

A Particle Swarm Optimization Based on P systems

Fen Zhou, Gexiang Zhang, Haina Rong
School of Electrical Engineering
Southwest Jiaotong University
Chengdu, 610031, P.R. China

Jixiang Cheng
School of Information Science & Technology
Southwest Jiaotong University
Chengdu, 610031, P.R. China

Marian Gheorghe
Department of Computer Science University of Sheffield
Sheffield S1 4DP, UK and Department of Computer
Science, University of Pitesti
Str. Targu din Vale 1, 110040 Pitesti, Romania

Florentin Ipatu, Raluca Lefticaru
Department of Computer Science, University of Pitesti
Str. Targu din Vale 1, 110040 Pitesti, Romania

Abstract—This paper presents a novel membrane algorithm, called particle swarm optimization based on P systems (PSOPS), which combines P systems and particle swarm optimization. The PSOPS uses the representation of individuals, evolutionary rules of particle swarm optimization, and a hierarchical membrane structure and transformation or communication-like rules in P systems to design its algorithm. Experiments conducted on seven bench function optimization problems and time-frequency atom decomposition demonstrate the effectiveness and practicality of the introduced method.

I. INTRODUCTION

Membrane computing, which describes models called P systems, was introduced in [1] as a new distributed parallel computing model abstracted from the functionality and structure of tissues or organs of living cells. A P system consists of a hierarchical structure composed of several membranes, objects placed inside membranes, and specific evolution rules applied to objects. Using multiple rules which are applied in a nondeterministic and maximally parallel manner, the initial multisets of objects may evolve in each membrane [1-2]. The significant features of P systems include evolutionary rules such as transformation and communication, intrinsic parallelism, and non-deterministic behavior.

Both P systems and evolutionary algorithms (EAs) are nature-inspired computation models. Both of them can solve NP-hard problems. Because of the suitable formal framework for parallel-distributed computation of P systems and the extensive applications of EAs in solving numerous optimization problems, it is possible to combine EAs with P systems [3]. Some researches on combining P systems with EAs have been done in recent years. In [4], a hybrid algorithm combining P systems and genetic algorithms was presented to solve numerical optimization problems. In [5], the similarities between distributed EAs and P systems were analyzed and new variants of distributed EAs are suggested and applied for some continuous optimization problems. In [3], Zhang et al. presented an evolutionary algorithm, QEPS, combining quantum-inspired evolutionary algorithms with P systems to solve a class of well-known combinatorial optimization problems, knapsack problems. In the QEPS, a novel membrane structure, called one level membrane structure (OLMS) was proposed. The QEPS was successfully applied to solve

satisfiability problems, also a class of well-known combinatorial optimization problems [6]. In [7-9], several variants of the QEPS were introduced to solve time-frequency atom decomposition.

In this paper, a novel membrane algorithm, called particle swarm optimization algorithm based on P systems (PSOPS), is presented. This is the first attempt to discuss the interaction between particle swarm optimization and P systems. Based on the concepts and principles of particle swarm optimization and P systems, the PSOPS employs the evolutionary rules of particle swarm optimization, the OLMS and transformation or communication-like rules in P systems. To verify the effectiveness and applicability of the PSOPS, a large number of experiments are carried out on function optimization problems and time-frequency atom decomposition for linear-frequency modulated (LFM) radar emitter signals. Experimental results show that PSOPS performs better than its counterpart particle swarm optimization algorithm (PSO).

II. PSOPS

This section commences presenting PSO and P systems briefly, and then illustrates the PSOPS in detail.

A. PSO

Particle swarm optimization algorithm, simply called PSO, is an optimization algorithm developed by Kennedy and Eberhart [10]. In the particle swarm optimization, each particle represents a possible solution to the optimization task and accelerates in the direction of its own personal best solution $pBest$ found so far, as well as in the direction of the global best position $gBest$ discovered so far by any of the particles in the swarm. Let X denote the current position in the search space, V is the current velocity. Each particle in the swarm is updated as following:

$$V(t+1) = w * V(t) + c_1 * rand * (pBest(t) - X(t)) + c_2 * rand * (gBest(t) - X(t)) \quad (1)$$

$$X(t+1) = X(t) + V(t+1) \quad (2)$$

Where, $rand \sim U(0,1)$ is the element from uniform random

sequence in the range (0,1), c_1 and c_2 are the acceleration coefficients, usually $c_1 = c_2 = 2.05 \cdot w$, $0.1 \leq w \leq 0.9$ is the weight coefficient. The value of V can be clamped to the range $[-V_{\max}, V_{\max}]$ to ensure particles in the search space [11]. The pseudocode algorithm of a PSO is shown in Fig. 1.

```

Begin
   $t \leftarrow 1$ 
  (i) Initialize  $X(t), V(t)$ 
  While (not termination condition) do
    For  $i=1:n$ 
      (ii) Store the best position  $pBest$  obtained by  $P_i$  so far
      (iii) Store the best position  $gBest$  obtained by all  $P_i$  so far
      (iv) Update particle's velocity  $V(t)$ 
      (v) Update particle's position  $X(t)$ 
    End
   $t \leftarrow t + 1$ 
End
  Store the best particle among  $X(t)$ 
End

```

Figure 1. Pseudocode algorithm of a PSO [12]

B. P systems

A membrane system, also known as a P system, is a new class of distributed and parallel computing devices inspired by the structure and functionality of living cells. The membrane structure of a P system is a hierarchical arrangement of membranes embedded in a main membrane called the skin membrane. Each membrane defines a region, containing some objects and a set of transformation and communication rules [2]. The multisets of objects evolve and move from a region to a neighboring one by applying the rules in a nondeterministic and maximally parallel way. The result of the computation is obtained in the output membrane or emit from the skin membrane [3]. The membrane structure of a cell-like P system is shown in Fig. 2, which can be formalized as follows [1-2]:

$$\Pi = (O, T, \mu, w_1, \dots, w_m, R_1, \dots, R_m, i_0)$$

Where

- (i) O is an alphabet of objects.
- (ii) $T \subseteq O$ (the output alphabet).
- (iii) μ is a membrane structure with m membranes and the regions labeled by the elements of a given set H . m is called the degree of Π .
- (iv) $w_i, 1 \leq i \leq m$, are strings which represent multisets over O associated with the regions $1, 2, \dots, m$ of μ .
- (v) i_0 is the output membrane.

C. PSOPS

This section uses the OLMS to specify the PSOPS algorithm. Some basic parts of P systems are used in this framework, and others will be used in a metaphoric way. A specific membrane

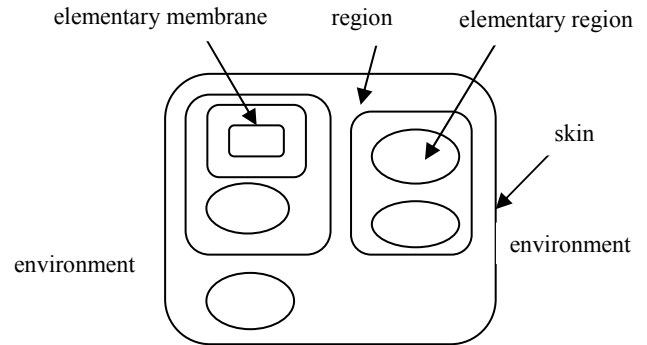


Figure 2. Basic ingredients of a P system [2]

structure will be initially specified at the beginning, but it will be changed later. The objects employed will be organized in multisets. The evolutionary rules of PSO, as well as communication-like rules of P systems will be responsible to evolve the system and select the best individuals [3].

The pseudocode algorithm of the PSOPS is shown in Fig. 3, where each step is described in detail as follows:

- (i) A membrane structure $[_0[_1[_1], [_2[_2], [_3[_3], \dots, [_m[_m]]_0]$ with m regions contained in the skin membrane, denoted by 0.
- (ii) A particle swarm population with n particles is initialized; All the particles are assigned randomly to each membrane without duplication. Each membrane includes one particle at least. Multisets are initialized as follows:

$$w_0 = \lambda,$$

$$w_1 = q_1 q_2 q_3 \dots q_{n_1}, \quad n_1 < n,$$

$$w_2 = q_{n_1+1} q_{n_1+2} \dots q_{n_2}, \quad n_1 + n_2 < n,$$

$$\dots \dots$$

$$w_m = q_{n_{(m-1)+1}} q_{n_{(m-1)+2}} \dots q_{n_m}, \quad n_1 + n_2 + \dots + n_m \leq n,$$
 where $q_i, 1 \leq i \leq n$, is an individual,
- (iii) Each of the compartments 1 to m implements the evolution rules of PSO independently (please see the description of PSO).
- (iv) Communication rules which send the best fit representation generated by the best particle into the skin membrane and the $pbest$ and $gbest$ of the best particle corresponding with the best fit representation from the skin to each compartment to affect the next generation update of all individuals in each region.

III. EXPERIMENTAL RESULTS

a) Parameters Setting

This section discusses the parameters setting in the PSOPS. The parameters include the number of elementary membrane m , the maximal number of generations $g_i, 1 \leq i \leq m$, for the i th elementary membrane.

Begin
$t \leftarrow 1$
(i) Initialize membrane structure
(ii) Allocate individuals for each elementary membrane
While (not termination condition) do
For $i = 1 : m$
(iii) Perform PSO in the i th elementary membrane
End
(iv) Execute communication rules
$t \leftarrow t + 1$
End
End

Figure 3. Pseudocode algorithm of PSOPS

At first, seven bench functions conducting experiments are shown in Table I. f_6 and f_7 are chosen to conduct experiments as delegates. In order to investigate the effect of the parameter m on the PSOPS performance, the population size is set to 30. The subpopulation size in the i th membrane n_i , $1 \leq i \leq m$, is set to a uniformly random integers ranged from 1 to $30-m+1$ on condition that the sum of n_1, n_2, \dots, n_m is 30. The evolutionary generation of each elementary membrane g_i , $1 \leq i \leq m$ is set to a uniformly random integers ranged from 1 to 40. On the above conditions, the number of elementary membranes m grows gradually from 2 to 30. The maximal number of generations is set to 1000 as the termination criterion. The mean fitness values of f_6 and f_7 over 30 runs are shown in Fig. 4. As shown in Fig. 4, the mean values of f_6 and f_7 have many changes in different

number of elementary membranes, but both attain the optimal results with 16 elementary membranes. So, the parameter m could be assigned as 16 in the following experiments.

To investigate the effect of the evolutionary generation g_i , $1 \leq i \leq m$ on the PSOPS performances, parameters are set as follows: population size 30, the number of elementary membranes 16. The maximal value of g_i varies from 5 to 200. The number 30000 of function evaluations is used as the stopping condition. The mean fitness values of f_6 and f_7 over 30 runs are shown in Fig.5. As shown in Fig.5, the mean values of f_6 and f_7 have many changes in different evolutionary generations, but both attain the optimal results when the parameter g_i , $1 \leq i \leq m$, is 5. So, the parameter g_i could be assigned as 5 in the following experiments.

A. Comparative Experiments

In order to demonstrate the effectiveness of the proposed PSOPS algorithm, a large number of experiments are carried out on the seven bench functions. A comparison between the PSOPS and PSO is also drawn. In the experiments, the size of PSOPS and PSO is set to 30. The maximal number of iterations for $f_1 \sim f_7$ is set to 1000. The worst, best and mean fitness values over 30 runs are listed in Table II.

The experimental results in Table II show that PSOPS obtains significantly better solutions than PSO for the seven functions and consume less time relatively in general. Because the novel algorithm, PSOPS, employs the formal framework and communication rules, it can avoid the premature convergence to a considerable degree, and hence has faster convergence rate. As for f_1, f_2, f_4, f_5 and f_6 , PSOPS obtains the optimal or close-to-optimal solutions.

Table I. THE MAIN FEATURES OF SEVEN BENCH FUNCTIONS

Functions	Dimension	Type	Range	Optimum
$f_1 = \frac{1}{4000} \sum_{i=1}^D x_i^2 - \prod_{i=1}^D (\cos(\frac{x_i}{\sqrt{i}})) + 1$	30	minimization	[-600,600]	0
$f_2 = \sum_{i=1}^D x_i^2$	30	minimization	[-5.12,5.12]	0
$f_3 = \sum_{i=1}^D (100(x_{i+1} - x_i^2)^2) + (1 - x_i)^2$	30	minimization	[-5.12,5.12]	0
$f_4 = -c_1 \times \exp(-c_2 \sqrt{\frac{1}{l} \times \sum_{i=1}^D x_i^2}) - \exp(\frac{1}{l} \sum_{i=1}^D \cos(c_3 x_i)) + c_1 + e$ $c_1 = 20, c_2 = 0.2, c_3 = 2\pi$	30	minimization	[-32.768, 32.768]	0
$f_5 = \sum_{i=1}^D [x_i + 0.5]^2$	30	minimization	[-100,100]	0
$f_6 = -\sum_{i=1}^D [\sin(x_i) + \sin(\frac{2x_i}{3})]$	30	Maximization	[3,13]	1.21598D
$f_7 = \frac{\sin^2(\sqrt{x_1^2 + x_2^2}) - 0.5}{(1 + 0.001(x_1^2 + x_2^2))^2} - 0.5$	2	minimization	[-100,100]	-1

IV. APPLICATION EXAMPLE

Time-frequency atom decomposition (TFAD) is an approach to decompose any signal into a linear expansion of waveforms selected from a redundant dictionary of time-frequency atoms [13]. It has become an important analysis technique in non-stationary signal analysis. However, the extremely high computational complexity greatly blocks its practical applications [14]. In this section, we use PSOPS to reduce the computational load and improve the signal representation of TFAD. The chirplet atom, defined as in (3), is chosen as the time-frequency atom. For the sake of simplicity, we name our method as Chirp-PSOPS.

$$g_r(t) = \frac{1}{\sqrt{s}} g\left(\frac{t-u}{s}\right) \exp\left[i\left(\xi(t-u) + \frac{c}{2}(t-u)^2\right)\right] \quad (3)$$

where the index $r = (s, u, \xi, c)$ is a set of parameters and s, u, ξ, c are scale, time center, frequency center and FM slope, respectively. They are discretized as follows: $r = (a^j, pa^j \Delta u, ka^{-j} \Delta \xi, la^{-2j} \Delta c)$, $a = 2$, $\Delta u = 1/2$, $\Delta \xi = \pi$, $\Delta c = \pi$, $0 < j \leq \log_2 N$, $0 \leq p \leq N2^{-j+1}$, $0 \leq k < 2^{j+1}$, $0 \leq l < 2^{j+1}$, $(j, p, k, l) \in Z^4$, where N is the length of the signal f .

In order to verify the effectiveness of Chirp-PSOPS, a LFM radar emitter signal with 10dB signal to noise ratio (SNR) is used to conduct our experiment. The radar emitter signal in time domain is shown in Fig. 6. Besides, a comparison is made with PSO based TFAD, named Chirp-PSO. The parameters are set as follows: in Chirplet-PSO, population size is set to 20 and the maximal number of iterations is set to 200; in Chirp-PSOPS, population size is set to 20, and the number of elementary membranes is set to 12, and the maximal number of generations for the skin membrane is set to 10 and the maximal number of generations for each elementary membrane randomly is 40. All experiments are executed for 20 independent runs. To compute the correlation between the original signal f and the restored signal f_N with a certain number of time-frequency atoms decomposed, the correlation ratio c_N of f and f_N is defined as [14]:

$$C_N = \frac{\langle f, f_N \rangle}{\|f\| \cdot \|f_N\|} \quad (4)$$

The correlation ratio 0.99 is used as the stopping condition for the algorithms, Chirp-PSO and Chirp-PSOPS. Experimental results are shown in Table III.

As can be seen from Table III, due to the relatively time-consuming communication rules, PSOPS spends relatively more computing time than PSO in a single run. However, PSOPS uses only 44.2 time-frequency atoms to obtain the same correlation ratio as PSO with 106.6 atoms averaged over 20 runs. Thus, PSOPS can spend smaller computing time than PSO to gain the same correlation ratio. This demonstrates that PSOPS have better performance than PSO, which also verifies the effectiveness and practicality of the method presented in this paper.

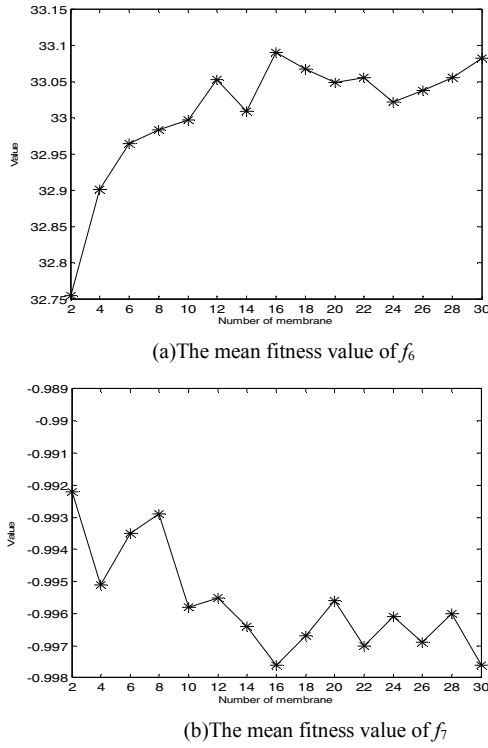


Figure 4. Experimental results with different number of elementary membrane

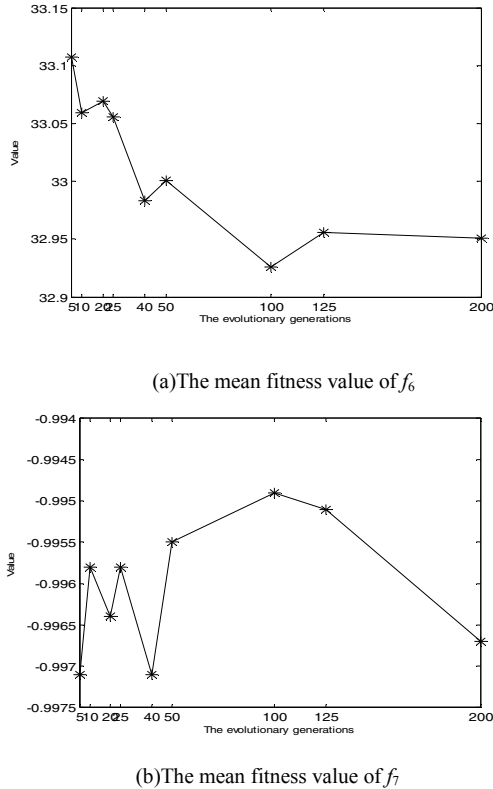


Figure 5. Experimental results with different maximal number of generations i

Table II. THE WORST, BEST AND MEAN FITNESS VALUES OF FUNCTIONS OVER 30 INDEPENDENT RUNS

Functions	PSOPS				PSO				optimum
	worst	best	mean	Mean time(s)	worst	best	mean	Mean time(s)	
f_1	2.4112e-004	3.0156e-006	5.7006e-005	4.92	19.1536	3.2913	13.5327	6.61	0
f_2	1.3006e-006	1.5839e-008	2.7392e-007	4.17	3.5733	1.4302e-007	0.9591	6.62	0
f_3	82.4121	11.5937	46.3721	4.62	541.5761	31.4834	183.0878	5.79	0
f_4	4.3233e-003	8.5873e-004	7.1456e-003	4.57	4.8866	1.7869e-003	2.1959	7.24	0
f_5	0	0	0	4.73	317	0	49.8667	6.5	0
f_6	32.7158	33.2315	33.0367	4.72	14.9269	18.8372	16.9052	5.83	36.4770
f_7	-0.9903	-1	-0.9974	3.57	-0.9903	-1	-0.9951	1.45	-1

Table III. PERFORMANCE COMPARISONS OF PSO AND PSOPS OVER 20 INDEPENDENT RUNS

	Chirp-PSO			Chirp-PSOPS		
	best	worst	mean	best	worst	mean
Correlation ratio CN	0.9901	0.9901	0.9901	0.9904	0.9900	0.9902
The number of atoms	93	120	106.6	40	51	44.2
Computing time (s)	82.70	128.34	98.73	60.18	75.51	64.84

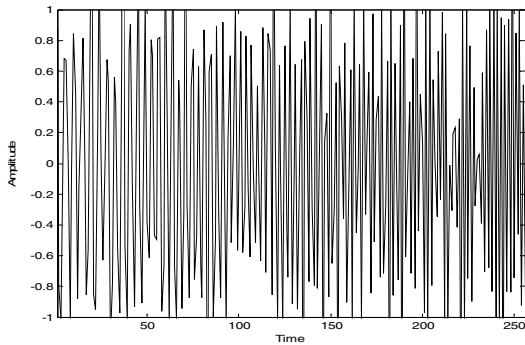


Figure 6. The original time-domain signal

V. CONCLUSIONS

This paper proposes a novel evolutionary algorithm, PSOPS, based on the concepts and principles of particle swarm optimization and P systems. This algorithm is characterized by employing the formal framework and communication rules of membrane computing and the evolutionary principles of PSO in each membrane. Its effectiveness and validity are verified by function optimization. Time-frequency analysis of a LFM radar emitter signal is also considered as an application example to investigate the performance of the introduced method. The results show that PSOPS to greatly decrease the computational load of TFAD, as compared with its counterpart PSO algorithm.

ACKNOWLEDGMENT

This work is supported partially by the National Natural Science Foundation of China (60702026, 60971103), the Scientific and Technological Funds for Young Scientists of Sichuan (09ZQ026-040) and the Open Research Fund of Key Laboratory of Signal and Information Processing, Xihua University (SZJJ2009-003). The work of MG, FI and RL was partially supported by CNC SIS – UEFISCSU, project number PNII – IDEI 643/2008.

REFERENCES

- [1] Gh.Păun, Computing with Membranes, Journal of Computer and System Sciences, Vol. 61, pp. 108-143, Nov. 2000.
- [2] Gh.Păun, G. Rozenberg, A Guide to Membrane Computing, Theoretical Computer Science, Vol. 287, pp. 73-100, Nov. 2002.
- [3] G. X. Zhang, M. Gheorghe, C. Z. Wu, A Quantum-Inspired Evolutionary Algorithm Based on P systems for a Class of Combinatorial optimization, Fundamenta Informaticae, Vol. 87, pp. 93-116, Nov. 2008.
- [4] L. Huang and N. Wang, An optimization algorithm inspired by membrane computing, ICNC 2006, Lecture Notes in Computer Science, Vol. 4222, pp. 49-52, 2006.
- [5] D. Zaharie and G. Ciobanu, Distributed evolutionary algorithms inspired by membranes in solving continuous optimization problems, WMC 2006, Lecture Notes in Computer Science, Vol. 4361, pp. 536-553, 2006.
- [6] G. X. Zhang, C. X. Liu, M. Gheorghe, F. Ipate. Solving Satisfiability Problems with Membrane Algorithms. in Proc. of BIC-TA, pp. 29-36, 2009.
- [7] C. X. Liu, G. X. Zhang, Y. H. Zhu, C. Fang, H. W. Liu, A Quantum-Inspired Evolutionary Algorithm Based on P systems for Radar Emitter Signals, in Proc. of BIC-TA, pp. 24-28, 2009.
- [8] C. X. Liu, G. X. Zhang, X. B. Zhang, H. W. Liu, A Memetic Algorithm Based on P Systems for IIR Digital Filter Design, in Proc. of PICom, pp. 330-334, 2009.
- [9] C. X. Liu, G. X. Zhang, H. W. Liu, M. Gheorghe, F. Ipate, An Improved Membrane Algorithm for Solving Time-frequency Atom Decomposition, Lecture Notes in Computer Science, Vol. 5957, pp. 371-384, 2010.
- [10] J. Kennedy, R. C. Eberhart, Particle Swarm Optimization, In Proc. of ICNN, pp. 1942-1948, 1995.
- [11] T. Li, L. B. Guo, Y. M. Wang, F. Hu, L. Xiao, Y. W. Wang, and Q. Cheng, Research on Logging Evaluation of Reservoir Contamination Based on PSO-BP Neural Network, Springer-Verlag Berlin Heidelberg, pp. 839-844, 2009.
- [12] H. M. Wang, X. Z. Gao, X. L. Huang, and Z. Y. Song, PSO-Optimized Negative Selection Algorithm for Anomaly Detection, Applications of Soft Computing, pp. 13-21, 2009.
- [13] S. G. Mallat, Z. F. Zhang, Matching Pursuits with Time-Frequency Dictionaries, IEEE Trans. on SP, Vol. 41, pp. 3397-3415, Nov. 1993.
- [14] G. X. Zhang, Time-Frequency Atom Decomposition with Quantum-Inspired Evolutionary Algorithms, Circuits Systems and Signal Processing, 2010, 29(2): 209-233.