given above. We say that we have a transition among configurations. A sequence of transitions which starts from the initial configuration is called a computation with respect to Π. A computation is complete if it cannot be continued: there is no rule which can be applied to objects and membranes in the last configuration. The result of the computation is the collection of objects expelled from the output membrane during the whole computation. Only halting computations give a result, non-halting computations give no output.

3.2 P System for Solving Commodity Recommendation problems

In this part, a P System with active membranes for solving commodity recommendation problems in e-commerce websites is proposed. The configurations and structures at each step of the computation in this new P system are shown in Fig. 2.

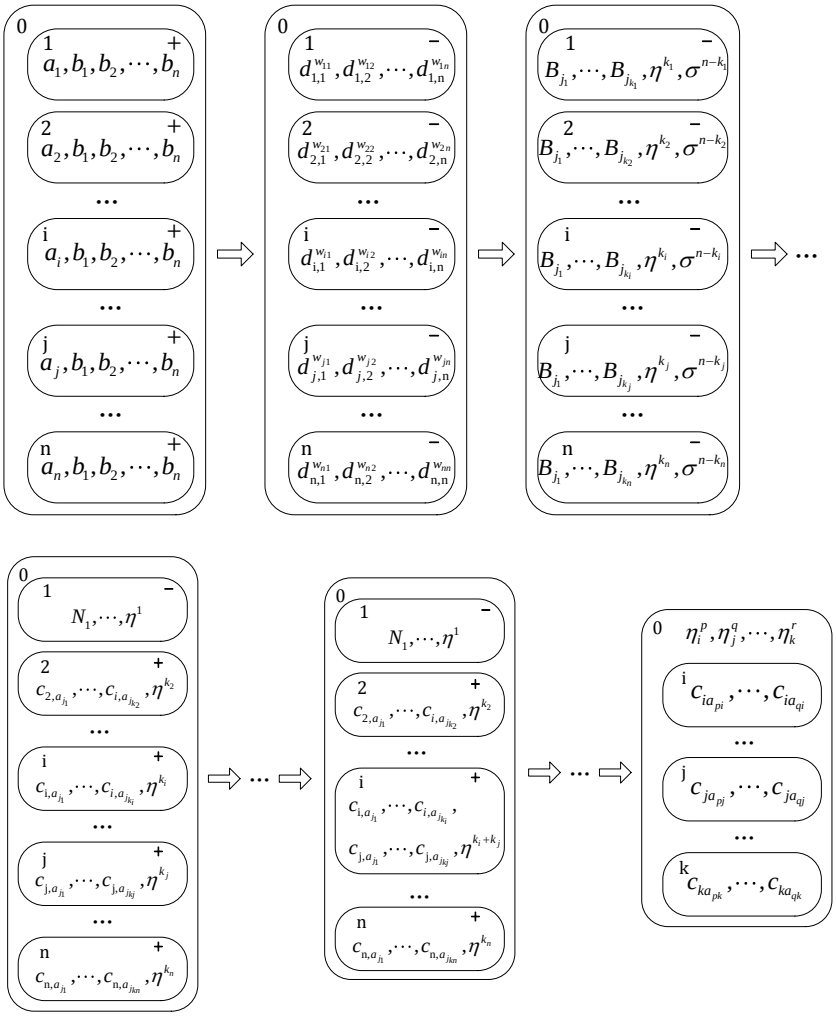

Fig.2 The configurations at each step of the computation

In this P system, there are $n+1$ membranes. Membrane 0 is the skin membrane which is regarded as output membrane as well. Membrane 1 to n respectively represents n commodities in the initial. In the process of solving commodity recommendation problems, not only the objects evolve but also the membrane structure. Through the rules of active membranes, the k membranes obtained finally indicate k clusters of commodities. Then the commodity recommendation mechanism will works on the basis of the clustering results. When buyers purchase one commodity in e-commerce websites, it will recommend other commodities in certain cluster that the purchased commodity belongs to.

Significantly, before the computation in P system, it is imperative to compare the dissimilarity between commodities. As a matter of convenience, we express the dissimilarity in the form of matrix D_{nn} and all w_{ij} are rounded to nonnegative integer variables.

The P system with active membranes for solving commodity recommendation problems is defined as follows:

$\Pi = (O, T, \mu, \omega_0, \omega_1, \dots, \omega_n, R_0, R_1, \dots, R_n, \rho)$ (4)

Where:

- Working alphabet:
  $O = \{a_i, b_j, d_{i,j}, \gamma_i, \sigma, B_j, C_{pa_i}, N_i\}$
- Output:
  $T = \{\gamma_i^k\},\ 1 \le i, k \le n$
- Membrane structure:
  $\mu = [\,[\,]_1^+[\,]_2^+ \dots [\,]_{n-1}^+[\,]_n^+]_0$
- Initial multisets:
  $\omega_0 = \{\lambda\}$; ω_0 represents the initial set of objects in skin membrane;
  $\omega_i = \{a_i, b_1, b_2, \cdots, b_n\colon\ 1 \le i \le n\}$;
- The set R of evolution rules consists of the following rules:
  Rules in the membrane labelled $i\,\{1 \le i \le n\}$:
  $r_1 = \{[a_i b_1 b_2 \cdots b_n]_i^+ \to [d_{i,1}^{w_{i1}} d_{i,2}^{w_{i2}} \cdots d_{i,n}^{w_{in}}]_i^- | 1 \le i \le n\}$
  $r_2 = \{[d_{i,j}^t]_i^- \to [\gamma_i B_j]_i^- | 1 \le i, j \le n, 0 \le t \le \varepsilon\} \cup \{[d_{i,j}^t]_i^- \to [\sigma]_i^- | 1 \le i, j \le n, t > \varepsilon\}$
  $r_3 = \{[\gamma_i^k]_i^- \to [\gamma_i^k]_i^+ | Min \le k \le \mathrm{n}, 1 \le i, j \le n\ \} \cup \{[\gamma_i^k]_i^- \to [N_i \gamma_i^1]_i^-\ | 0 < k < Min, 1 \le i, j \le n\}$
  $r_4 = \{[B_j]_i^+ \to [C_{i,a_j}]_i^+ | 1 \le i, j \le n\}$
  $r_5 = \{[\sigma]^+ \to [\lambda]^+\} \cup \{[\sigma]^- \to [\lambda]^-\} \cup \{[B_j]^- \to [\lambda]^- | 1 \le j \le n\}$
  $r_6 = \{[C_{ia_j}\gamma_i^k]_i^+ [C_{ja_i} C_{ja_p} \gamma_j]_j^+ \to [C_{ia_j}\gamma_i^{k+1}]_i^+ [C_{ja_i} C_{ia_p}]_j^- | 1 \le i < j \le n,\ 1 \le p, k \le n,\ i \ne p\ \}$
  $r_7 = \{([\,]_j^-[\,]_i^+ \to [\,]_i^+) \cap ([C_{qa_j}]_q^+ \to [C_{qa_i}]_q^+) | 1 \le i < j \le n, 1 \le q \le n\}$
  $r_8 = \{([C_{ia_p} C_{ia_p} \gamma_i^k]_i^+ \to [C_{ia_p}\gamma_i^{k-1}]_i^+) \cup ([C_{ja_i}\gamma_j]_i^+ \to [\lambda]_i^+) | 1 \le i, j, p, k \le n\ \}$
  $r_9 = \{[C_{ia_p}]_i^+ [N_p \gamma_p^k]_p^- \to [C_{ia_p}]_i^+ | 1 \le i, p, k \le n\ \}$
  $r_{10} = \{([\,]_i^+ \to [\,]_i) \cap ([\,]_i^- \to [\,]_i) | 1 \le i \le n\ \}$
  $r_{11} = \{[\gamma_i^k]_i \to \gamma_i^k [\,]_i | 1 \le i, k \le n\ \}$
- The priority relations ρ over $R_i\,\{1 \le i \le n\}$: $\rho = \{r_i > r_{i+1}\}$

3.3 The Computations in P System

This part will give a comprehensive description of the P system we designed for solving commodity recommendation problems based on DBSCAN clusters algorithm. Firstly, we calculate the dissimilarities (defining as w_{ij}) between all commodities based on the Commodity Property Table and equation (2).

At the beginning of a computation, the membrane labelled $i\,\{1 \le i \le n\}$ with positive electric charge contains objects $a_i, b_1, \cdots, b_n$. The object a_i represents the i-th commodity of the dataset and $b_1, \cdots, b_n$ represents all commodities in dataset. Initially in the P system the only rules that can be applied is r_1 [18] in membrane labelled i. It modifies the polarization of membrane i and produces the object $d_{i,j}^{w_{ij}}$ $(1 \le i, j \le n)$ which is used to compare the distances between a_i and other objects in dataset. Then γ_i and σ is obtained according to the distance $d_{i,j}^{w_{ij}}$. The object γ_i indicates that the distance between a_i and b_j is less than ε. The object σ indicates that the distance between a_i and b_j is more than ε. If the number of γ_i is more than minimum number *Min*, we call a_i a core point and mark its directly density-reachable

members as C_{ia_j} with the positive change of membrane polarization. Or this point is defined to be a noise and it is marked as N_i without the change of membrane polarization. At the same time, the multiplicity of the object γ_i represents the number of density–reachable objects for the core object a_i.

After comparison stage, the rules for aggregating objects are triggered. The object evolution rule r_6 is executed extremely to find all directly density-reachable objects of the core object. In positive membrane, when there is C_{ia_j} in the membrane i and C_{ja_i} in the membrane j as well, the rule r_6 is activated to transform C_{ja_p}into C_{ia_p} and increase the multiplicity of the object γ_i with the multiplicity of the object γ_j declined. Concurrently, the polarization of membrane j is change to negative polarity. Then the two membranes i and j with different polarization are merged into a single membrane i, and the objects of the former membranes are put together in the membrane i and the object C_{qa_j} in membrane q is transformed by C_{qa_i} to make the core object a_i continue to collect density-reachable objects. However, if a noise is directly density-reachable from a core object, it also be gathered into the membrane of the core object. Meanwhile, the rule r_8 remove the same object and the redundant object in the new membrane and decrease the multiplicity of the object γ_i. This indicates the combination of the similar object, which is the process of DBSCAN clustering.

Finally, the polarity of all membranes in skin membrane is changed to neutral and the *send-out* communication rule r_{10} is carried out to send the object γ_i to the skin membrane. Each index i indicates one cluster and the multiplicity of the object γ_i indicate the number of members in each cluster. Particularly, when the multiplicity of the object γ_p is 1, it means a_p is a noise in this dataset.

These rules are used maximum parallel in each membrane when calculating. This P system will halt if no more rules can be executed and no more objects γ_i can be obtained. And at this moment, the configuration of each membrane is achieved to be stable. As a consequence, each membrane in the skin membrane represents one group, and the objects in each membrane represent the commodities in each group. After the process of clustering all commodities, the commodity recommendation mechanism in e-commerce website will operate successfully.

In this paper, we analyze the computation complexity based on the operation times. In order to obtain the final result, we need $\log n$ loops. In every loop, the maximum execution times of each rule are listed in Table 3. After analyzing the rules, the computation complexity of DBSCAN clustering algorithm in this P system with active membranes is $O(n \log n)$, while the computation complexity of the original DBSCAN clustering algorithm is $O(n^2)$ without spatial query. As a consequence, this P system can reduce the computation complexity of clustering process and improve the efficiency to solve the problems of commodity recommendation.

Table 3: The execution analysis of each rule

r_1	r_2	r_3	r_4	r_5	r_6
$O(1)$	$O(n)$	$O(1)$	$O(n)$	$O(1)$	$O(n/2)$
r_7	r_8	r_9	r_{10}	r_{11}	
$O(n/2)$	$O(n/2)$	$O(1)$	$O(1)$	$O(1)$	

4 Test and Analysis

In order to illustrate how this P system with active membranes works for solving commodity recommendation problems based on DBSCAN clusters algorithm, we propose a real-world case to verify the feasibility and effectiveness of recommending commodities to customers. Table 4 points out 15 records in a month of different commodities in the region of Digital Products on AMAZON.CN, a famous e-commerce website. As we got the value of $|p_i| \cap |p_j|$ and $|p_i| \cup |p_j|$ between commodities from historical data on AMAZON.CN, $S(p_i, p_j)$ and w_{ij} are calculated by Eq. (1) and Eq. (2). To facilitate the computation, there are three characteristics in the table in proper order: $|p_i| \cap |p_j|$, $|p_i| \cup |p_j|$ and w_{ij}. For the convenience of calculation in P system, as we round the distances to integer, then the distance matrix of these 15 commodities can be obtained as $D_{15,15}$. And in this real-world case, we suppose that $\varepsilon = 4$, *Min*=3. The initial structure of this P system is shown in Fig.3.

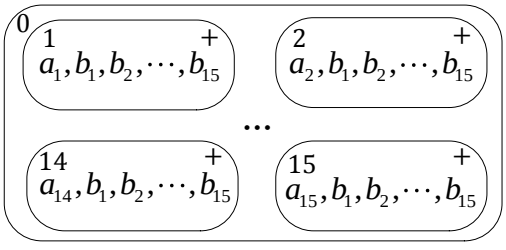

Fig.3 The initial structure of this P system

Table 4: Three characteristics of commodities

$\frac{\lvert p_i\rvert \cap \lvert p_j\rvert}{\lvert p_i\rvert \cup \lvert p_j\rvert}$ w_{ij}	1	2	3	4	5	…	13	14	15
p_1		378	162	94	70	…	40	40	23
		756	649	566	488		711	800	452
	0	2	4.006	6.021	6.971		17.775	20	19.652
p_2	378		490	224	164	…	34	30	30
	756		979	896	818		543	543	543
	2	0	1.998	4	4.988		15.971	18.1	18.1
p_3	162	490		395	237	…	67	64	42
	649	979		789	711		934	1023	675
	4.006	1.998	0	1.997	3		13.94	15.984	16.071
p_4	94	224	395		628	…	71	67	42
	566	896	789		628		851	940	592
	6.021	4	1.997	0	1		11.986	14.03	14.095
p_5	70	164	237	628		…	70	66	40
	488	818	711	628			773	862	514
	6.971	4.988	3	1	0		11.043	13.061	12.85
…	…	…	…	…	…	…	…	…	…
p_{13}	40	34	67	71	70	…		543	369
	711	543	934	851	773			1085	737
	17.775	15.971	13.94	11.986	11.043		0	1.998	1.997
p_{14}	40	30	64	67	66	…	543		413
	800	543	1023	940	862		1085		826
	20	18.1	15.984	14.03	13.061		1.998	0	2
p_{15}	23	30	42	42	40	…	369	413	
	452	543	675	592	514		737	826	
	19.652	18.1	16.071	14.095	12.85		1.997	2	0

$$D_{15,15} = \begin{pmatrix} 0 & 2 & 4 & 6 & 7 & 6 & 9 & 8 & 11 & 16 & 18 & 20 & 18 & 20 & 20 \\ 2 & 0 & 2 & 4 & 5 & 4 & 7 & 6 & 9 & 14 & 16 & 18 & 16 & 18 & 18 \\ 4 & 2 & 0 & 2 & 3 & 2 & 5 & 4 & 7 & 12 & 14 & 16 & 14 & 16 & 16 \\ 6 & 4 & 2 & 0 & 1 & 4 & 3 & 6 & 5 & 10 & 12 & 14 & 12 & 14 & 14 \\ 7 & 5 & 3 & 1 & 0 & 3 & 2 & 5 & 4 & 9 & 11 & 13 & 11 & 13 & 13 \\ 6 & 4 & 2 & 4 & 3 & 0 & 3 & 2 & 5 & 10 & 12 & 14 & 12 & 14 & 14 \\ 9 & 7 & 5 & 3 & 2 & 3 & 0 & 3 & 2 & 7 & 9 & 11 & 9 & 11 & 11 \\ 8 & 6 & 4 & 6 & 5 & 2 & 3 & 0 & 3 & 8 & 10 & 12 & 10 & 12 & 12 \\ 11 & 9 & 7 & 5 & 4 & 5 & 2 & 3 & 0 & 5 & 7 & 9 & 7 & 9 & 9 \\ 16 & 14 & 12 & 10 & 9 & 10 & 7 & 8 & 5 & 0 & 2 & 4 & 2 & 4 & 4 \\ 18 & 16 & 14 & 12 & 11 & 12 & 9 & 10 & 7 & 2 & 0 & 2 & 2 & 2 & 2 \\ 20 & 18 & 16 & 14 & 13 & 14 & 11 & 12 & 9 & 4 & 2 & 0 & 4 & 2 & 4 \\ 18 & 16 & 14 & 12 & 11 & 12 & 9 & 10 & 7 & 2 & 2 & 4 & 0 & 2 & 2 \\ 20 & 18 & 16 & 14 & 13 & 14 & 11 & 12 & 9 & 4 & 2 & 2 & 2 & 0 & 2 \\ 20 & 18 & 16 & 14 & 13 & 14 & 11 & 12 & 9 & 4 & 2 & 4 & 2 & 2 & 0 \end{pmatrix}$$

Operating in P system with membranes, we can obtain the computational process which is shown in appendix (Table 5-Table 8). Particularly, some repeated steps in the process are omitted. In the process, Table 5 shows the dissimilarities between 15 commodities and then indicates to determine if each object is core object. Table 6 and Table 7 respectively reflect the first and the second circulation of membrane mergence between similar commodities. In the beginning, $p_1$and p_2, p_3 and p_4, $p_5$and p_6, $p_7$and p_8, $p_{10}$and p_{11}, $p_{12}$and p_{13}, $p_{14}$and p_{15} are separately grouped together and then are p_1 and p_3, p_5 and p_7, $p_{10}$and p_{12}. The computation process continues until no membranes can be merged together. Finally, and the clustering of these *15* commodities in this P system is finished. The membrane structure and objects of this P system are changed by the rules of active membranes which are shown in Fig.4 and Table 8. According to the final result, the objects γ_{10}^6, γ_{10}^6 and two membranes remaining in skin membrane indicate that two clusters are produced in the end, and there are 9 objects in the first cluster and 6 objects in the second. The objects in each membrane reveal the members in each cluster. Then the commodities in two groups are respectively $p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9$ and $p_{10}, p_{11}, p_{10}, p_{10}, p_{10}, p_{10}$.

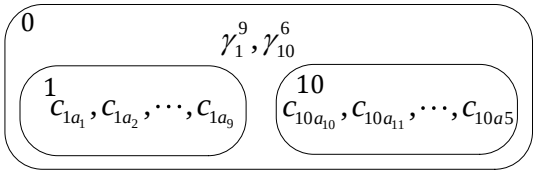

Fig.4 The final structure of this P system

With the assigned parameters of ε and *Min* in the DBSCAN algorithm, a simulation experiment is conducted by using Matlab R2010a. After working with 15 product records, the clustering distribution is shown in Fig.5. Compared the result drawn from the P system with the clusters obtained by simulation experiment, the product clustering in this recommendation system is feasible and accurate.

In this paper, the commodity recommendation mechanism works on the basis of the clustering results. Before recommendation system works, the commodities in database are clustered according to the density-based clustering method by P system. For the first group in commodity recommendation system, commodity 1 and 2 are both purchased by common customers and commodity 2 and 4 are both purchased by common customers, then the commodity 1 and 4 is grouped together owing to commodity 2 with the existence of C_{1a_2}, C_{2a_1}, C_{2a_4}, C_{4a_2}. The same operations also happen in other commodities and then all commodities are divided into different groups. When buyers select one commodity to purchase in e-commerce websites, the commodity recommendation mechanism will recommend other commodities in certain cluster that the purchased commodity belongs to.

Currently, content-based recommendation (CBR) and collaborative filtering recommendation (CFR) are two traditional and mature recommendation methods in e-commerce websites. And data mining techniques, particularly clustering methods, are frequently used as specific techniques. However, with the increase of online customers and online products, the traditional recommendation technique encounters the bottleneck states in speed. As a distributed and parallel model, P system can be used in a parallel computation to solve this problem successfully. By means of P system with active membranes in this recommendation system, the computation complexity of DBSCAN clustering algorithm in commodity recommendation is reduced to $O(n \log n)$, which is more efficient than original method with the growth of n.

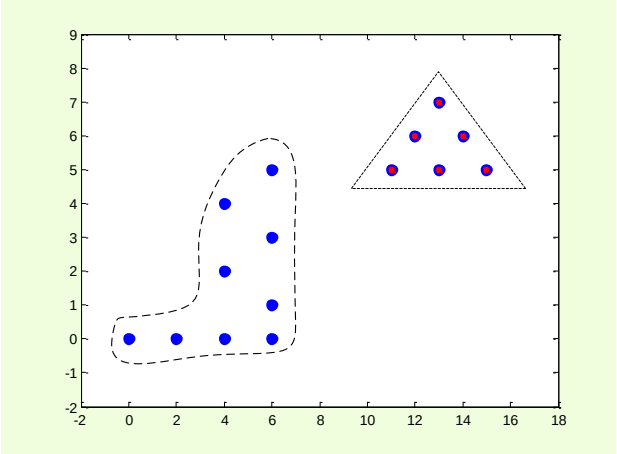

Fig.5 The result of clustering problems

5 Conclusion

The main feature of membrane computing is great parallelism. The parallel execution of the rules and the characteristics of changing membrane structure in P system with active membranes are considered to be more suitable to solve clustering problems. Due to the problems traditional recommendation technique encountered in speed, this paper proposes

a P system with active membranes to solve commodity recommendation problem based on DBSCAN clustering algorithm. The commodity recommendation method in this paper works on the basis of the clustering results. When buyers have purchased one commodity in e-commerce websites, it will recommend other commodities in certain cluster that the purchased commodity belongs to. Therefore the efficiency of clustering process is the key point for recommendation techniques. In this paper, The computation complexity of DBSCAN clustering algorithm is kept to $O(n \log n)$, which can solve the commodity recommendation problems at a higher rate of speed.

Recently, membrane computing develops rapidly, but most work lies in the theoretical stage. In this paper, we combine membrane computing techniques with clustering algorithms to realize practical problems. Due to the feasibility and effectiveness of this system, it has a great significance on improving the efficiency to recommend more suitable commo--dities to customers in the e-commerce websites. However, there is some weakness in this work without larger database of records to be clustered by this system. In our future study, it is prospective to apply membrane computing techniques to realize more clustering algorithms and solve more practical issues with large database.

6 Acknowledgment

This work was supported by the Natural Science Foundation of China (No.61170038), Natural Science Foundation of Shandong Province, China (No.ZR2011FM001), Humani--ties and Social Sciences Project of Ministry of Education, China (No.12YJA630152), Social Science Fund of Shandong Province, China (No.11CGLJ22), Science-Technology Program of the Higher Education Institutions of Shandong Province, China (No.J12LN22).

References:

[1] M. J. Li, Study on E-Commerce Recommenda--tion System Based on Web Mining, M.E. dissertation, Dept. Economics and Management, Beijing Jiaotong Univ., Beijing, China, 2008.

[2] S. Zeng, Trust Model of Social Network-based recommendation system of commodity, M.E. dissertation, Dept. Computer Science and Engineering, South China of Technology Univ., Guangzhou, China, 2012.

[3] G. Paun, A quick introduction to membrane computing, The Journal of Logic and Algebraic Programming, Vol. 79, 2010, pp. 291-294.

[4] G. Paun, G. Rozenberg and A. Salomaa. *Membrane Computing*. New York: oxford university press, 2010, pp. 282-301.

[5] X. M. Hu, Research on Recommender System Based on Product Attributes, Ph.D. dissertation, Dept. Science and Technology, Huazhong Univ., Hubei, China, 2012.

[6] J. Han and M. Kambr, *Data Mining Concepts and Techniques*. USA: Elsevier Inc., 2012, ch.8.

[7] D. L. Xing, M. H. Zhao and W. C. Chen, A Destiny-Based DBSCAN Algorithm, Science paper Online, Aug.2007.

[8] H. Y. Zhang and X. Y. Liu, A CLIQUE algorithm using DNA computing techniques based on closed-circle DNA sequences, Biosystems, Vol.105, No.1, 2011, pp. 73-82.

[9] M. Cardona, M. A. Colomer and M. J. Pérez-Jiménez, Hierarchical clustering with membrane computing, Computing and informatics, Vol. 27, No. 3, 2008, pp. 497-513.

[10] G. Paun, Computing with membranes: Attacking NP–complete problems, Unconvent--ional Models of Computation, UMC'2K (I. Antoniou, C. Calude, M.J. Dinneen, eds.), Springer–Verlag, 2000, pp. 94–115.

[11] M. J. Pérez–Jiménez, A. Riscos-Nunez, Solving the Subset-Sum problem by active membranes, New Generation Computing, Vol. 23, No. 4, 2005, pp. 339-356.

[12] M. J. Pérez–Jiménez, A. Riscos-Nunez, A linear–time solution to the Knapsackproblem using P systems with active membranes, Membrane Computing (C. Martın-Vide, G. Paun, G. Rozenberg, A. Salomaa, eds.), Vol. 2933, 2004, pp. 250–268.

[13] M. J. Pérez–Jiménez, A. Romero-Jiménez, F. Sancho-Caparrini, A polynomial complexity class in P systems using membrane division, Proceedings of the 5th Workshop on Descriptional Complexity of Formal Systems, 2003, pp. 284-294.

[14] M. A. Gutierrez-Naranjo, M. J. Pérez–Jiménez, A. Riscos-Nunez, A fast P system for finding a balanced 2-partition, Soft Computing, Vol. 9, No.9, 2005, pp. 673–678.

[15] M. J. Pérez–Jiménez, F. J. Romero–Campero, Attacking the Common Algorithmic Problem by recognizer P systems, Machines, Computations and Universality, MCU'2004, Saint Petersburg, Russia, September 21-24, 2004, Revised Selected Papers, LNCS 2005, Vol.3354, pp. 304-315.

[16] J. Xue, X. Y. Liu, Spatial Cluster by Communication P System on Trees, Journal of Information and Computational Science, Vol. 9, No.13, 2012, pp. 3883-3893.
[17] Petr Sos ́k, Alfonso Rodr ́guez-Pat ́n, P Systems with Active Membranes Characterize PSPACE, DNA Computing, Vol. 4287, 2006, pp. 33-46.
[18] Y. Z. Zhao, X. Y. Liu and J. H. Qu, The K-Medoids Clustering Algorithm by a Class of P System, Journal of Information and Computational Science, Vol. 9, No. 18, 2012, pp. 5777-5790.

Appendix :

Table 5: the computational process in P system

	Step0(initial state)	*Step1*	*Step2*
Mem0	λ	λ	λ
Mem1	$+, a_1, b_1, b_2, \cdots, b_{14}, b_{15}$	$-, d^0_{1,1}, d^2_{1,2}, \cdots, d^{20}_{1,14}, d^{20}_{1,15}(r_1)$	$-, B_1, B_2, B_3, \gamma^3_1, \sigma^{12}(r_2)$
Mem2	$+, a_2, b_1, b_2, \cdots, b_{14}, b_{15}$	$-, d^5_{2,1}, d^0_{2,2}, \cdots, d^{18}_{2,14}, d^{18}_{2,15}(r_1)$	$-, B_1, B_2, B_3, B_4, B_6, \gamma^5_2, \sigma^{10}(r_2)$
Mem3	$+, a_3, b_1, b_2, \cdots, b_{14}, b_{15}$	$-, d^6_{3,1}, d^1_{3,2}, \cdots, d^{116}_{3,14}, d^{16}_{3,15}(r_1)$	$-, B_1, B_2, B_3, B_4, B_5, B_6, B_8, \gamma^7_3, \sigma^8(r_2)$
…	…	…	…
Mem14	$+, a_{14}, b_1, b_2, \cdots, b_{14}, b_{15}$	$-, d^{12}_{14,1}, d^{13}_{14,2}, \cdots, d^0_{14,14}, d^2_{14,15}(r_1)$	$-, B_{10}, B_{11}, B_{12}, B_{13}, B_{14}, B_{15}, \gamma^6_{14}, \sigma^9(r_2)$
Mem15	$+, a_{15}, b_1, b_2, \cdots, b_{14}, b_{15}$	$-, d^{16}_{15,1}, d^{11}_{15,2}, \cdots, d^2_{15,14}, d^0_{15,15}(r_1)$	$-, B_{10}, B_{11}, B_{12}, B_{13}, B_{14}, B_{15}, \gamma^6_{15}, \sigma^9(r_2)$

	Step3	*Step4*
Mem0	λ	λ
Mem1	$+, B_1, B_2, B_3, \gamma^3_1, \sigma^{12}(r_3)$	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, \gamma^3_1, \sigma^{12}(r_4)$
Mem2	$+, B_1, B_2, B_3, B_4, B_6, \gamma^5_2, \sigma^{10}(r_3)$	$+, C_{2a_1}, C_{2a_2}, C_{2a_3}, C_{2a_4}, C_{2a_6}, \gamma^5_2, \sigma^{10}(r_4)$
Mem3	$+, B_1, B_2, B_3, B_4, B_5, B_6, B_8, \gamma^7_3, \sigma^8(r_3)$	$+, C_{3a_1}, C_{3a_2}, C_{3a_3}, C_{3a_4}, C_{3a_5}, C_{3a_6}, C_{3a_8}, \gamma^7_3, \sigma^8(r_4)$
…	…	…
Mem14	$+, B_{10}, B_{11}, B_{12}, B_{13}, B_{14}, B_{15}, \gamma^6_{14}, \sigma^9(r_3)$	$+, C_{14a_{10}}, C_{14a_{11}}, C_{14a_{12}}, C_{14a_{13}}, C_{14a_{14}}, C_{14a_{15}}, \gamma^6_{14}, \sigma^9(r_4)$
Mem15	$+, B_{10}, B_{11}, B_{12}, B_{13}, B_{14}, B_{15}, \gamma^6_{15}, \sigma^9(r_3)$	$+, C_{15a_{10}}, C_{15a_{11}}, C_{15a_{12}}, C_{15a_{13}}, C_{15a_{14}}, C_{15a_{15}}, \gamma^6_{15}, \sigma^9(r_4)$

	Step5	*Step6*
Mem0	λ	λ
Mem1	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, \gamma^3_1(r_5)$	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, \gamma^7_1(r_6)$
Mem2	$+, C_{2a_1}, C_{2a_2}, C_{2a_3}, C_{2a_4}, C_{2a_6}, \gamma^5_2(r_5)$	$-, C_{2a_1}, C_{1a_2}, C_{1a_3}, C_{1a_4}, C_{1a_6}, \gamma^1_2(r_6)$
Mem3	$+, C_{3a_1}, C_{3a_2}, C_{3a_3}, C_{3a_4}, C_{3a_5}, C_{3a_6}, C_{3a_8}, \gamma^7_3(r_5)$	$+, C_{3a_1}, C_{3a_2}, C_{3a_3}, C_{3a_4}, C_{3a_5}, C_{3a_6}, C_{3a_8}, \gamma^{12}_3(r_6)$
…	…	…
Mem14	$+, C_{14a_{10}}, C_{14a_{11}}, C_{14a_{12}}, C_{14a_{13}}, C_{14a_{14}}, C_{14a_{15}}, \gamma^6_{14}(r_5)$	$+, C_{14a_{10}}, C_{14a_{11}}, C_{14a_{12}}, C_{14a_{13}}, C_{14a_{14}}, C_{14a_{15}}, \gamma^{11}_{14}(r_6)$
Mem15	$+, C_{15a_{10}}, C_{15a_{11}}, C_{15a_{12}}, C_{15a_{13}}, C_{15a_{14}}, C_{15a_{15}}, \gamma^6_{15}(r_5)$	$-, C_{14a_{10}}, C_{14a_{11}}, C_{14a_{12}}, C_{14a_{13}}, C_{15a_{14}}, C_{14a_{15}}, \gamma^1_{15}(r_6)$

Table 6: steps of the first circulation in P system

	Step7	⋯	*Step13*
Mem0	λ	⋯	λ
Mem1	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, \gamma_1^7, C_{2a_1}, C_{1a_2}, C_{1a_3}, C_{1a_4}, C_{1a_6}, \gamma_2^1(r_7)$	⋯	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, \gamma_1^5, C_{1a_4}, C_{1a_6}(r_8)$
Mem3	$+, C_{3a_1}, C_{3a_1}, C_{3a_3}, C_{3a_4}, C_{3a_5}, C_{3a_6}, C_{3a_8}, \gamma_3^{12}, C_{3a_1}, C_{4a_3}, C_{3a_4}, C_{3a_5}, C_{3a_6}, C_{3a_7}, \gamma_4^1(r_7)$	⋯	$+, C_{3a_1}, C_{3a_3}, C_{3a_4}, C_{3a_5}, C_{3a_6}, C_{3a_8}, \gamma_3^7, C_{3a_7}(r_8)$
Mem5	$+, C_{5a_3}, C_{5a_3}, C_{5a_5}, C_{5a_6}, C_{5a_7}, C_{5a_9}, \gamma_5^{12}, C_{5a_1}, C_{5a_3}, C_{5a_3}, C_{6a_5}, C_{5a_6}, C_{5a_7}, C_{5a_8}, \gamma_6^1(r_7)$	⋯	$+, C_{5a_3}, C_{5a_5}, C_{5a_6}, C_{5a_7}, C_{5a_9}, \gamma_5^7, C_{5a_1}, C_{5a_8}(r_8)$
Mem7	$+, C_{7a_3}, C_{7a_5}, C_{7a_5}, C_{7a_7}, C_{7a_8}, C_{7a_9}, \gamma_7^{10}, C_{7a_3}, C_{7a_5}, C_{8a_7}, C_{7a_8}, C_{7a_9}, \gamma_8^1(r_7)$	⋯	$+, C_{7a_3}, C_{7a_5}, C_{7a_7}, C_{7a_8}, C_{7a_9}, \gamma_7^5, (r_8)$
Mem9	$+, C_{9a_5}, C_{9a_7}, C_{9a_8}, C_{9a_9}, \gamma_9^4$	⋯	$+, C_{9a_5}, C_{9a_7}, C_{9a_8}, C_{9a_9}, \gamma_9^4$
Mem10	$+, C_{10a_{10}}, C_{10a_{11}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{10}^{11}, C_{11a_{10}}, C_{10a_{11}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{11}^1(r_7)$	⋯	$+, C_{10a_{10}}, C_{10a_{11}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{10}^6, (r_8)$
Mem12	$+, C_{12a_{10}}, C_{12a_{10}}, C_{12a_{12}}, C_{12a_{13}}, C_{12a_{14}}, C_{12a_{15}}, \gamma_{12}^{11}, C_{12a_{10}}, C_{12a_{10}}, C_{13a_{12}}, C_{12a_{13}}, C_{12a_{14}}, C_{12a_{15}}, \gamma_{13}^1(r_7)$	⋯	$+, C_{12a_{10}}, C_{12a_{12}}, C_{12a_{13}}, C_{12a_{14}}, C_{12a_{15}}, \gamma_{12}^5, (r_8)$
Mem14	$+, C_{14a_{10}}, C_{14a_{10}}, C_{14a_{12}}, C_{14a_{12}}, C_{14a_{14}}, C_{14a_{15}}, \gamma_{14}^{11}, C_{14a_{10}}, C_{14a_{10}}, C_{14a_{12}}, C_{14a_{12}}, C_{15a_{14}}, C_{14a_{15}}, \gamma_{15}^1(r_7)$	⋯	$+, C_{14a_{10}}, C_{14a_{12}}, C_{14a_{14}}, C_{14a_{15}}, \gamma_{14}^4, (r_8)$

Table 7: steps of the second circulation in P system

	Step14		*Step15*	⋯	*Step20*	⋯
Mem0	λ	*Mem0*	λ	⋯	λ	⋯
Mem1	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, \gamma_1^{11}, C_{1a_4}, C_{1a_6}(r_6)$	*Mem1*	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, \gamma_1^{11}, C_{1a_4}, C_{1a_6}, C_{3a_1}, C_{1a_3}, C_{1a_4}, C_{1a_5}, C_{1a_6}, C_{1a_8}, \gamma_3^1, C_{1a_7}(r_7)$	⋯	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, \gamma_1^8, C_{1a_4}, C_{1a_5}, C_{1a_6}, C_{1a_8}, C_{1a_7}(r_8)$	⋯
Mem3	$-, C_{3a_1}, C_{1a_3}, C_{1a_4}, C_{1a_5}, C_{1a_6}, C_{1a_8}, \gamma_3^1, C_{3a_7}(r_6)$					
Mem5	$+, C_{5a_3}, C_{5a_5}, C_{5a_6}, C_{5a_7}, C_{5a_9}, \gamma_5^{11}, C_{5a_1}, C_{5a_8}(r_6)$	*Mem5*	$+, C_{5a_1}, C_{5a_5}, C_{5a_6}, C_{5a_7}, C_{5a_9}, \gamma_5^{11}, C_{5a_1}, C_{5a_8}, C_{5a_1}, C_{7a_5}, C_{5a_7}, C_{5a_8}, C_{5a_9}, \gamma_7^1(r_7)$	⋯	$+, C_{5a_1}, C_{5a_5}, C_{5a_6}, C_{5a_7}, \gamma_5^7,, C_{5a_8}, C_{5a_9}(r_8)$	⋯
Mem7	$-, C_{5a_3}, C_{7a_5}, C_{5a_7}, C_{5a_8}, C_{5a_9}, \gamma_7^1(r_6)$					
Mem9	$+, C_{9a_5}, C_{9a_7}, C_{9a_8}, C_{9a_9}, \gamma_9^4$	*Mem9*	$+, C_{9a_5}, C_{9a_7}, C_{9a_8}, C_{9a_9}, \gamma_9^4$	⋯	$+, C_{9a_5}, C_{9a_7}, C_{9a_8}, C_{9a_9}, \gamma_9^4$	⋯
Mem10	$+, C_{10a_{10}}, C_{10a_{11}}, C_{10a_{12}}, (r_6) C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{10}^{10}$	*Mem10*	$+, C_{10a_{10}}, C_{10a_{11}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{10}^{10}, C_{12a_{10}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{12}^1, (r_7)$	⋯	$+, C_{10a_{10}}, C_{10a_{11}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{10}^6(r_8)$	⋯
Mem12	$-, C_{12a_{10}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{12}^1, (r_6)$					
Mem14	$+, C_{14a_{10}}, C_{14a_{12}}, C_{14a_{14}}, C_{14a_{15}}, \gamma_{14}^4, (r_6)$	*Mem14*	$+, C_{14a_{10}}, C_{14a_{10}}, C_{14a_{14}}, C_{14a_{15}}, \gamma_{14}^4, (r_7)$	⋯	$+, C_{14a_{10}}, C_{14a_{10}}, C_{14a_{14}}, C_{14a_{15}}, \gamma_{14}^4, (r_8)$	⋯

Table 8: the final result in P system

	Step34	*Step35*	*Step43*
Mem0	λ	λ	$\gamma_1^8, \gamma_{10}^6(r_{11})$
Mem1	$+, C_{1a_1}, C_{1a_2}, C_{1a_3}, C_{1a_4}, C_{1a_5}, C_{1a_6}, C_{1a_7}, C_{1a_8}, C_{1a_9}, \gamma_1^8(r_8)$	$C_{1a_1}, C_{1a_2}, C_{1a_3}, C_{1a_4}, C_{1a_5}, C_{1a_6}, C_{1a_7}, C_{1a_8}, C_{1a_9}, \gamma_1^8(r_{10})$	$C_{1a_1}, C_{1a_2}, C_{1a_3}, C_{1a_4}, C_{1a_5}, C_{1a_6}, C_{1a_7}, C_{1a_8}, C_{1a_9}(r_{11})$
Mem10	$+, C_{10a_{10}}, C_{10a_{11}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{10}^6(r_8)$	$C_{10a_{10}}, C_{10a_{11}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}, \gamma_{10}^6(r_{10})$	$C_{10a_{10}}, C_{10a_{11}}, C_{10a_{12}}, C_{10a_{13}}, C_{10a_{14}}, C_{10a_{15}}(r_{11})$