

## Article

# An Extended Membrane System with Monodirectional Tissue-like P Systems and Enhanced Particle Swarm Optimization for Data Clustering

Lin Wang <sup>1,\*</sup>, Xiyu Liu <sup>2</sup> , Jianhua Qu <sup>2</sup>, Yuzhen Zhao <sup>2</sup>, Liang Gao <sup>1</sup> and Qianqian Ren <sup>3</sup>

<sup>1</sup> School of Management Engineering, Shandong Jianzhu University, Jinan 250101, China; gaoliang20@sdjzu.edu.cn

<sup>2</sup> School of Business, Shandong Normal University, Jinan 250399, China; xyliu@sdnu.edu.cn (X.L.); qjh@sdnu.edu.cn (J.Q.); zhaoyuzhen@sdnu.edu.cn (Y.Z.)

<sup>3</sup> School of Computer Science, Qufu Normal University, Rizhao 276826, China; qianqianren@qfnu.edu.cn

\* Correspondence: wanglin92@sdjzu.edu.cn

**Abstract:** In order to establish a highly efficient P system for resolving clustering problems and overcome the computation incompleteness and implementation difficulty of P systems, an attractive clustering membrane system, integrated with enhanced particle swarm optimization (PSO) based on environmental factors and crossover operators and a distributed parallel computing model of monodirectional tissue-like P systems (MTP), is constructed and proposed, which is simply named ECPSO-MTP. In the proposed ECPSO-MTP, two kinds of evolution rules for objects are defined and introduced to rewrite and modify the velocity of objects in different elementary membranes. The velocity updating model uses environmental factors based on partitioning information and randomly replaces global best to improve the clustering performance of ECPSO-MTP. The crossover operator for the position of objects is based on given objects and other objects with crossover probability and is accomplished through the hybridization of the global best of elementary membranes to reject randomness. The membrane structure of ECPSO-MTP is abstracted as a network structure, and the information exchange and resource sharing between different elementary membranes are accomplished by evolutionary symport rules with promoters for objects of MTP, including forward and backward communication rules. The evolution and communication mechanisms in ECPSO-MTP are executed repeatedly through iteration. At last, comparison experiments, which are conducted on eight benchmark clustering datasets from artificial datasets and the UCI Machine Learning Repository and eight image segmentation datasets from BSDS500, demonstrate the effectiveness of the proposed ECPSO-MTP.

**Keywords:** particle swarm optimization; monodirectional tissue-like P systems; environmental factors; crossover operator; data clustering; image segmentation



**Citation:** Wang, L.; Liu, X.; Qu, J.; Zhao, Y.; Gao, L.; Ren, Q. An Extended Membrane System with Monodirectional Tissue-like P Systems and Enhanced Particle Swarm Optimization for Data Clustering. *Appl. Sci.* **2023**, *13*, 7755. <https://doi.org/10.3390/app13137755>

Academic Editor: Byung-Gyu Kim

Received: 5 June 2023

Revised: 26 June 2023

Accepted: 29 June 2023

Published: 30 June 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Membrane computing (MC) is a crucial area of nature-inspired computation that draws inspiration from the biological mechanisms and phenomena of organisms and was designed and developed by Păun [1]. The MC of modeling is the abstraction of the construction, activation, cooperation, and function of living cells in tissues or organs and is also known as P systems or membrane systems. In general, the computational model of MC consists of three essential elements, including the structure of membranes, objects, and rules [2]. There are two main types of MC, including Cell-like P systems (CPs) with a hierarchical structure and tissue-like P systems (TPs) or neural-like P systems (NPs) with a net structure [3]. It has been proven that these classic systems and their corresponding variations are Turing complete [4].

TPs are inspired by the construction and communication methods of living cells in tissues or organs. Information interchange between cells is accomplished through channel states [5,6] or symport/antiport rules [7]. Therefore, the substructure of TPs is vividly depicted by an undigraph. The study of TPs has been broadly divided into theoretical and application studies [8]. In theoretical studies, many kinds of TPs based on various biological motivations have been constructed and developed to expand the types of MC [9,10]. The analysis of these extended systems on computing power and computational efficiency is an essential point in theoretical works [11,12]. TPs with evolutionary symport/antiport are presented to rewrite and modify objects in the communication process [13,14]. Inspired by the presence of biocatalysts in biological reactions, a variation of TPs with promoters has been proposed to recruit promoters for solving image processing problems [15]. A novel kind of TP, monodirectional tissue-like P systems [16] and simply called MTPs, which is based on the biological fact that objects only move in one direction, has been constructed to enhance computing power in solving NP-complete problems [17–19].

The combination of evolutionary computation (EC) and TPs is a prominent feature of application studies and an important component of evolutionary membrane computing (EMC). Membrane-inspired evolutionary algorithms (MIEAs), also known as membrane algorithms (MAs), have become one of the mainstream approaches in EMC based on TPs [20]. MAs based on TPs have been merged with various heuristic algorithms, such as genetic algorithm (GA) [21], differential evolution (DE) [22] and its variations [23], particle swarm optimization (PSO) [24] and its variations [25], ant colony optimization (ACO) [26], and artificial bee colony algorithm (ABC) [27], to comprehensively utilize the strengths of heuristic algorithms with stronger practicality and high robustness, as well as the low complexity and effectiveness of TPs. MAs based on TPs have been successfully applied to resolve various advanced problems in the world [28–31].

PSO is a type of stochastic optimization approach that was originally initiated and designed by Kennedy and Eberhart [32]. The computing model is based on the collective behavior of birds flocking, and the trajectory of particles is adaptively adjusted according to individual and social experiences to balance exploration and exploitation of the algorithm [33]. Compared to other swarm intelligence (SI) approaches, PSO has fewer adjusting parameters and a simple implementation, which shows great potential for solving practical problems [34–36]. However, under the basic framework of SI, PSO can also easily fall into local optima and appear premature [37]. Therefore, several variants of PSO have been designed and developed to improve its search performance [38,39].

A lot of research has been conducted on adjusting the parameters of PSO through various optimization strategies, mainly consisting of adjusting the inertia weight [40,41] and the acceleration coefficients based on self-cognition and social-cognition adjustment [42]. Several kinds of modified position and velocity updating models have been constructed and presented to enhance the local and global optimization capacities of PSO, such as the crossover operator [43,44], which includes the multi-crossover operator [45] and the vertical crossover operator [46], and the mutation operator [47,48]. Furthermore, improved PSO integrates with various learning strategies that have been designed and developed to alter the exemplar particle in the population, including adaptive comprehensive learning strategies [49,50], adaptive learning strategies [51,52], and reinforcement learning strategies [53,54]. Some studies on PSO have utilized the multi-population strategy to split the whole population of particles into multiple smaller subpopulations to avoid prematurity [55,56]. Communication between different subpopulations is allowed to realize information exchange and sharing [57,58]. In order to utilize the advantages of both approaches, various variants of PSO have been proposed that combine PSO with other heuristic algorithms or optimization strategies to enhance the performance of PSO in solving engineering application problems [59–61].

Based on the above studies, MAs based on TPs provide a new way of enhancing the performance of PSO, which distinguishes them from previous improvements to PSO. Furthermore, the combination of PSO and TPs is also a new attempt to overcome limitations,

such as incomplete basic operations and implementation complexities in the original model. The properties of distributed and parallel processing in P systems are introduced to address the issue of exponential complexity increase in PSOs with problem scale enlargement. Under the computing framework of TPs, the particle population is divided into multiple sub-populations with a network structure, and communication between different sub-populations is achieved through the symport/antiport rules of TPs. Unlike previous related works about MAs, this paper modifies the evolutionary mechanism of PSO to better suit the computing framework of MTP. Each object in the elementary membrane has two evolution rules for updating velocity, which can be selected randomly to avoid premature convergence. Additionally, two evolution rules based on the classic PSO and crossover operator are defined to update the position of objects in different elementary membranes in order to balance the exploration and exploitation of the system. The exchange of information based on global best practices in a single direction is achieved through the specific communication mechanism of MTP, which distinguishes it from the related works on MAs based on TPs.

This work focuses on the combination of extended TPs with PSO for solving clustering problems. A novel variant of MIEAs or MAs with SNS, called ECPSO-MTP, is proposed and designed by integrating MTP and an improved PSO based on environmental factors and crossover operators. The proposed ECPSO-ECP comprises the evolutionary mechanism of modified PSO and the computation framework of MTP to establish a highly efficient P system for solving clustering problems. In ECPSO-MTP, two types of evolution rules are defined and introduced to rewrite and modify objects in different elementary membranes based on environmental factors and crossover operators. The environmental factors utilize the partitioning information to randomly replace the global best of objects and attend to the evolution process to improve the clustering performance of ECPSO-MTP. The crossover operator is achieved by hybridizing the global best of elementary membranes to reject randomness. The information exchange and resource sharing between different elementary membranes in ECPSO-MTP are accomplished through the evolutionary symport with promoters for objects of MTP, including forward and backward communication rules. Furthermore, eight test clustering and image datasets are employed in comparison experiments to verify the effectiveness of ECPSO-MTP compared with other existing approaches. The results from comparison experiments demonstrate the efficiency of the proposed ECPSO-MTP.

The structure of this paper is organized as follows: Section 2 provides a brief introduction to the basic framework of TPs with evolutionary symport/antiport and promoters and MTPs with evolutionary symport and promoters. Section 3 details the evolutionary mechanism of PSO based on environmental factors and crossover operators. Section 4 presents a detailed depiction of the proposed ECPSO-MTP, including its general framework, evolution rules, communication rules, and complexity analysis. Experimental results and discussions of the proposed ECPSO-MTP, which were conducted on eight clustering datasets and compared with five existing methods, are discussed and analyzed in Section 5. Section 6 illustrates the performance of the proposed ECPSO-MTP based on the comparison results of three clustering approaches on eight segmentation images. Finally, the summarization and conclusions for this paper and recommendations for future works are outlined in Section 7.

## 2. Tissue-like P Systems

### 2.1. TPs with Evolutional Symport/Antiport and Promoters

As a variant of TPs, TPs with evolutionary symport/antiport and promoters are based on the fact of biological reactions [62]. In classic TPs, substances or objects are transmitted through the execution of communication rules in different cells or regions, and communication between two given membranes is realized with the help of symport/antiport rules. In the presence of certain chemical substances, rules with promoters are executed to dynamically change the working manner of the computing model. The purpose of this ex-

tended P system is to modify and rewrite objects during the communication process using the framework of evolutionary mechanisms. Therefore, a recognizer TP with evolutionary symport/antiport and promoters is designed and defined as a tuple in the following [9]:

$$\Pi = (\Gamma, \varepsilon, \mu, \omega_1, \dots, \omega_m, R, \sigma_{in}, \sigma_{out})$$

where

- (1)  $\Gamma$  is nonempty finite set of alphabets consisting of elements, and all elements in this alphabet are called objects;
- (2)  $\varepsilon$  is a finite alphabet set of  $\Gamma$ , which represents objects initially placed in the environment, such that  $\varepsilon \subseteq \Gamma$ ;
- (3)  $\mu$  is the structure of the P system, which consists of  $m$  membranes;
- (4)  $\omega_1, \dots, \omega_m$  are finite multisets of objects over  $\Gamma$ , initially placed in  $m$  membranes, where  $\omega_i \subseteq \Gamma$ , for  $1 \leq i \leq m$ ;
- (5)  $R$  is a finite set of communication rules for the system, which contains two kinds of rules with promoters of the following restrictions.
  - ① Evolutional symport rules with promoters:  $[u|_p]_i \llbracket \rrbracket_j \rightarrow \llbracket \rrbracket_i [u']_j$ , where  $0 \leq i \neq j \leq m, u \in \Gamma^+, u' \in \Gamma^*, p \in \Gamma, |u| > 0$ . It can only be applied to a configuration if both the multiset of objects  $u$  and promoter objects  $p$  appear in the same existing membrane  $i$ . When such an evolutional symport rule with promoters associated with membrane  $i$  and membrane  $j$  is employed, the multiset of objects  $u$  under the presence of promoter objects  $p$  in membrane  $i$  is simultaneously sent to membrane  $j$  and evolved into new objects  $u'$ ;
  - ② Evolutional antiport rules with promoters:  $[u|_p]_i [v]_j \rightarrow [v']_i [u']_j$ , where  $0 \leq i \neq j \leq m, u, v \in \Gamma^+, u', v' \in \Gamma^*, p \in \Gamma, |u|, |v| > 0$ . It can only be applied to a configuration if both  $u$  and  $p$  appear in the same existing membrane  $i$ , and another membrane  $j$  in the same configuration contains  $v$ . When such an evolutional antiport rule with promoters associated with  $i$  and  $j$  is employed,  $u$  under the presence of  $p$  in  $i$  is simultaneously sent to  $j$  and evolved into  $u'$ , at the same time,  $v$  in  $j$  is also simultaneously sent to  $i$  and evolved into new objects  $v'$ ;
- (6)  $\sigma_{in}$  represents an input membrane or region, where  $\sigma_{in} \in \{\sigma_0, \sigma_1, \dots, \sigma_m\}$ ;
- (7)  $\sigma_{out}$  represents an output membrane or region, where  $\sigma_{out} \in \{\sigma_0, \sigma_1, \dots, \sigma_m\}$ .

In the TPs with evolutionary symport/antiport and promoters, unlike other objects involved in computation, the promoter objects do not directly take part in the execution process of evolutionary symport/antiport rules. They are not modified or changed during computation. However, it only alters the order in which the evolutionary symport/antiport rules are executed. In the presence of promoter objects, a greater number of corresponding evolutionary symport/antiport rules are implemented within the fixation time. Once the promoters are empty, the execution of evolutionary symport/antiport rules will cease to be effective. Overall, the TPs with evolutionary symport/antiport and promoters are applied in a maximally parallel manner, and each membrane of the system also works in a maximally parallel way.

Specifically, the evolution rules for objects are introduced in the computing model of TPs, which are based on the biological facts that molecules or substances can be modified and changed in membranes [63]. The evolution of objects is achieved through the execution of rewriting rules, which are of the form:  $[u \rightarrow v]_i$ , where  $u, v \in \Gamma^+$ . It can only be applied to a configuration if the multiset of objects  $u$  appears in the existing membrane  $i$ . When such an evolution rule is applied, objects  $u$  in  $i$  evolve into objects  $v$ , noted that the position of  $u$  remaining unchanged during the evolutionary computation.

### 2.2. MTPs with Evolutional Symport and Promoters

MTPs are an attractive variant of TPs that are based on the biological phenomenon of molecules or substances moving from high to low concentration in membranes or regions [16]. The computational power of MTPs has been proven in a flat maximally parallel model [18]. Furthermore, restrictive conditions of evolutionary symport rules with promoters are introduced in MTPs, which are referred to as MTPs. For any two given membranes or regions, only evolutionary symport rules with promoters are permitted to realize the movement and modification of objects in one direction.

The evolutionary symport rules with promoters in MTP are given in the form,  $[u|p]_i[\lambda]_j \rightarrow [\lambda]_i[v]_j$ , or  $[\lambda|p]_i[u]_j \rightarrow [v]_i[\lambda]_j$ , where  $0 \leq i \neq j \leq m$ ,  $u, v \in \Gamma^+$ ,  $p \in \Gamma$ ,  $|u| > 0$ . The first kind of rule is applied to a configuration if both  $u$  and  $p$  appear in the same existing membrane  $i$ . When such an evolutionary symport rule with promoters associated with  $i$  and  $j$  is employed, under the presence of  $p$ , objects  $u$  in  $i$  are simultaneously sent to  $j$  and revised to  $v$ . The second kind of rule is applied to a configuration if  $p$  and  $u$  appear in  $i$  and  $j$  respectively. When such an evolutionary symport rule with promoters associated with  $i$  and  $j$  is employed, under the presence of  $p$  in  $i$ , objects  $u$  in  $j$  are simultaneously sent to  $i$  and revised to  $v$ .

## 3. Improved Particle Swarm Optimization

### 3.1. Environmental Factors

From a biological perspective, the social behavior of birds is affected by their living environment. Specifically, the flying space of birds is constrained by environmental factors within a certain period. Additionally, the characteristics of birds can also affect their living environment, resulting in differences between individuals. To enhance the diversity of species, environmental factors are introduced to the classic model of PSO [59]. At each iteration  $t + 1$ , the new velocity  $V_i(t + 1)$  of particle  $i$  is defined by (1) in the following:

$$V_i(t + 1) = \omega * V_i(t) + c_1 * r_1 * (X_i^{lbest}(t) - X_i(t)) + c_2 * r_2 * (X^{gbest}(t) - X_i(t)) + c_3 * r_3 * (X_i^{ebest}(t) - X_i(t)) \tag{1}$$

where  $t$  is the iteration counter.  $\omega$  represents the inertia weight.  $c_1$  and  $c_2$  are acceleration coefficients based on self-cognition and social cognition.  $r_1$  and  $r_2$  are two uniform random numbers.  $X_i^{lbest}(t)$  is the local best of  $i$  at  $t$ .  $X^{gbest}(t)$  is the global best of  $i$  at  $t$ . Specifically,  $c_3$  is the positive regulation constant with a uniform random number.  $X_i^{ebest}(t)$  is the environmental factor of  $i$  at  $t$ , which is defined by (2) in the following:

$$X_i^{ebest}(t) = \frac{\sum_{j=1}^{n_k} y_{k,j}}{n_k} \tag{2}$$

where  $X_i^{ebest}(t)$  consisting of a finite set of cluster centers for  $i$  at  $t$ , and the number of finite set is  $K$ ,  $X_i^{ebest}(t) = \{X_{i1}^{ebest}(t), X_{i2}^{ebest}(t), \dots, X_{iK}^{ebest}(t)\}$ ,  $K$  is the number of clusters.  $y_{k,j}$  is the feature of data point  $j$  belonging to the corresponding cluster  $k$ , for  $1 \leq k \leq K$ . And  $n_k$  is the totality of data points attaching to cluster  $k$ .

### 3.2. Crossover Operator

In the standard model of PSO, the trajectory adjustment of particles is based on their local and global experiences. Local and global best are introduced to regulate motion, resulting in rapid information propagation throughout the particle population. However, this rapid propagation in population leads to the standard PSO becoming trapped in local optima. Therefore, a novel variant of the updating method that integrates PSO and GA is proposed to improve performance. An arithmetic crossover operator is introduced into the generation process of the particle population to inject randomness by adding a weighted position between a random particle and a given particle [43]. Specifically, the crossover

operator only takes place in a given particle  $i$  and random particle  $r$  with a crossover probability  $\varphi$ , at  $t + 1$ , the new position  $X_i(t + 1)$  of  $i$  is defined by (3) in the following:

$$X_i(t + 1) = X_i(t) + \gamma(t) * X_r(t + 1), \tag{3}$$

where  $\gamma(t)$  is the weight parameter  $\gamma$  at  $t$ .  $X_r(t)$  is the position corresponds to a randomly selected particle  $r$  from particle's population at  $t$ , noted that  $r \neq i$ . Additionally, at  $t + 1$ , the crossover probability  $\varphi$  is determined by (4) in the following:

$$\varphi(t + 1) = \theta_\varphi * \varphi(t), \tag{4}$$

where  $\theta_\varphi$  is the damping parameter. At  $t + 1$ , the weight parameter  $\gamma$  is determined by (5) in the following:

$$\gamma(t + 1) = \theta_\gamma * \gamma(t), \tag{5}$$

where  $\theta_\gamma$  is the damping factor. Specially, the initial values of the crossover probability  $\varphi$  and weight parameter  $\gamma$  are typically set to 1.

#### 4. The Proposed ECPSO-MTP

In this section, a novel clustering membrane system is designed and developed that combines the computation framework of MTP and the evolutionism of PSO based on environmental factors and a crossover operator and is simply named ECPSO-MTP. The evolution mechanism of the system involves the evolution of objects in different membranes, which is achieved through the velocity updating model based on environmental factors and the position updating model based on the crossover operator. The communication mechanism is accomplished through evolutionary symport with promoters of MTP to realize the transport and exchange of objects between membranes. Thus, the computing model of ECPSO-MTP consists of two mechanisms for objects, i.e., the evolution mechanism and the communication mechanism. More details about the proposed ECPSO-MTP are depicted as follows.

##### 4.1. The Common Framework of ECPSO-MTP

The membrane system of the proposed ECPSO-MTP is defined and described as a tuple, which is completely depicted in the following:

$$\Pi = (\Gamma, \varepsilon, \mu, \omega_1, \dots, \omega_m, R, R', \sigma_{in}, \sigma_{out}),$$

where

- (1)  $\Gamma$  is a non-empty, finite alphabet for objects;
- (2)  $\varepsilon \subseteq \Gamma$  is a finite alphabet set for objects that are initially placed in the input membrane;
- (3)  $\mu$  is the structure of the ECPSO-MTP, which contains  $m + 4$  elementary membranes;
- (4)  $\omega_1, \dots, \omega_m$  are finite multisets for objects that are initially placed in  $m$  elementary membranes, where  $\omega_i \subseteq \Gamma$ , for  $1 \leq i \leq m$ ;
- (5)  $R$  is the finite set of evolution rules for objects in the proposed ECPSO-MTP, where  $R = \{R_1, R_2, \dots, R_m, R_{m+2}\}$ .  $R_i$  is a finite subset of  $R$  associated with membrane  $i$ , for  $1 \leq i \leq m$  or  $i = m + 2$ , and is of the form:  $R_i = [u \rightarrow u']_i$ , for  $u, u' \in \Gamma^+$ . When such an evolution rule  $R_i$  in  $i$  is applied, objects  $u$  in  $i$  evolve into  $u'$ ;
- (6)  $R'$  is the finite set of communication rules for objects in the proposed ECPSO-MTP, where  $R' = \{R'_1, R'_2, \dots, R'_{m+2}\}$ .  $R'_i$  is a finite subset of  $R'$  associated with  $i$  using the evolutionary symport with promoters of MTP. Specifically,  $R'_{ij} \subseteq R'_i$  and is of the form:  $R'_{i,j} : [u|_p]_i[\lambda]_j \rightarrow [\lambda]_i[u']_j$ , or  $R'_{i,j} : [\lambda|_p]_i[u]_j \rightarrow [u']_i[\lambda]_j$ , where  $1 \leq i \neq j \leq m + 2$ ,  $u \in \Gamma^+$ ,  $u' \in \Gamma^*$ ,  $p \in \Gamma$ ,  $|u| > 0$ ;
- (7)  $\sigma_{in}$  is the input membrane of the proposed ECPSO-MTP;

- (8)  $\sigma_{out}$  is the output membrane of the proposed ECPSO-MTP. When the computation of this extended P system is completed, objects in the output membrane  $\sigma_{out}$  will be sent to the environment  $\sigma_0$ , which is regarded as the final computational result of the system. The membrane structure of the proposed ECPSO-MTP is graphically depicted in Figure 1.

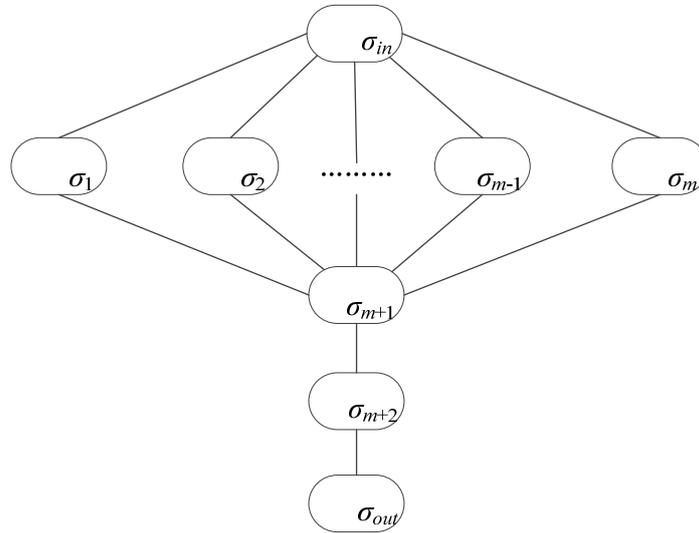


Figure 1. The membrane structure of the proposed ECPSO-MTP.

The membrane structure of ECPSO-MTP can be abstracted as the hierarchical structure in mathematics, which is depicted in Figure 1. This extended P system contains  $m + 4$  membranes, including an input membrane  $\sigma_{in}$  and an output membrane  $\sigma_{out}$ , while the others are labeled from 1 to  $m + 2$ . Specifically, if there are no membranes contained within a membrane, it is called an elementary membrane. In this case, membranes  $\sigma_1$  to  $\sigma_{m+2}$  are elementary membranes. Additionally,  $\sigma_{m+2}$  is also called a comparison membrane, and the best object selected from  $\sigma_1$  to  $\sigma_{m+1}$  is stored as the best object of  $\sigma_{m+2}$ .

#### 4.2. Evolution Rules

In ECPSO-MTP, two types of evolution rules are defined and described for objects in different elementary membranes, including elementary membrane  $\sigma_o$  ( $1 \leq o \leq m$ ) and elementary membrane  $\sigma_{m+1}$ , respectively. The evolution rules are used to evolve objects. And the evolutionary mechanism of modified PSO with environmental factors and crossover operators is adopted to accomplish this evolution in different membranes. Specifically, an object  $u_i$  is composed of two essential components, including velocity  $V_i$  and position  $X_i$ , where  $u_i = \{V_i, X_i\}$ .

Two types of evolution rules are introduced to modify the velocity of objects in elementary membranes, based on the velocity updating model of classic PSO and modified PSO with environmental factors. Thus,  $V_i(t + 1)$  of  $u_i$  in  $\sigma_o$  at  $t + 1$  is defined by (6) in the following:

$$V_i(t + 1) = \omega * V_i(t) + c_1 * r_1 * (X_i^{lbest}(t) - X_i(t)) + c_2 * r_2 * (X_o^{gbest}(t) - X_i(t)), \quad (6)$$

Another evolution rule is defined by (7) in the following:

$$V_i(t + 1) = \omega * V_i(t) + c_1 * r_1 * (X_i^{lbest}(t) - X_i(t)) + c_2 * r_2 * (X_i^{gbest}(t) - X_i(t)), \quad (7)$$

And  $\omega$  is determined by (8) in the following:

$$\omega(t) = \omega_{min} + (\omega_{max} - \omega_{min}) * (t/t_{max}), \quad (8)$$

where  $\omega_{\min}$  and  $\omega_{\max}$  represent the minimum and maximum values of  $\omega$ . The maximum number of iterations is denoted by  $t_{\max}$ .  $c_1$  and  $c_2$  are typically set to 2.  $X_i^{lbest}(t)$  represents the local best of  $u_i$ , and is also indicated as  $u_i^{lbest}(t)$ .  $X_o^{gbest}(t)$  represents the global best of  $u_i$  in  $\sigma_o$ , and is also denoted by  $u_o^{gbest}(t)$ .  $X_i^{ebest}(t)$  is the environmental factor of  $u_i$  and is determined by Equation (2).

The position updating model of the classic PSO is employed to modify the position of objects, and  $X_i(t + 1)$  of  $u_i$  in  $\sigma_o$  is defined by (9) in the following:

$$X_i(t + 1) = X_i(t) + V_i(t + 1), \tag{9}$$

Then,  $u_i^{lbest}(t + 1)$  of  $u_i$  at  $t + 1$  is described by (10) in the following:

$$u_i^{lbest}(t + 1) = \begin{cases} X_i(t + 1), & \text{if } f(X_i(t + 1)) < f(X_i^{lbest}(t)), \\ X_i^{lbest}(t), & \text{otherwise} \end{cases} \tag{10}$$

where  $f()$  represents the fitness of the fitness function. And  $u_o^{gbest}(t + 1)$  of  $\sigma_o$  at  $t + 1$  is described by (11) in the following:

$$u_o^{gbest}(t + 1) = \begin{cases} X_i^{lbest}(t + 1), & \text{if } f(X_i^{lbest}(t + 1)) < f(X_o^{gbest}(t)) \\ X_o^{gbest}(t), & \text{otherwise} \end{cases} \tag{11}$$

In addition, the position updating model based on the crossover operator is adopted to modify the position of objects in  $\sigma_{m+1}$  using the Equations (3)–(5). And the global best  $u_{m+1}^{gbest}(t + 1)$  of  $\sigma_{m+1}$  at  $t + 1$  is the best position in  $\sigma_{m+1}$ .

### 4.3. Communication Rules

In ECPSO-MTP, evolutional symport with promoters of MTP is introduced to facilitate communication between elementary membranes. Two kinds of communication rules for objects through the conveyor direction of information, including forward and backward communication rules, are defined and adopted to enable communication to take place in different elementary membranes. More details about these communication rules are depicted as follows.

#### 4.3.1. Forward Communication Rules

Under the hierarchical structure of the proposed ECPSO-MTP, forward communication rules are applied to establish transitive relationships between elementary membranes. There are two types of communication rules in the onward direction, including  $\sigma_o$  ( $1 \leq o \leq m$ ) to  $\sigma_{m+1}$  and  $\sigma_{m+1}$  to  $\sigma_{m+2}$ . It is noted that there is no interaction between elementary membranes at the same level of membrane structure.

The forward communication rules from  $\sigma_o$  to  $\sigma_{m+1}$  are given in the form,  $R'_{o,m+1} : [u_o^{gbest}(t)|_p]_o [\lambda]_{m+1} \rightarrow [\lambda]_o [u_o(t)]_{m+1}$ , for  $1 \leq o \leq m$ . It only can be applied on a configuration if both  $u_o^{gbest}(t)$  and  $p$  appear in  $\sigma_o$  at  $t$ . When this communication rule associated with  $\sigma_o$  and  $\sigma_{m+1}$  is applied, under the presence of  $p$ ,  $u_o^{gbest}(t)$  in  $\sigma_o$  is simultaneously sent to  $\sigma_{m+1}$  and evolve into  $u_o(t)$ . Therefore, the totality of objects from  $\sigma_1$  to  $\sigma_m$  in  $\sigma_{m+1}$  is  $m$ .

The forward communication rules from  $\sigma_{m+1}$  to  $\sigma_{m+2}$  are given in the form,  $R'_{m+1,m+2} : [u_o^{lbest}(t)|_p]_{m+1} [\lambda]_{m+2} \rightarrow [\lambda]_{m+1} [u_o(t)]_{m+2}$ , for  $1 \leq o \leq m$ . It only can be applied on a configuration if both  $u_o^{lbest}(t)$  and  $p$  appear in  $\sigma_{m+1}$  at  $t$ . When this communication rule associated with  $\sigma_{m+1}$  and  $\sigma_{m+2}$  is applied, under the presence of  $p$ ,  $u_o^{lbest}(t)$  in  $\sigma_{m+1}$  is simultaneously sent to  $\sigma_{m+2}$  and revised to  $u_o(t)$ . Thus, the best object from  $\sigma_{m+1}$  is selected and stored as the global best  $u_{m+2}^{gbest}(t)$  of  $\sigma_{m+2}$  at  $t$ .

#### 4.3.2. Backward Communication Rules

Under the hierarchical structure of proposed ECPSO-MTP, backward communication rules based on the backward direction of information are applied from  $\sigma_o$  ( $1 \leq o \leq m$ ) to  $\sigma_{m+2}$ . The backward communication rules from  $\sigma_o$  to  $\sigma_{m+2}$  are given in the form,  $R'_{o,m+2} : [\lambda|q]_o [u_{m+2}^{gbest}(t)]_{m+2} \rightarrow [u_o^{gbest}(t)]_o [\lambda]_{m+2}$ , for  $1 \leq o \leq m$ . It only can be applied on a configuration if both  $q$  and  $u_{m+2}^{gbest}(t)$  are respectively present in  $\sigma_o$  and  $\sigma_{m+2}$  at  $t$ , noted that  $q \neq p$ . When this communication rule associated with  $\sigma_o$  and  $\sigma_{m+2}$  is applied, under the presence of  $q$  in  $\sigma_o$ ,  $u_{m+2}^{gbest}(t)$  in  $\sigma_{m+2}$  is simultaneously sent to  $\sigma_o$  and evolves into  $u_o^{gbest}(t)$ .

#### 4.4. Computation of Proposed ECPSO-MTP

##### (1) Initialization

###### <1> Parameters initialized

In the proposed ECPSO-MTP, the values of adjusting parameters are preset by adding the following values to the existing ones, including  $N$  and  $n_o$ , where  $N$  is the totality of objects in the system,  $n_o$  is the totality of objects in  $\sigma_o$  ( $1 \leq o \leq m$ ). The static membrane structure of the proposed ECPSO-MTP is shown in Figure 1 in more details. All objects involved in the computational procedure of the system are located in  $\sigma_{in}$  during initialization;

###### <2> Velocity and position initialized

To initialize the velocity and position for all objects in  $\sigma_{in}$ , a random generation strategy in search space is employed. Then these objects are sent to the elementary membrane  $\sigma_1$  to  $\sigma_m$  respectively, where  $n_o = N/m$ ;

###### <3> Local and global best updated

The local best  $u_i^{lbest}$  ( $1 \leq i \leq n_o$ ) and global best  $u_o^{gbest}$  in  $\sigma_o$  are generated by the Equations (10) and (11);

##### (2) Evolution mechanism for $\sigma_1$ to $\sigma_m$

###### <1> Velocity and position updated

The evolution rules for the velocity updating of all objects in  $\sigma_o$ , based on classic PSO and modified PSO with environmental factors, are randomly employed according to Equations (6)–(8), and the position updating is determined by Equation (9);

###### <2> Local and global best updated

The local best  $u_i^{lbest}$  ( $1 \leq i \leq n_o$ ) and global best  $u_o^{gbest}$  in  $\sigma_o$  are generated by the Equations (10) and (11);

##### (3) Forward communication mechanism from $\sigma_o$ to $\sigma_{m+1}$

Under the presence of both  $p$  and  $u_o^{gbest}$  in  $\sigma_o$ , the first type of forward communication rules are employed to transmit  $u_o^{gbest}$  from  $\sigma_o$  to  $\sigma_{m+1}$  and evolve it into  $u_o$  in  $\sigma_{m+1}$ . Specifically, the promoter objects  $p$  can be described as logical judgements, where  $p = \{t < t_{max}\}$ ;

##### (4) Evolution mechanism for $\sigma_{m+1}$

###### <1> Position updated

The evolution rules in  $\sigma_{m+1}$  based on the crossover operator are employed to update objects in  $\sigma_{m+1}$  according to Equations (3)–(5);

###### <2> Local best updated

Update  $u_o^{lbest}$  for all objects in  $\sigma_{m+1}$  according to Equation (10);

##### (5) Forward communication mechanism from $\sigma_{m+1}$ to $\sigma_{m+2}$

Under the presence of both  $p$  and  $u_o^{lbest}$  in  $\sigma_{m+1}$ , the second type of forward communication rules is employed to transmit  $u_o^{lbest}$  from  $\sigma_{m+1}$  to  $\sigma_{m+2}$  and evolve it into  $u_o$  in  $\sigma_{m+2}$ . The best object from  $\sigma_{m+1}$  is regarded as the global best  $u_{m+2}^{gbest}$  of  $\sigma_{m+2}$ . Then  $u_{m+2}^{gbest}$  is sent to  $\sigma_{out}$  as the computed result of the system at the moment;

(6) Backward communication mechanism from  $\sigma_0$  to  $\sigma_{m+2}$

Under the presence of  $q$  in  $\sigma_0$  and  $u_{m+2}^{gbest}$  in  $\sigma_{m+2}$ , the backward communication rules from  $\sigma_0$  to  $\sigma_{m+2}$  are utilized to transmit  $u_{m+2}^{gbest}$  from  $\sigma_{m+2}$  to  $\sigma_0$  and evolve it into  $u_0^{gbest}$  in  $\sigma_0$ . Specifically, the promoter objects  $q$  can be described as logical judgements, where  $q = \{f(u_{m+2}^{gbest}) < f(u_0^{gbest})\}$ . The communication relationships within the system are graphically depicted in Figure 2, with black arrows indicating the direction of communication;

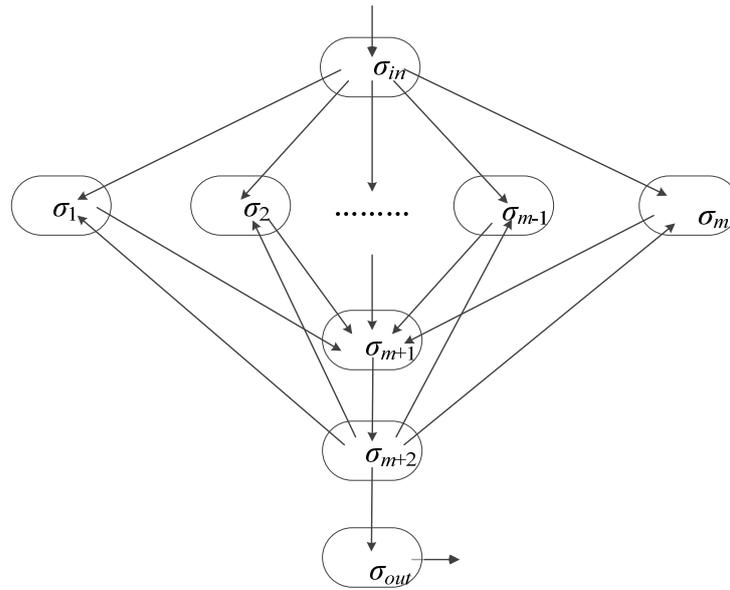


Figure 2. The communication relationship in the proposed ECPSO-MTP.

(7) Termination and output

The evolution-communication mechanism in ECPSO-MTP is executed repeatedly through iteration until the maximum number of iterations is attained. When the system stops, the last object of  $u_{m+2}^{gbest}$  in  $\sigma_{out}$  is transmit to the environment  $\sigma_0$ , which can be represented by the final computed result of the proposed ECPSO-MTP. The pseudocode for the computation of the proposed ECPSO-MTP is depicted in the Algorithm 1 as follows.

---

**Algorithm 1** ECPSO-MTP

---

Input:  $N, t_{max}, c_1, c_2, r_1, r_2, \omega_{min}, \omega_{max}, \theta_\varphi, \theta_\gamma m;$

(1) Initialization

<1> Velocity and position initialized

for  $i = 1$  to  $N$

Velocity of object  $u_i$ :  $V_i = rand(s^l, s^u);$

Position of object  $u_i$ :  $X_i = rand(s^l, s^u);$

end

<2> Local and global best updated

for  $o = 1$  to  $m$

for  $i = 1$  to  $n$

Update local best  $u_i^{lbest}$  of objects  $u_i$  according to Equation (10);

Update global best  $u_o^{gbest}$  of object  $u_i$  according to Equation (11);

end

end

---

---

**Algorithm 1** ECPSO-MTP

---

```

(2) Evolution mechanism for  $\sigma_1$  to  $\sigma_m$ 
    <1> Velocity and position updated
        for  $o = 1$  to  $m$ 
            for  $i = 1$  to  $n$ 
                Update velocity  $V_i$  of object  $u_i$  based on a randomly selection strategy according to
                Equations (6)–(8);
                Update position  $X_i$  of object  $u_i$  according to Equation (9);
            end
        end
    <2> Local and global best updated
(3) Forward communication mechanism from  $\sigma_o$  to  $\sigma_{m+1}$ 
    if  $p = \{t < t_{\max}\}$ 
        for  $o = 1$  to  $m$ 
             $R'_{o,m+1} : [u_o^{gbest}(t)|_p]_o [\lambda]_{m+1} \rightarrow [\lambda]_o [u_o(t)]_{m+1};$ 
        end
    end
(4) Evolution mechanism for  $\sigma_{m+1}$ 
    <1> Position updated
        Update position  $X_i$  of object  $u_i$  in  $\sigma_{m+1}$  according to Equations (3)–(5);
    <2> Local best updated
(5) Forward commutation mechanism from  $\sigma_{m+1}$  to  $\sigma_{m+2}$ 
    if  $p = \{t < t_{\max}\}$ 
         $R'_{m+1,m+2} : [u_o^{gbest}(t)|_p]_{m+1} [\lambda]_{m+2} \rightarrow [\lambda]_{m+1} [u_o(t)]_{m+2};$ 
    end
(6) Backward communication mechanism from  $\sigma_o$  to  $\sigma_{m+2}$ 
    if  $q = \{f(u_{m+2}^{gbest}) < f(u_o^{gbest})\}$ 
        for  $o = 1$  to  $m$ 
             $R'_{o,m+2} : [\lambda|_q]_o [u_{m+2}^{gbest}(t)]_{m+2} \rightarrow [u_o^{gbest}(t)]_o [\lambda]_{m+2};$ 
        end
    end
(7) Termination and output
    if  $t > t_{\max}$ 
        Best position of P system:  $u_{m+2}^{gbest}$ ;
        Best Fitness of P system:  $f(u_{m+2}^{gbest})$ ;
    end
Output: Best position of P system; best fitness of P system;

```

---

**4.5. Complexity Analysis**

The complexity of ECPSO-MTP for solving clustering problems is discussed and analyzed in this subsection. Firstly, some clarifications are given as follows:  $M$  represents the totality of data points in the dataset and  $D$  represents the dimension of data points.  $K$  represents the totality of clusters in the dataset, usually  $D \ll M$ , and  $K \ll M$ .

The computation of the proposed ECPSO-MTP consists of three main steps, including initialization, evolution, and communication phases. In the initialization phase, the computation time is mostly determined by the calculation cost of the fitness function, and the time needed for distance computation for one object is  $KMD$ , which can be simplified to  $M$ . The computation time for all objects in the system is  $nM$ , and the complexity for the initialized state of ECPSO-MTP is  $O(nM)$ . In the evolution phase, the time of evolution rules needed by executing once in  $\sigma_o$  is  $nM$ . The computation time of the evolution phase for  $\sigma_1$  to  $\sigma_m$  under the distributed and parallel processing of P systems is  $nM$ . And the computation time of the evolution phase for  $\sigma_{m+1}$  is  $mM$ . Then the complexity of the evolution state of ECPSO-MTP is  $O((n + m)M)$ . In the communication phase, the number of communication rules needed to be executed once in ECPSO-MTP is set to 1. The communication time from

$\sigma_0$  to  $\sigma_{m+1}$ ,  $\sigma_{m+1}$  to  $\sigma_{m+2}$ ,  $\sigma_0$  to  $\sigma_{m+2}$ , is equal to 3, and the complexity of the communication state of ECPSO-MTP is  $O(3)$ . Therefore, the time of the whole system by executing one iteration is  $(n + m)M + 3$ , and the cumulative time of the system is  $nM + ((n + m)M + 3)t_{\max}$ . Then the complexity of the proposed ECPSO-MTP is  $O(nMt_{\max})$ .

### 5. The Proposed ECPSO-MTP for Data Clustering

In this section, computational experiments with accuracy and convergence speed are introduced in order to verify the effectiveness of the proposed ECPSO-MTP, and a suite of commonly used datasets, including Artificial and UCI datasets, are adopted. To further demonstrate the validity of the proposed ECPSO-MTP, five previous clustering approaches are employed as comparison approaches. All clustering approaches are executed on a Dell desktop computer equipped with an Intel 8.00 GHz i7-8550U processor and 16 GB of RAM, operating under the Windows 11 environment.

#### 5.1. Test Datasets

Test datasets, including artificial and UCI datasets, are commonly used to testify to the effectiveness of clustering approaches. In this comparison experiment, eight datasets, consisting of two artificial and six UCI datasets, have been previously reported and adopted as benchmarks in research. Two artificial datasets, i.e., Data\_9\_2 and Square4, are generated manually by existing literature [64]. Six UCI datasets, i.e., Iris, Newthyroid, Seeds, Yeast, Glass, and Wine, are from the UCI Machine Learning Repository [65]. Additional details of these test datasets are briefly described in Table 1.

**Table 1.** Description of eight test datasets in the comparison experiment.

Datasets	Data Instances	Features	Clusters
Data_9_2	900	2	9
Square4	1000	2	4
Iris	150	4	3
Newthyroid	215	5	3
Seeds	210	7	3
Yeast	1484	8	10
Glass	214	9	6
Wine	178	13	3

#### 5.2. Comparison with Other Existing Approaches

Five existing clustering approaches, including classic PSO, genetic algorithm (GA), differential evolution (DE), PSO with environmental factors (EPSO) [59], and PSO with an enhanced learning strategy and crossover operator (PSO-LC) [43], are compared in this computational experiment. And EPSO and PSO-LC employ different strategies to improve performance. EPSO integrates environmental factors based on a partitioning of the dataset into the generation process of particles. Three strategies, such as altering the exemplar particles, modifying adjusting parameters, including inertia weight and learning factors, and integrating PSO with GA, are applied in PSO-LC. The values of the employed parameters used in these comparative approaches are presented in Table 2.

Specifically, mean squared error (MSE) [67], as the fitness function of comparative clustering approaches, is introduced in the comparison experiments, which is determined by (12) in the following:

$$\min_{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K} f = \min f(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K) = \min \frac{1}{M} \sum_{i=1}^K \sum_{j=1}^M \rho_{ij} \|\mathbf{x}_j - \mathbf{c}_i\|^2, \tag{12}$$

where  $\rho$  is the partition matrix of the dataset, if  $\rho_{ij} = 1$ , where indicates that data point  $\mathbf{x}_j$  belongs to cluster  $i$ , otherwise,  $\rho_{ij} = 0$ .  $\mathbf{x}_j$  represents the  $j$ -th data point in the dataset, for  $1 \leq j \leq M$ .  $\mathbf{c}_i$  is the clustering center of cluster  $i$ , for  $1 \leq i \leq K$ . The convergence results of

the six tested comparative approaches on eight datasets for typical runs are depicted in Figure 3.

**Table 2.** Preferences in the comparison experiments.

Parameters	Comparative Approaches					
	PSO	GA	DE	EPSO	PSO-LC	ECPSO-MTP
$N$	200	200	200	200	200	200
$t_{max}$	200	200	200	200	200	200
$c_1, c_2$	2,2	N	N	0.6,3	(0,1)	2,2
$c_3$	N	N	N	(0,1)	N	N
$r_1, r_2$	(0,1)	N	N	(0,1)	N	(0,1)
$r_3$	N	N	N	(0,1)	N	N
$\omega_{min}, \omega_{max}$	0.4,1.2	N	N	0.4,0.6	N	0.4,1.2
$\theta_{\omega}$	N	N	N	N	0.99	N
$P_c$	N	0.6	0.6	N	N	N
$P_m$	N	0.02	N	N	N	N
$\theta_{\varphi}$	N	N	N	N	0.994	0.994
$\theta_{\gamma}$	N	N	N	N	0.995	0.995
$F$	N	N	(0.5,1)	N	N	N
$m$	N	N	N	N	N	10 [66]

It can be clearly observed in Figure 3 that the proposed ECPSO-MTP achieves mostly the best MSE values on eight test datasets. To eliminate randomness, each test clustering method was independently implemented 50 times. Simple statistical results, including the worst values (Worst), best values (Best), mean values (Mean), and standard deviations (SD) of MSE, for these optimization approaches on the test datasets are presented in Table 3.

**Table 3.** Performance of six clustering approaches on test datasets (MSE).

Datasets	Statistics	Comparative Approaches					
		PSO	GA	DE	EPSO	PSO-LC	ECPSO-MTP
Data_9_2	Worst	0.7260	0.5597	0.6505	0.5240	0.6223	0.5248
	Best	0.6476	0.5240	0.5230	0.5215	0.5218	0.5212
	Mean	0.6854	0.5407	0.5361	0.5222	0.5343	0.5214
	S.D.	0.0184	0.0079	0.0275	0.0006	0.0252	0.0007
Square4	Worst	7.3164	7.0736	6.9807	6.9520	6.9525	6.9520
	Best	6.9780	6.9733	6.9520	6.9519	6.9519	6.9519
	Mean	7.1100	7.0195	6.9569	6.9519	6.9520	6.9519
	S.D.	0.0862	0.0274	0.0051	$6.81 \times 10^{-6}$	$8.59 \times 10^{-5}$	$5.49 \times 10^{-6}$
Iris	Worst	1.0158	0.5873	0.9578	0.6619	0.5729	0.5263
	Best	0.5268	0.5450	0.5266	0.5263	0.5287	0.5263
	Mean	0.5663	0.5621	0.5433	0.5357	0.5418	0.5263
	S.D.	0.1178	0.0099	0.0614	0.0236	0.0103	$1.09 \times 10^{-5}$
Newthyroid	Worst	152.0870	157.1877	175.1488	136.4225	138.1924	132.9317
	Best	135.9901	133.9973	133.1129	132.8665	135.0995	132.8379
	Mean	143.5047	139.2553	137.6728	133.9052	136.5018	132.8426
	S.D.	4.3043	4.6420	7.2724	1.2267	0.7346	0.0187
Seeds	Worst	4.8188	3.0923	3.0838	3.0387	3.2346	2.7968
	Best	2.8261	2.9253	2.8306	2.8119	2.7975	2.7968
	Mean	3.2956	3.0059	2.9215	2.8862	2.9042	2.7968
	S.D.	0.7528	0.0382	0.0467	0.0530	0.1070	$2.03 \times 10^{-6}$
Yeast	Worst	0.0822	0.0797	0.0661	0.0465	0.0668	0.0341
	Best	0.0521	0.0578	0.0586	0.0388	0.0494	0.0306
	Mean	0.0721	0.0674	0.0623	0.0427	0.0585	0.0315
	S.D.	0.0082	0.0076	0.0016	0.0019	0.0043	0.0008

Table 3. Cont.

Datasets	Statistics	Comparative Approaches					
		PSO	GA	DE	EPSO	PSO-LC	ECPSO-MTP
Glass	Worst	5.0427	3.3902	2.7652	2.2345	3.8383	1.6684
	Best	2.7025	2.1930	2.4661	1.7229	2.3312	1.5705
	Mean	3.5114	2.6536	2.6326	2.0258	2.6359	1.5795
	S.D.	0.5915	0.3020	0.0713	0.1198	0.2870	0.0241
Wine	Worst	13,616.9408	13,415.0092	14,752.6728	13,362.1693	13,396.0147	13,318.5101
	Best	13,318.6675	13,327.0278	13,320.7178	13,325.9217	13,327.5156	13,318.4817
	Mean	13,407.7736	13,361.6433	13,355.5274	13,340.4952	13,353.1583	13,318.4858
	S.D.	61.9835	18.9552	201.6508	8.3583	18.6450	0.0057

Compared to other clustering approaches, the proposed ECPSO-MTP mostly achieves the best performance, as shown in Table 3. The external index, i.e., Purity [68], is introduced as an evaluation criterion for the clustering results obtained by these approaches, which is defined by (13) in the following [69]:

$$Purity = \frac{1}{M} \sum_{i=1}^K \max_j |D_i \cap C_j|, \tag{13}$$

where  $D_i$  is the  $i$ -th cluster obtained by a clustering approach.  $C_j$  is the  $j$ -th label of the real cluster.  $|D_i \cap C_j|$  is the totality of data points that belong to both  $D_i$  and  $C_j$ . The clustering results of Purity obtained by these clustering approaches from eight datasets are given in Table 4, and overall, ECPSO-MTP has the best performance among the six clustering approaches. Therefore, all these comparison results validate the clustering efficiency of the proposed ECPSO-MTP.

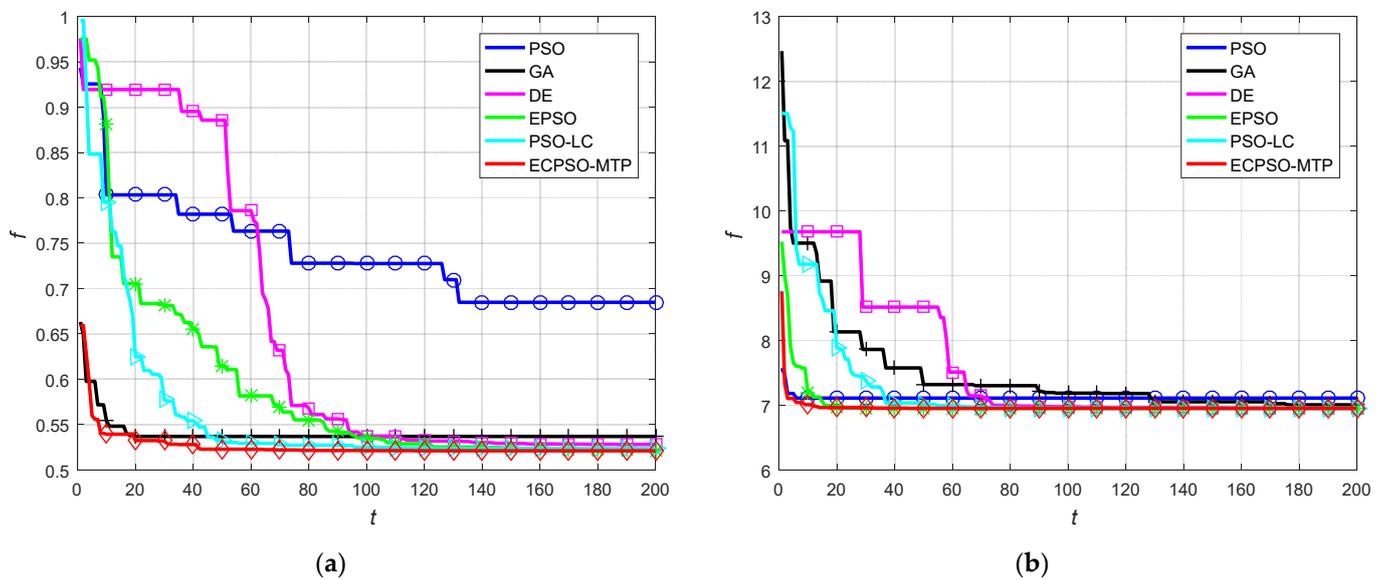
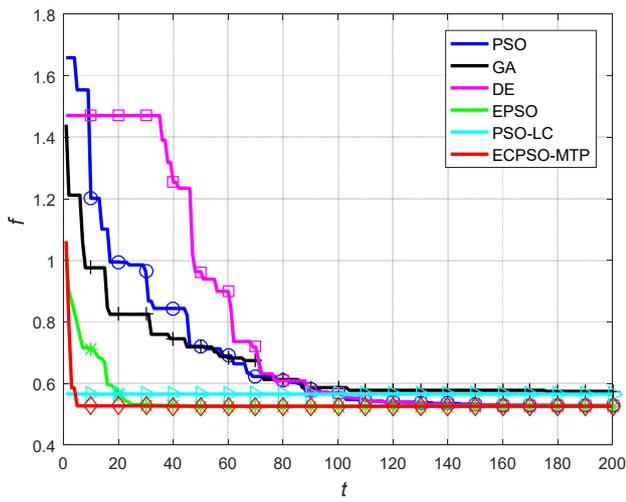
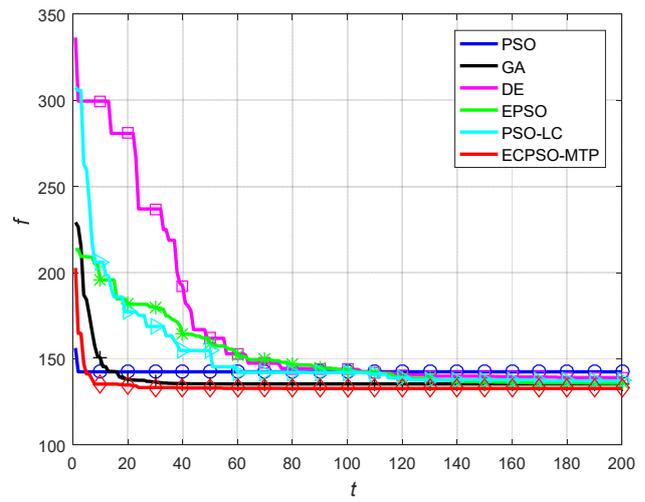


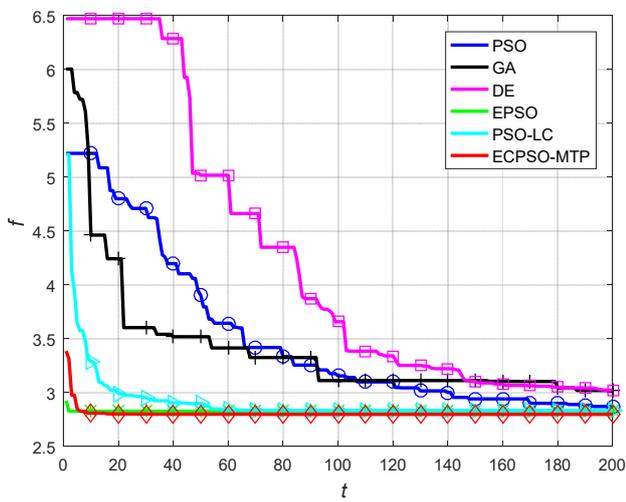
Figure 3. Cont.



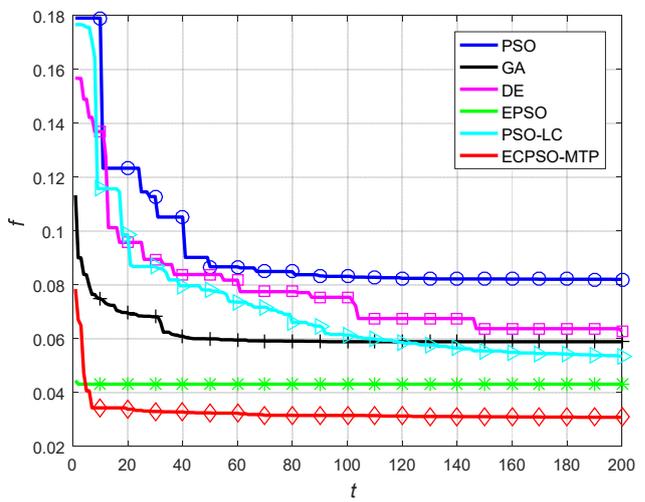
(c)



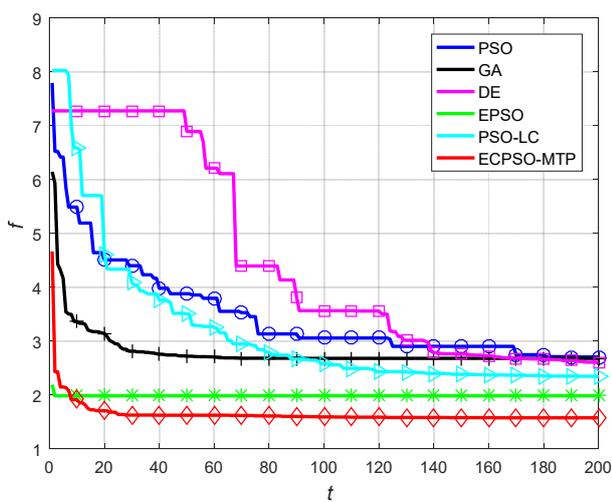
(d)



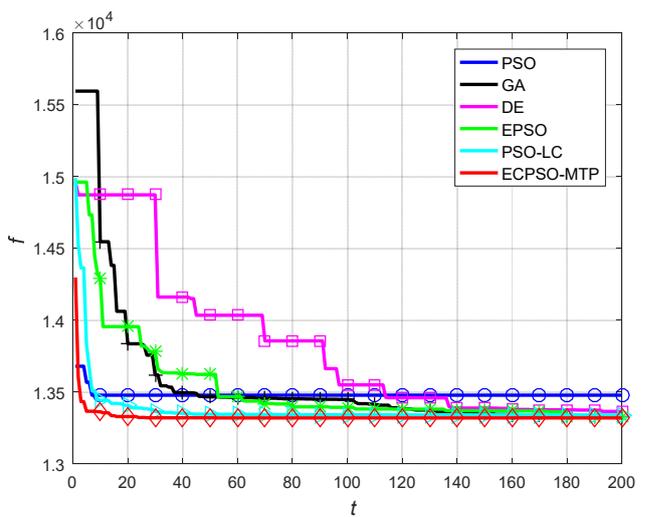
(e)



(f)



(g)



(h)

**Figure 3.** Comparison of convergence results from six test clustering approaches on eight datasets. (a) Data\_9\_2; (b) Square4; (c) Iris; (d) Newthyroid; (e) Seeds; (f) Yeast; (g) Glass; (h) Wine.

**Table 4.** Performance of six clustering approaches on eight datasets (Purity).

Datasets	Statistics	Comparative Approaches					
		PSO	GA	DE	EPSO	PSO-LC	ECPSO-MTP
Data_9_2	Mean	0.8413	0.9128	0.9179	0.9212	0.9144	0.9214
	S.D.	0.0382	0.0078	0.0304	0.0022	0.0201	0.0015
Square4	Mean	0.9313	0.9333	0.9346	0.9350	0.9350	0.9350
	S.D.	0.0033	0.0027	0.0018	$3.36 \times 10^{-16}$	0.0001	$3.36 \times 10^{-16}$
Iris	Mean	0.8771	0.8923	0.8857	0.8907	0.8932	0.8933
	S.D.	0.0551	0.0106	0.0319	0.0056	0.0107	0.0026
Newthyroid	Mean	0.8007	0.8361	0.8536	0.8541	0.8502	0.8605
	S.D.	0.0448	0.0144	0.0273	0.0392	0.0298	$7.85 \times 10^{-16}$
Seeds	Mean	0.8510	0.8879	0.8929	0.8952	0.8947	0.8976
	S.D.	0.0906	0.0077	0.0065	0.0119	0.0084	$7.85 \times 10^{-16}$
Yeast	Mean	0.3494	0.3559	0.3925	0.4385	0.3785	0.5044
	S.D.	0.0260	0.0266	0.0287	0.0235	0.0179	0.0215
Glass	Mean	0.4911	0.4967	0.5185	0.5355	0.4965	0.5877
	S.D.	0.0450	0.0199	0.0154	0.0222	0.0152	0.0062
Wine	Mean	0.7013	0.7016	0.7018	0.7022	0.7022	0.7022
	S.D.	0.0058	0.0022	0.0041	$2.24 \times 10^{-16}$	$2.24 \times 10^{-16}$	$2.24 \times 10^{-16}$

5.3. Friedman Test Statistics

To investigate the statistical significance of the proposed ECPSO-MTP, the Friedman test is employed in this analysis, and the average of MSE obtained by test clustering approaches in comparison experiments is also introduced as the reference object [70]. In this case, the null hypothesis assumes that all test comparative approaches have equal values of MSE in the experiment. More details about the Friedman test in mathematical description are given as follows [71]:

The average MSE obtained by comparative approaches on eight datasets is ranked from smallest to largest. Let  $r_{ij}$  is the rank associated with approach  $j$  on dataset  $i$ , for  $1 \leq i \leq 8, 1 \leq j \leq 6$ . Noted that if  $r_{ij} = 1$ , which represents the lowest value of MSE among these approaches.  $\frac{1}{2}(p + 1)$  represents the average of these ranks, and its value is equal to 3 in this case, where  $p$  is the totality of test clustering approaches, and,  $p = 6$ . The Friedman test statistic is defined by (14) in the following:

$$\chi_r^2 = \frac{12}{np(p + 1)} \sum_{j=1}^p \left( \sum_{i=1}^n r_{ij} \right)^2 - 3n(p + 1), \tag{14}$$

where  $n$  is the totality of datasets, and,  $n = 8$ . The ranks of the average obtained by the test clustering approaches are presented in Table 5. The value of the Friedman test statistic  $\chi_r^2$  is 38.393, as shown in Table 5. At the significance level of  $\alpha = 5\%$ , the null hypothesis is rejected with  $p - 1$  degrees of freedom, and the critical value of  $\chi^2$  is 11.070. Therefore, the proposed ECPSO-MTP is statistically superior to other clustering approaches in terms of the MSE measure.

**Table 5.** Computation results obtained by Friedman test statistics (Mean).

Datasets	Comparative Approaches					
	PSO	GA	DE	EPSO	PSO-LC	ECPSO-MTP
Data_9_2	6	5	4	2	3	1
Square4	6	5	4	1	3	1
Iris	6	5	4	2	3	1
Newthyroid	6	5	4	2	3	1
Seeds	6	5	4	2	3	1

Table 5. Cont.

Datasets	Comparative Approaches					
	PSO	GA	DE	EPSO	PSO-LC	ECPSO-MTP
Yeast	6	5	4	2	3	1
Glass	6	5	3	2	4	1
Wine	6	5	4	2	3	1
Total Rank	48	40	31	15	25	8
Average Rank	6	5	3.875	1.875	3.125	1
Deviation	2.5	1.5	0.375	−1.625	−0.375	−2.5

## 6. The Proposed ECPSO-MTP for Image Segmentation

In this section, comparison experiments with accuracy that were conducted on some test images are introduced to further investigate the clustering performance of the proposed ECPSO-MTP. Eight images from the public image segmentation dataset of BSDS500 are employed as test images for clustering approaches, and three compared approaches, including classic PSO, K-means, and spectral clustering (SC), are adopted to demonstrate the effectiveness of the proposed ECPSO-MTP.

### 6.1. Test Images

In the clustering experiment, eight test images, including Lawn, Agaric, Church, Castle, Elephants, Lane, Starfish, and Pyramid, which have been reported and provided from the Berkeley Segmentation Dataset and Benchmark, are utilized as we have mentioned above [72]. The size of all test images is set to  $481 \times 321$ . And these test images are clearly depicted in Figure 4. It is noted that the number of clusters in the Lawn and Agaric images is set to 2, where  $K = 2$ , the number of clusters of Church, Castle, Elephants, and Lane images is set to 3, where  $K = 3$ , and the number of clusters in the Starfish and Pyramid images is set to 4, where  $K = 4$  [25]. More details about the label information on these images are graphically depicted in Figure 5.

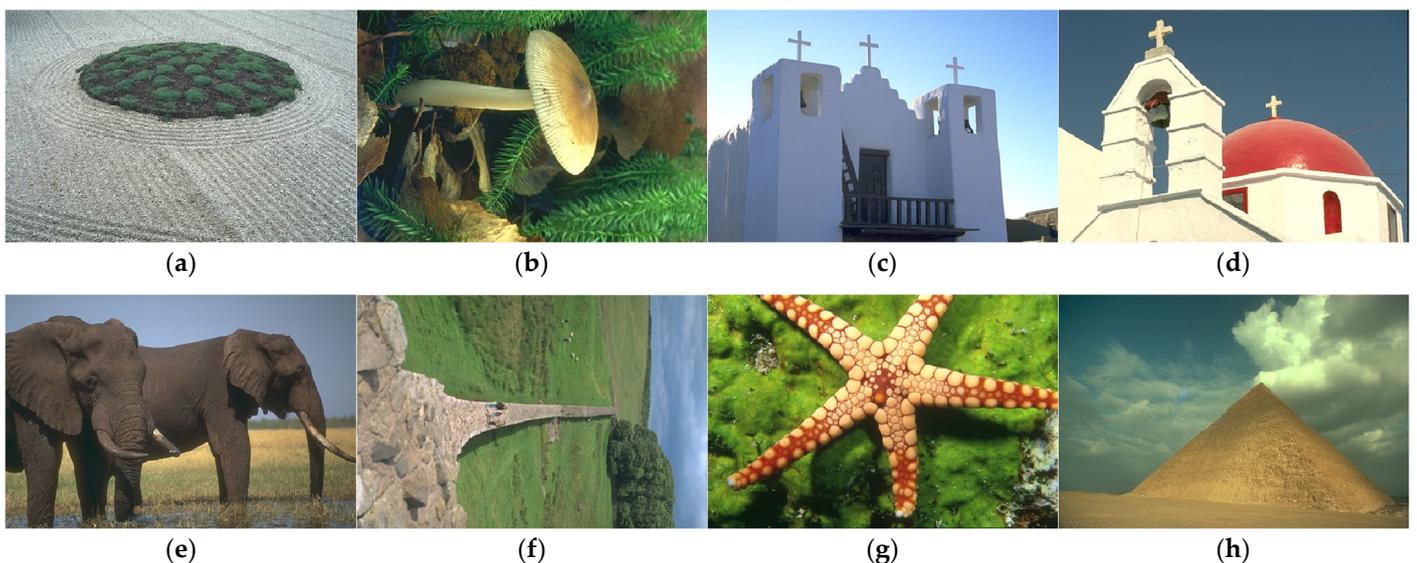
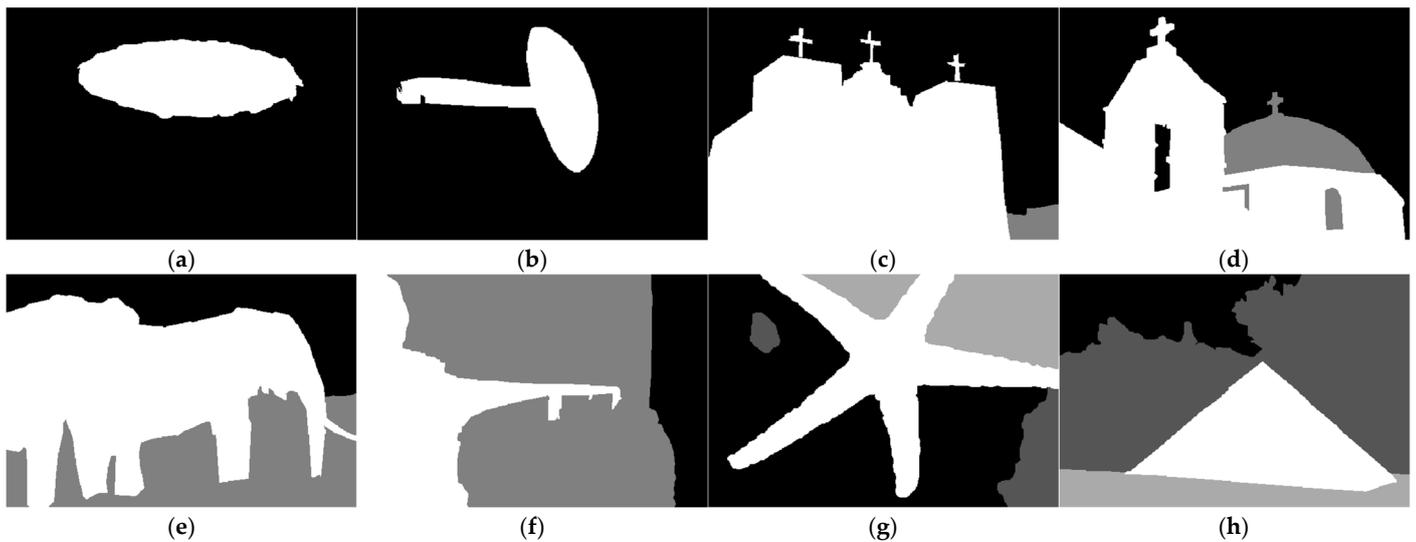


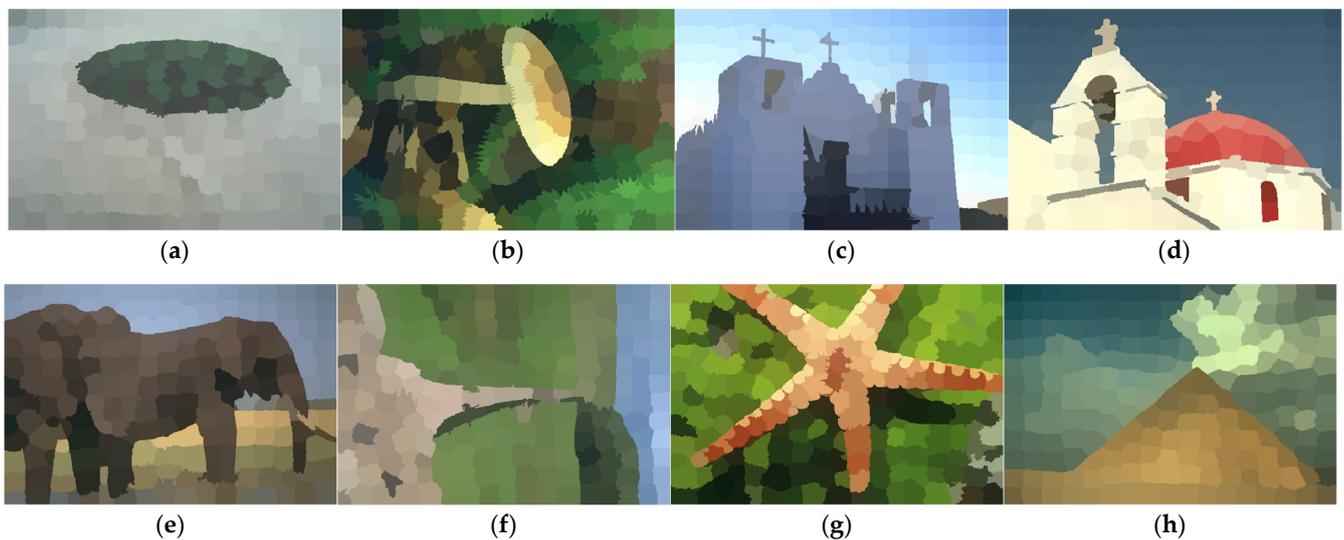
Figure 4. Eight test images. (a) Lawn; (b) Agaric; (c) Church; (d) Castle; (e) Elephants; (f) Lane; (g) Starfish; (h) Pyramid.



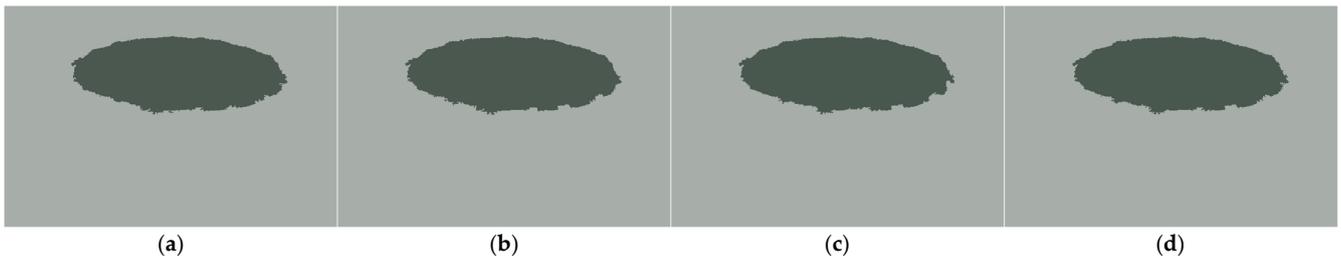
**Figure 5.** The labels of eight test images. (a) Lawn; (b) Agaric; (c) Church; (d) Castle; (e) Elephants; (f) Lane; (g) Starfish; (h) Pyramid.

### 6.2. Comparison with Other Clustering Approaches

In particular, the simple linear iterative clustering (SLIC) technique is first introduced to segment the test images [73] before conducting the clustering experiment. The total number of superpixels is set to 200 for each test image. The segmented images obtained by applying SLIC to the eight test images are given in Figure 6. Then clustering approaches, including the proposed ECPSO-MTP and compared approaches such as K-means, SC, and classic PSO, are employed to cluster the superpixels. The achievable segmentation results of these clustering approaches on the test images are shown in Figures 7–14.



**Figure 6.** Segmentation results obtained by SLIC for eight test images. (a) Lawn; (b) Agaric; (c) Church; (d) Castle; (e) Elephants; (f) Lane; (g) Starfish; (h) Pyramid.



**Figure 7.** Comparison of clustering results on the Lawn image. (a) K-means; (b) SC; (c) PSO; (d) ECPSO-MTP.



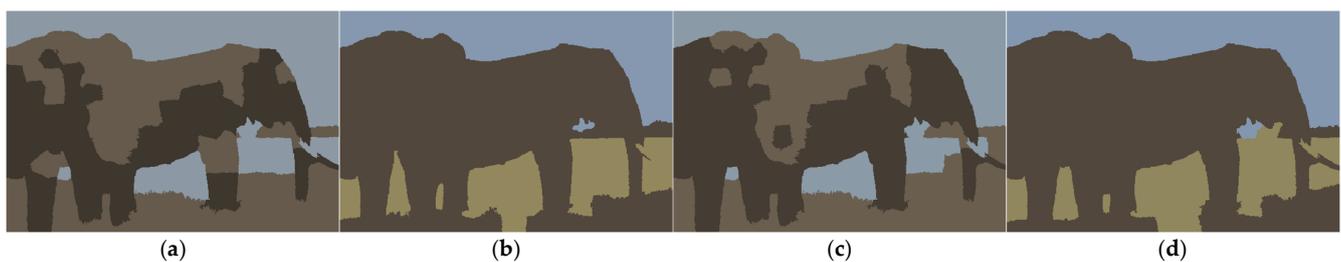
**Figure 8.** Comparison of clustering results on the Agaric image. (a) K-means; (b) SC; (c) PSO; (d) ECPSO-MTP.



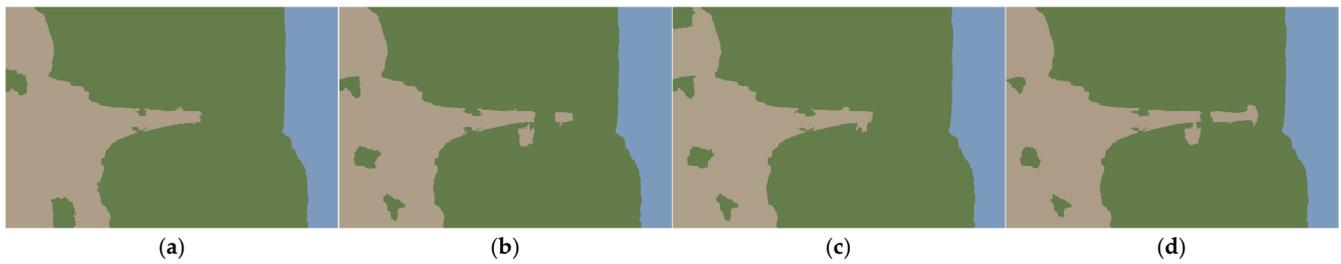
**Figure 9.** Comparison of clustering results on the Church image. (a) K-means; (b) SC; (c) PSO; (d) ECPSO-MTP.



**Figure 10.** Comparison of clustering results on the Castle image. (a) K-means; (b) SC; (c) PSO; (d) ECPSO-MTP.



**Figure 11.** Comparison of clustering results on the Elephants image. (a) K-means; (b) SC; (c) PSO; (d) ECPSO-MTP.



**Figure 12.** Comparison of clustering results on the Lane image. (a) K-means; (b) SC; (c) PSO; (d) ECPSO-MTP.



**Figure 13.** Comparison of clustering results on the Starfish image. (a) K-means; (b) SC; (c) PSO; (d) ECPSO-MTP.



**Figure 14.** Comparison of clustering results on the Pyramid image. (a) K-means; (b) SC; (c) PSO; (d) ECPSO-MTP.

The proposed ECPSO-MTP exhibits the best segmentation quality on mostly test images, as shown in Figures 7–14. Furthermore, each test clustering approach was independently implemented 50 times to reduce the influence of random factors. Simple statistical results of Purity from these clustering approaches are presented in Table 6, which validate the effectiveness of the proposed ECPSO-MTP.

**Table 6.** Performance of four clustering approaches on eight test images (Purity).

Image	Statistics	Comparative Approaches			
		K-Means	SC	PSO	ECPSO-MTP
Lawn	Worst	0.9933	0.9937	0.9934	0.9943
	Best	0.9933	0.9937	0.9934	0.9943
	Mean	0.9933	0.9937	0.9934	0.9943
	S.D.	$2.28 \times 10^{-16}$	$2.34 \times 10^{-16}$	$5.61 \times 10^{-16}$	$3.42 \times 10^{-16}$
Agaric	Worst	0.9477	0.9517	0.9485	0.9567
	Best	0.9477	0.9517	0.9485	0.9567
	Mean	0.9477	0.9517	0.9485	0.9567
	S.D.	$1.24 \times 10^{-16}$	$2.65 \times 10^{-16}$	$4.49 \times 10^{-16}$	$3.16 \times 10^{-16}$
Church	Worst	0.7892	0.8902	0.7876	0.7967
	Best	0.8646	0.8902	0.8903	0.8962
	Mean	0.8593	0.8902	0.8641	0.8903
	S.D.	0.0147	$1.12 \times 10^{-16}$	0.0445	0.0239

Table 6. Cont.

Image	Statistics	Comparative Approaches			
		K-Means	SC	PSO	ECPSO-MTP
Castle	Worst	0.9302	0.9411	0.9327	0.9413
	Best	0.9377	0.9411	0.9377	0.9413
	Mean	0.9346	0.9411	0.9368	0.9413
	S.D.	0.0033	$3.36 \times 10^{-16}$	0.0019	$4.49 \times 10^{-16}$
Elephants	Worst	0.6930	0.8881	0.7669	0.7624
	Best	0.9248	0.8940	0.7820	0.9248
	Mean	0.7802	0.8882	0.7807	0.8978
	S.D.	0.0862	0.0008	0.0029	0.0558
Lane	Worst	0.8205	0.9668	0.8318	0.8241
	Best	0.9730	0.9730	0.9730	0.9795
	Mean	0.9124	0.9676	0.9537	0.9696
	S.D.	0.0668	0.0027	0.0482	0.0671
Starfish	Worst	0.4805	0.5232	0.4910	0.5911
	Best	0.5860	0.7259	0.6287	0.8435
	Mean	0.5597	0.6233	0.5829	0.6344
	S.D.	0.0406	0.0620	0.0183	0.0639
Pyramid	Worst	0.7200	0.7366	0.7497	0.7914
	Best	0.7661	0.8394	0.7594	0.8410
	Mean	0.7533	0.7648	0.7551	0.8354
	S.D.	0.0157	0.0421	0.0038	0.0073

6.3. Friedman Test Statistics

In the Friedman statistical test, the average of Purity obtained by these comparative approaches in the segmentation experiment is used as the preference object. The null hypothesis assumes that all clustering approaches have equal Purity [71]. The ranks of the average of comparative approaches are presented in Table 7. The Friedman test statistic  $\chi_r^2$ , computed using Table 7 according to Equation (14), is 24.000. Therefore, at a significance level of 5%, the null hypothesis is rejected with 3 degrees of freedom due to the critical value of  $\chi^2$  is equal to 7.815. Thus, the proposed ECPSO-MTP is statistically superior to the other clustering approaches in terms of Purity.

Table 7. Computation results of the Friedman test statistics (Mean).

Images	Comparative Approaches			
	K-Means	SC	PSO	ECPSO-MTP
Lawn	4	2	3	1
Agaric	4	2	3	1
Church	4	2	3	1
Castle	4	2	3	1
Elephants	4	2	3	1
Lane	4	2	3	1
Starfish	4	2	3	1
Pyramid	4	2	3	1
Total Rank	32	16	24	8
Average Rank	4	2	3	1
Deviation	1.5	-0.5	0.5	-1.5

7. Conclusions

In this paper, an attractive extended TP combining with MTP and an improved PSO based on environmental factors and a crossover operator is constructed and proposed, referred to as ECPSO-MTP. The proposed ECPSO-MTP aims to establish a highly efficient P system for solving clustering problems that is integrated with the computing framework of

MTP and the evolutionism of PSO. Two types of evolution rules are defined and described in ECPSO-MTP, which are applied to different elementary membranes to evolve objects, including evolution rules with environmental factors and evolution rules with crossover operators. The environmental factor, which uses partitioning information, is randomly allocated to objects to replace global best and participate in the evolution process. The crossover operator is accomplished through the hybridization of the global best of elementary membranes to reject randomness. Information exchange and resource sharing between elementary membranes are accomplished by evolutionary symport with promoters of MTP, including forward and backward communication rules. At last, comparison experiments are conducted on benchmark clustering and image segmentation datasets to validate the clustering efficiency of the proposed ECPSO-MTP.

The distributed and parallel processing of P systems provides an efficient way for solving complex problems with polynomial or linear complexity. However, the application of P systems is limited by computational incompleteness and implementation difficulty. To address these limitations, many variations of P systems have been designed and developed, particularly extended P systems integrated with EC. The proposed ECPSO-MTP is an instance of such an extended P system. The monodirectional connection between elementary membranes in ECPSO-MTP is straightforward and simple to implement. Future studies will explore the combination of this novel variant of P systems with other SI approaches. In addition, reducing the complexity of this extended P system may become an important area of research in the future. Finally, it should be noted that only low-dimensional or small datasets were used for clustering experiments, and the clustering performance of ECPSO-MTP may be limited by high-dimensional or large datasets. Much additional research is necessary to effectively utilize this extended P system for resolving complex clustering problems.

**Author Contributions:** Conceptualization, X.L.; methodology and writing—original draft preparation, L.W. and Q.R.; validation, J.Q.; formal analysis, L.G.; data curation, Y.Z. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The two artificial datasets used in the experiments were provided from the Indian Statistical Institute, Artificial datasets, which can be accessed at: <https://www.isical.ac.in/content/research-data> (accessed on 16 June 2020). The six UCI datasets used in the experiments were provided from the UCI Machine Learning Repository, UCI datasets, which can be accessed at: <http://archive.ics.uci.edu/ml/datasets.php> (accessed on 20 July 2022). The eight segmentation images used in the experiments were provided from the Berkely Computer Vision Group, the Berkely segmentation dataset and benchmark (BSDS500), which can be accessed at: <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/> (accessed on 13 October 2021).

**Acknowledgments:** This research project was supported by the Doctoral Foundation of Shandong Jianzhu University, with the number of Doctoral Foundation is X21008Z.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Păun, G. Membrane computing: An introduction. *Theor. Comput. Sci.* **2002**, *287*, 73–100. [CrossRef]
2. Păun, G. Computing with membranes. *J. Comput. Syst. Sci.* **2000**, *61*, 108–143. [CrossRef]
3. Pan, L.; Zeng, L.; Song, T. *Membrane Computing an Introduction*, 1st ed.; Huazhong University of Science and Technology Press: Wuhan, China, 2012; pp. 1–10.
4. Păun, G. Membrane computing and economics: A General View. *Int. J. Comput. Commun. Control.* **2016**, *11*, 105–112. [CrossRef]
5. Freund, R.; Păun, G.; Mario, J. Tissue P systems with channel states. *Theor. Comput. Sci.* **2005**, *330*, 101–116. [CrossRef]
6. Li, Y.Y.; Song, B.S.; Zeng, X.X. Rule synchronization for monodirectional tissue-like P systems with channel states. *Inf. Comput.* **2022**, *285*, 104895. [CrossRef]
7. Păun, A.; Păun, G. The power of communication P systems with symport/antiport. *New Gener. Comput.* **2002**, *20*, 295–305. [CrossRef]

8. Jiang, Z.N.; Liu, X.Y.; Zang, W.K. A kernel-based intuitionistic weight fuzzy k-modes algorithm using coupled chained P system combines DNA genetic rules for categorical data. *Neurocomputing* **2023**, *528*, 84–96. [\[CrossRef\]](#)
9. Song, B.; Hu, Y.; Adorna, H.N.; Xu, F. A quick survey of tissue-like P systems. *Rom. J. Inf. Sci. Technol.* **2018**, *21*, 310–321.
10. Kujur, S.S.; Sahana, S.K. Medical image registration utilizing tissue P systems. *Front. Pharmacol.* **2022**, *13*, 949872. [\[CrossRef\]](#)
11. Martin, O.D.; Cabrera, V.L.; Mario, J.P. The environment as a frontier of efficiency in tissue P systems with communication rules. *Theor. Comput. Sci.* **2023**, *956*, 113812. [\[CrossRef\]](#)
12. Pan, L.Q.; Song, B.S.; Zandron, C. On the computational efficiency of tissue P systems with evolutionary symport/antiport rules. *Knowl. Based Syst.* **2023**, *262*, 110266. [\[CrossRef\]](#)
13. Luo, Y.; Guo, P.; Jiang, Y.; Zhang, Y. Timed homeostasis tissue-Like P systems with evolutionary symport/antiport rules. *IEEE Access* **2020**, *8*, 131414–131424. [\[CrossRef\]](#)
14. Orellana-Martín, D.; Valencia-Cabrera, L.; Song, B.; Pan, L.; Pérez-Jiménez, M.J. Tissue P systems with evolutionary communication rules with two objects in the left-hand side. *Nat. Comput.* **2023**, *22*, 119–132. [\[CrossRef\]](#)
15. Song, B.S.; Pan, L.Q. The computational power of tissue-like P systems with promoters. *Theor. Comput. Sci.* **2016**, *641*, 43–52. [\[CrossRef\]](#)
16. Song, B.; Zeng, X.; Jiang, M.; Pérez-Jiménez, M.J. Monodirectional tissue P systems with promoters. *IEEE Trans. Cybern.* **2021**, *51*, 438–450. [\[CrossRef\]](#)
17. Song, B.S.; Zeng, X.X.; Paton, R.A. Monodirectional tissue P systems with channel states. *Inf. Sci.* **2021**, *546*, 206–219. [\[CrossRef\]](#)
18. Song, B.S.; Li, K.L.; Zeng, X.X. Monodirectional evolutionary symport tissue P systems with promoters and cell division. *IEEE Trans. Parallel Distrib. Syst.* **2022**, *33*, 332–342. [\[CrossRef\]](#)
19. Song, B.S.; Li, K.L.; Zeng, X.X. Monodirectional evolutionary symport tissue P systems with channel states and cell division. *Sci. China-Inf. Sci.* **2023**, *66*, 139104. [\[CrossRef\]](#)
20. Zhang, G.; Jiménez, M.; Gheorghe, G. *Real-Life Applications with Membrane Computing*, 1st ed.; Springer Press: Berlin/Heidelberg, Germany, 2017; pp. 11–12.
21. Tian, X.; Liu, X.Y. Improved hybrid heuristic algorithm inspired by tissue-like membrane system to solve job shop scheduling problem. *Processes* **2021**, *20*, 219. [\[CrossRef\]](#)
22. Zhang, G.X.; Chen, J.X.; Gheorghe, M. QA hybrid approach based on differential evolution and tissue membrane systems for solving constrained manufacturing parameter optimization problems. *Appl. Soft Comput.* **2013**, *13*, 1528–1542. [\[CrossRef\]](#)
23. Peng, H.; Shi, P.; Wang, J.; Riscos-Núñez, A.; Pérez-Jiménez, M.J. Multiobjective fuzzy clustering approach based on tissue-like membrane systems. *Knowl. Based Syst.* **2017**, *125*, 74–82. [\[CrossRef\]](#)
24. Lagos-Eulogio, P.; Seck-Tuoh-Mora, J.C.; Hernandez-Romero, N.; Medina-Marin, J. A new design method for adaptive IIR system identification using hybrid CPSO and DE. *Nonlinear Dyn.* **2017**, *88*, 2371–2389. [\[CrossRef\]](#)
25. Wang, L.; Liu, X.; Qu, J.; Zhao, Y.; Jiang, Z.; Wang, N. An extended tissue-like P System based on membrane systems and quantum-behaved particle swarm optimization for image segmentation. *Processes* **2022**, *10*, 287. [\[CrossRef\]](#)
26. Luo, Y.G.; Guo, P.; Zhang, M.Z. A framework of ant colony P system. *IEEE Access* **2019**, *7*, 157655–157666. [\[CrossRef\]](#)
27. Peng, H.; Wang, J. A hybrid approach based on tissue P systems and artificial bee colony for IIR system identification. *Neural Comput. Appl.* **2017**, *28*, 2675–2685. [\[CrossRef\]](#)
28. Chen, H.J.; Liu, X.Y. An improved multi-view spectral clustering based on tissue-like P systems. *Sci. Rep.* **2022**, *12*, 18616. [\[CrossRef\]](#)
29. Sharif, E.A.; Agoyi, M. Using tissue-like P system to solve the nurse rostering problem at the medical centre of the national university of malaysia. *Appl. Nanosci.* **2022**, *2022*, 3145. [\[CrossRef\]](#)
30. Issac, T.; Silas, S.; Rajsingh, E.B. Investigative prototyping a tissue P system for solving distributed task assignment problem in heterogeneous wireless sensor network. *J. King Saud Univ. -Comput. Inf. Sci.* **2022**, *34*, 3685–3702. [\[CrossRef\]](#)
31. Chen, H.J.; Liu, X.Y. Reweighted multi-view clustering with tissue-like P system. *PLoS ONE* **2023**, *18*, e269878. [\[CrossRef\]](#)
32. Kennedy, J.; Eberhart, R. Particle swarm optimization. In Proceedings of the IEEE International Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995; pp. 1942–1948.
33. Kassoul, K.; Zufferey, N.; Cheikhrouhou, N.; Belhaouari, S.B. Exponential particle swarm optimization for global optimization. *IEEE Access* **2022**, *10*, 78320–78344. [\[CrossRef\]](#)
34. Wang, D.S.; Tan, D.P.; Liu, L. Particle swarm optimization algorithm: An overview. *Soft Comput.* **2017**, *22*, 387–408. [\[CrossRef\]](#)
35. Bi, J.X.; Zhao, M.O.; Chai, D.S. PSOSVRPos: WiFi indoor positioning using SVR optimized by PSO. *Expert Syst. Appl.* **2023**, *222*, 119778. [\[CrossRef\]](#)
36. Anbarasi, M.P.; Kanthalakshmi, S. Power maximization in standalone photovoltaic system: An adaptive PSO approach. *Soft Comput.* **2023**, *27*, 8223–8232. [\[CrossRef\]](#)
37. Peng, J.; Li, Y.; Kang, H.; Shen, Y.; Sun, X.; Chen, Q. Impact of population topology on particle swarm optimization and its variants: An information propagation perspective. *Swarm Evol. Comput.* **2022**, *69*, 100990. [\[CrossRef\]](#)
38. Harrison, K.; Engelbrecht, A.; Berman, B. Self-adaptive particle swarm optimization: A review and analysis of convergence. *Swarm Intell.* **2018**, *12*, 187–226. [\[CrossRef\]](#)
39. Huang, C.; Zhou, X.; Ran, X.; Liu, Y.; Deng, W.; Deng, W. Co-evolutionary competitive swarm optimizer with three-phase for large-scale complex optimization problem. *Inf. Sci.* **2023**, *619*, 2–18. [\[CrossRef\]](#)

40. Chen, Z.; Wang, Y.; Chan, T.H.; Li, X.; Zhao, S. A particle swarm optimization algorithm with sigmoid increasing inertia weight for structural damage identification. *Appl. Sci. Basel* **2022**, *12*, 3429. [[CrossRef](#)]
41. Wang, J.; Wang, X.; Li, X.; Yi, J. A hybrid particle swarm optimization algorithm with dynamic adjustment of inertia weight based on a new feature selection method to optimize SVM parameters. *Entropy* **2023**, *25*, 531. [[CrossRef](#)] [[PubMed](#)]
42. Duan, Y.; Chen, N.; Chang, L.; Ni, Y.; Kumar, S.V.N.S.; Zhang, P. CAPSO: Chaos adaptive particle swarm optimization algorithm. *IEEE Access* **2022**, *10*, 29393–29405. [[CrossRef](#)]
43. Molaei, S.; Moazen, H.; Najjar-Ghabel, S.; Farzinvas, L. Particle swarm optimization with an enhanced learning strategy and crossover operator. *Knowl. Based Syst.* **2021**, *215*, 106768. [[CrossRef](#)]
44. Pan, L.Q.; Zhao, Y.; Li, L.H. Neighborhood-based particle swarm optimization with discrete crossover for nonlinear equation systems. *Swarm Evol. Comput.* **2022**, *69*, 101019. [[CrossRef](#)]
45. Das, P.K.; Jena, P.K. Multi-robot path planning using improved particle swarm optimization algorithm through novel evolutionary operators. *Appl. Soft Comput.* **2020**, *92*, 106312. [[CrossRef](#)]
46. Pu, H.; Song, T.; Schonfeld, P.; Li, W.; Zhang, H.; Hu, J.; Peng, X.; Wang, J. Mountain railway alignment optimization using stepwise & hybrid particle swarm optimization incorporating genetic operators. *Appl. Soft Comput.* **2019**, *78*, 41–57.
47. Gu, Q.; Wang, Q.; Chen, L.; Li, X.; Li, X. A dynamic neighborhood balancing-based multi-objective particle swarm optimization for multi-modal problems. *Expert Syst. Appl.* **2022**, *205*, 117313. [[CrossRef](#)]
48. Moazen, H.; Molaei, S.; Farzinvas, L.; Sabaei, M. PSO-ELPM: PSO with elite learning, enhanced parameter updating, and exponential mutation operator. *Inf. Sci.* **2023**, *628*, 70–91. [[CrossRef](#)]
49. Lynn, N.; Suganthan, P.N. Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. *Swarm Evol. Comput.* **2015**, *24*, 11–24. [[CrossRef](#)]
50. Cao, Y.L.; Zhang, H. Comprehensive learning particle swarm optimization algorithm with local search for multimodal functions. *IEEE Trans. Evol. Comput.* **2019**, *23*, 718–731. [[CrossRef](#)]
51. Wang, F.; Zhang, H.; Li, K.; Lin, Z.; Yang, J.; Shen, X.-L. A hybrid particle swarm optimization algorithm using adaptive learning strategy. *Inf. Sci.* **2018**, *436*, 162–177. [[CrossRef](#)]
52. Zhou, X.; Zhou, S.; Han, Y.; Zhu, S. Levy flight-based inverse adaptive comprehensive learning particle swarm optimization. *Math. Biosci. Eng.* **2022**, *19*, 5241–5268. [[CrossRef](#)]
53. Ge, Q.; Guo, C.; Jiang, H.; Lu, Z.; Yao, G.; Zhang, J.; Hua, Q. Industrial power load forecasting method based on reinforcement learning and PSO-LSSVM. *IEEE Trans. Cybern.* **2022**, *52*, 1112–1124. [[CrossRef](#)]
54. Li, W.; Liang, P.; Sun, B.; Sun, Y.; Huang, Y. Reinforcement learning-based particle swarm optimization with neighborhood differential mutation strategy. *Swarm Evol. Comput.* **2023**, *78*, 101274. [[CrossRef](#)]
55. Lu, L.W.; Zhang, J.; Sheng, J.A. Enhanced multi-swarm cooperative particle swarm optimizer. *Swarm Evol. Comput.* **2022**, *69*, 100989. [[CrossRef](#)]
56. Li, D.; Wang, L.; Guo, W.; Zhang, M.; Hu, B.; Wu, Q. A particle swarm optimizer with dynamic balance of convergence and diversity for large-scale optimization. *Appl. Soft Comput.* **2023**, *32*, 109852. [[CrossRef](#)]
57. Zhang, Y.; Gong, D.W.; Ding, Z.H. Handling multi-objective optimization problems with a multi-swarm cooperative particle swarm optimizer. *Expert Syst. Appl.* **2011**, *38*, 13933–13941. [[CrossRef](#)]
58. Li, T.; Shi, J.; Deng, W.; Hu, Z. Pyramid particle swarm optimization with novel strategies of competition and cooperation. *Appl. Soft Comput.* **2022**, *121*, 108731. [[CrossRef](#)]
59. Song, W.; Ma, W.; Qiao, G. Particle swarm optimization algorithm with environmental factors for clustering analysis. *Soft Comput.* **2017**, *21*, 283–293. [[CrossRef](#)]
60. Nenavath, H.; Jatoth, E.K. Hybridizing sine cosine algorithm with differential evolution for global optimization and object tracking. *Appl. Soft Comput.* **2018**, *62*, 1019–1043. [[CrossRef](#)]
61. Pozna, C.; Precup, R.-E.; Horvath, E.; Petriu, E.M. Hybrid particle filter-particle swarm optimization algorithm and application to fuzzy controlled servo systems. *IEEE Trans. Fuzzy Syst.* **2022**, *30*, 4286–4297. [[CrossRef](#)]
62. Song, B.S.; Zhang, C.; Pan, L.Q. Tissue-like P systems with evolutionary symport/antiport rules. *Inf. Sci.* **2017**, *378*, 177–193. [[CrossRef](#)]
63. Pan, T.; Xu, J.; Jiang, S.; Xu, F. Cell-like spiking neural P systems with evolution rules. *Soft Comput.* **2019**, *23*, 5401–5409. [[CrossRef](#)]
64. Artificial Datasets. Available online: <https://www.isical.ac.in/content/research-data> (accessed on 16 June 2020).
65. UCI Repository of Machine Learning Databases. Available online: <http://archive.ics.uci.edu/ml/datasets.php> (accessed on 20 July 2022).
66. Peng, H.; Wang, J.; Shi, P.; Pérez-Jiménez, M.J.; Riscos-Núñez, A. An extended membrane system with active membranes to solve automatic fuzzy clustering problems. *Int. J. Neural Syst.* **2016**, *26*, 1650004. [[CrossRef](#)]
67. Malinen, M.I.; Franti, P. Clustering by analytic functions. *Inf. Sci.* **2012**, *217*, 31–38. [[CrossRef](#)]
68. Tang, Y.; Huang, J.; Pedrycz, W.; Li, B.; Ren, F. A fuzzy cluster validity index induced by triple center relation. *IEEE Trans. Cybern.* **2023**, *99*, 1–13. [[CrossRef](#)] [[PubMed](#)]
69. Ge, F.H.; Liu, X.Y. Density peaks clustering algorithm based on a divergence distance and tissue-Like P System. *Appl. Sci. Basel* **2023**, *13*, 2293. [[CrossRef](#)]
70. Kumari, B.; Kumar, S. Chaotic gradient artificial bee colony for text clustering. *Soft Comput.* **2016**, *20*, 1113–1126.

71. Friedman, M. The use of Ranks to avoid the assumption of normality implicit in the analysis of variance. *Publ. Am. Stat. Assoc.* **1937**, *32*, 675–701. [[CrossRef](#)]
72. The Berkeley Segmentation Dataset and Benchmark. Available online: <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/grouping/segbench/> (accessed on 13 October 2021).
73. Achanta, R.; Shaji, A.; Smith, K.; Lucchi, A.; Fua, P.; Süsstrunk, S. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2274–2282. [[CrossRef](#)]

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.