2009 International Workshop on Nonlinear Circuits and Signal Processing
NCSP'09, Waikiki, Hawaii, March 1-3, 2009

Particle Swarm Optimizer Arranged in Two-Dimensional Array

Haruna Matsushita† and Yoshifumi Nishio†

†Tokushima University
2-1 Minami-Josanjima, Tokushima, Japan.
Phone:+81-88-656-7470, Fax:+81-88-656-7471
Email: {haruna, nishio}@ee.tokushima-u.ac.jp

Abstract

This study proposes a Particle Swarm Optimization with Map Structure (PSOMS). All particles of PSOMS are connected to adjacent particles by neighborhood relation, which dictates the topology, of the 2-dimensional map. Each particle is updated depending on the neighborhood distance between it and a winner, whose function value is best among all particles. Simulation results show the searching efficiency of PSOMS.

1. Introduction

Particle Swarm Optimization (PSO) [1] is an evolutionary algorithm to simulate the movement of flocks of birds. Due to the simple concept, easy implementation, and quick convergence, PSO has attracted much attention and is used to wide applications in different fields in recent years. However, PSO greatly depends on its parameters and converge prematurely in case of solving complex problems which have local optima. Furthermore, in PSO algorithm, there are no special relationships between particles. Each particle position is updated according to its personal best position and the best particle position among the all particles, and their weights are determined at random in every generation. On the other side, in the real world, various personal relationships exist, such as the hierarchical relationship, the trust relationships, the parents-child relationship and so on.

Various topological neighborhoods have been considered by researches [2]–[6]. They have applied ring neighborhood, the von Neumann neighborhood, or some other topological neighborhoods. However, the parameters are increased by using fourth term for considering the neighboring best position when updating the velocity. Moreover, their methods are complex algorithms although the one of advantages of the standard PSO is the simple concept.

In this study, we propose a new Particle Swarm Optimization with Map Structure (PSOMS) which is the simple algorithm as the standard PSO. All particles of PSOMS are arranged in 2-dimensional grid and are connected to adjacent particles by neighborhood relation which dictates the topology of the grid. In every generation, we find a winner particle, whose function value is best among all particles, and each particle is updated depending on the neighborhood distance between it and the winner on the map.

In Section 3, the algorithm of the proposed PSOMS is explained in detail. We apply PSOMS to two test functions, which are unimodal and multimodal function. Simulation results and comparisons with the standard PSO are shown in Section 4, and we confirm that the proposed PSOMS can effectively enhance the searching efficiency. Furthermore, in Section 5, we investigate the effect of the parameters on performance quality and their sensitivity. We confirm that PSOMS is more effective and its parametrical dependence is not stronger than the standard PSO.

2. Standard Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) is an evolutionary algorithm to simulate the movement of flocks of birds. In the algorithm of PSO, multiple solutions called "particles" coexist. At each time step, the particle flies towards its own past best position and the best position among all particles. Each particle has two informations; position and velocity. The position vector of each particle i and its velocity vector are represented by $\boldsymbol{X}_i = (x_{i1}, \cdots, x_{id}, \cdots, x_{iD})$ and $\boldsymbol{V}_i = (v_{i1}, \cdots, v_{id}, \cdots, v_{iD})$, respectively, where $(d = 1, 2, \cdots, D)$, $(i = 1, 2, \cdots, M)$ and $x_{id} \in [x_{\min}, x_{\max}]$.

(PSO1) (Initialization) Let a generation step $t = 0$. Randomly initialize the particle position $\boldsymbol{X}_i$ and its velocity $\boldsymbol{V}_i$ for each particle i, and initialize $\boldsymbol{P}_i = (p_{i1}, p_{i2}, \cdots, p_{iD})$ with a copy of $\boldsymbol{X}_i$. Evaluate the objective function $f(\boldsymbol{X}_i)$ for each particle i and find $\boldsymbol{P}_g$ with the best function value among the all particles.

(PSO2) Evaluate the fitness $f(\boldsymbol{X}_i)$. For each particle i, if $f(\boldsymbol{X}_i) < f(\boldsymbol{P}_i)$, the personal best position (called *pbest*) $\boldsymbol{P}_i = \boldsymbol{X}_i$. Let $\boldsymbol{P}_g$ represents the best position with the best fitness among all particles so far (called *gbest*). Update $\boldsymbol{P}_g$, if needed.

(PSO3) Update $\boldsymbol{V}_i$ and $\boldsymbol{X}_i$ of each particle i depending on its *pbest* and *gbest* according to

$$
\begin{aligned}
v_{id}(t+1) = & \; w v_{id}(t) + c_1 \mathrm{rand}(\cdot)\,(p_{id} - x_{id}(t)) \\
& + c_2 \mathrm{Rand}(\cdot)\left(p_g - x_{id}(t)\right), \\
x_{id}(t+1) = & \; x_{id}(t) + v_{id}(t+1),
\end{aligned}
\tag{1}
$$

where w is the inertia weight determining how much of the previous velocity of the particle is preserved. c_1 and c_2 are two positive acceleration coefficients, generally $c_1 = c_2$. $\mathrm{rand}(\cdot)$ and $\mathrm{Rand}(\cdot)$ are two uniform random numbers samples from $U(0,1)$.

(PSO4) Let $t = t+1$ and go back to (PSO2).

3. Particle Swarm Optimization with Map Structure (PSOMS)

In the algorithm of PSO, multiple solutions called "particles" coexist. The most important feature of PSOMS is that all particles are organized on a rectangular 2-dimensional grid. In other words, the particles are connected to adjacent particles by neighborhood relation, which dictates the topology, of the map. The position vector of each particle i and its velocity vector are represented by $\boldsymbol{X}_i = (x_{i1}, \cdots, x_{id}, \cdots, x_{iD})$ and $\boldsymbol{V}_i = (v_{i1}, \cdots, v_{id}, \cdots, v_{iD})$, respectively, where $(d = 1, 2, \cdots, D)$, $(i = 1, 2, \cdots, M)$ and $x_{id} \in [x_{\min}, x_{\max}]$.

(PSOMS1) (Initialization) Let a generation step $t = 0$. Randomly initialize the particle position $\boldsymbol{X}_i$ and its velocity $\boldsymbol{V}_i$ for all particles i, and initialize $\boldsymbol{P}_i = (p_{i1}, p_{i2}, \cdots, p_{iD})$ with a copy of $\boldsymbol{X}_i$. Evaluate the objective function $f(\boldsymbol{X}_i)$ for all particle i and find $\boldsymbol{P}_g$ with the best function value among the all particles. Define g as the winner c.

(PSOMS2) Evaluate the fitness $f(\boldsymbol{X}_i)$ and find the winner particle c with the best fitness among the all particles at current time.

$$
c = \arg\min_i \{ f(\boldsymbol{X}_i(t)) \}. \tag{2}
$$

For each particle i, if $f(\boldsymbol{X}_i) < f(\boldsymbol{P}_i)$, the personal best position (called *pbest*) $\boldsymbol{P}_i = \boldsymbol{X}_i$.

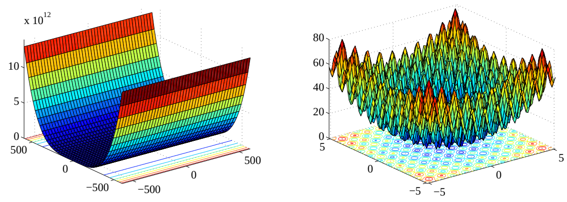
Let $\boldsymbol{P}_g$ represents the best position with the best fitness among all particles so far (called *gbest*). If $f(\boldsymbol{X}_c) < f(\boldsymbol{P}_g)$, update *gbest* $\boldsymbol{P}_g = \boldsymbol{X}_c$, where $\boldsymbol{X}_c$ is the position of the winner c.

(PSOMS3) Update $\boldsymbol{V}_i$ and $\boldsymbol{X}_i$ of each particle i depending on its *pbest*, the position of the winner c and the distance on the map between i and the winner c, according to

$$
\begin{aligned}
v_{id}(t+1) = & \; w v_{id}(t) + c_1 \mathrm{rand}(\cdot)\,(p_{id} - x_{id}(t)) \\
& + c_2 h_{c,i}\,(x_{cd} - x_{id}(t)), \\
x_{id}(t+1) = & \; x_{id}(t) + v_{id}(t+1),
\end{aligned}
\tag{3}
$$

where $h_{c,i}$ is the fixed neighborhood function defined by

$$
h_{c,i} = \exp\left(-\frac{\|\boldsymbol{r}_i - \boldsymbol{r}_c\|^2}{2\sigma^2} \right), \tag{4}
$$



(a) (b)

Figure 1: Two test functions with two variables. First and second variables are on the x-axis and y-axis, respectively, and z-axis shows its function value. (a) Rastrigin function f_1 which is unimodal function. (b) Rastrigin function f_2 which is multimodal function.

where $\|\boldsymbol{r}_i - \boldsymbol{r}_c\|$ is the distance between map nodes c and i on the map, the fixed parameter σ corresponds to the width of the neighborhood function. Therefore, the large σ strengthens particles' spreading force to the whole space, and the small σ strengthens their convergent force toward the winner.

(PSOMS4) Let $t = t+1$ and go back to (PSOMS2).

4. Simulation Experimentation

In order to evaluate the performance of PSOMS, we use two benchmark optimization problems. One is the Rosenbrock function f_1 as Eq. (5), which is unimodal function shown in Fig. 1(a), and the other is the Rastrigin function f_2 as Eq. (6) which is multimodal function shown in Fig. 1(b) with numerous local minima.

$$
\begin{aligned}
f_1(x) = & \sum_{d=1}^{D-1} \left(100 \left(x_d^2 - x_{d+1} \right)^2 + (1 - x_d)^2 \right), \\
& x \in [-2.048, 2.047]^D
\end{aligned}
\tag{5}
$$

$$
\begin{aligned}
f_2(x) = & \sum_{d=1}^{D} \left(x_d^2 - 10\cos(2\pi x_d) + 10 \right), \\
& x \in [-5.12, 5.12]^D
\end{aligned}
\tag{6}
$$

For both two functions, we use $D = 100$ variables. The optimum solutions x^* of f_1 and f_2 are $[1, 1, \ldots, 1]$ and $[0, 0, \ldots, 0]$, respectively, and the optimum function values $f(x^*)$ of both functions are 0.

The population size is set to 36 in PSO, and the network size is 6×6 in the proposed PSOMS. We choose the best parameters for each algorithm by the trial-and-error method although PSOMS can obtain better results than PSO even if PSOMS uses same parameters as PSO. For PSO, $w = 0.7$ and $c_1 = c_2 = 1.6$. For PSOMS, $w = 0.8$, $c_1 = c_2 = 1.8$ and $\sigma = 1.0$. We carry out the simulations repeated 30 times for each optimization function with 2000 time steps.

Table 1: Comparison Results of PSO and PSOMS for f_1 and f_2.

f	Method	Mean	Minimum	Maximum
f_1	PSO	210.655	81.773	313.365
	PSOMS	108.529	94.227	147.550
f_2	PSO	442.639	334.330	577.101
	PSOMS	190.151	135.687	252.401

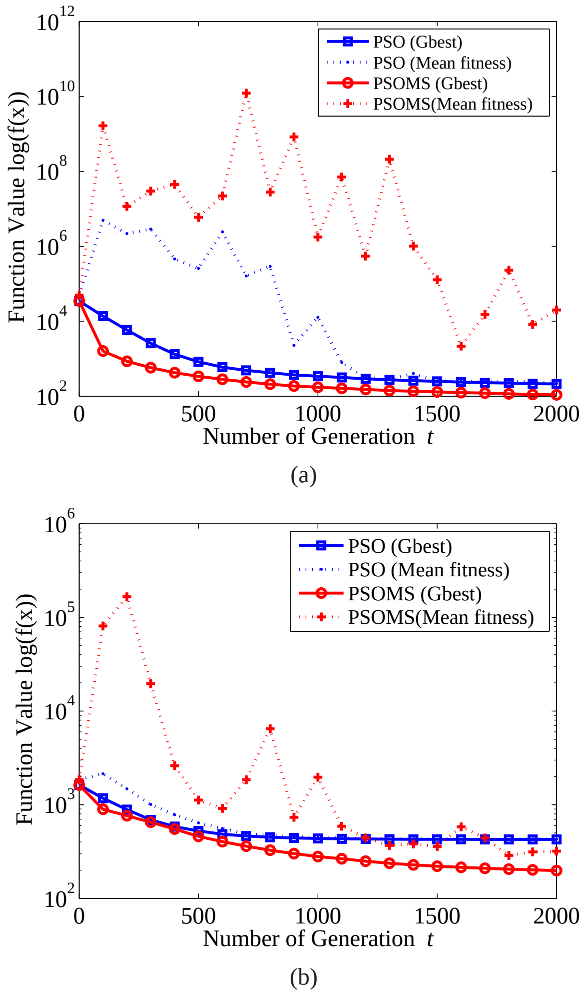


Figure 2: Mean *gbest* value of every generation, and mean fitness value of all particles of every generation over 30 runs. (a) Rosenbrock function f_1. (b) Rastrigin function f_2.

4.1. Experiment Results

The performance of PSO and PSOMS with their corresponding minimum and mean function values are listed in Table 1. We can see that the results of PSOMS have better accuracy. In PSO, the number of particles which move toward *gbest* or toward *pbest* is decided by random on every generation and is not stable. On the other hand, the neighborhood gaussian function is used in PSOMS, therefore, the particles move according to the neighborhood distance between the winner and them. The winner's neighborhood particles move beyond the winner so that they spread to whole space. The particles, which are connected at a little distance from the winner, move toward the winner. The other particles fly toward their *pbest*. In other words, the roles of the PSOMS particles are determined by the connection relationship.

Figures 2 shows the mean *gbest* value and the mean function value of all the particles of every generation over 30 runs for two functions. In other words, the mean *gbest* value of every generation shows the swarm convergence, and the mean fitness value of all the particles shows the particle-diversity of the swarm. From these figures, the convergence rate of PSOMS is almost same or slower than the standard PSO, and it is clear that PSOMS has the more diversity of the particles than the standard PSO. We can appear this behavior prominently in Rastrigin function f_2 which is the multimodal function. This is because that the roles of the PSOMS particles are determined by the connection relationship, therefore they produce the diversity of the particles. These effects avert the premature convergence, and the particles of PSOMS can easily escape from the local optima.

5. Parameter Dependence

Furthermore, in order to investigate the effect of the parameters; the inertia weight w and the acceleration coefficients c_1 and c_2, on performance quality and their sensitivity, Figs. 3 and 4 show the mean function values with different parameters. The fixed parameters are same as above simulations. We can see that PSOMS achieve better performances than the standard PSO for both test functions even if the parameters are varied. If we decrease or increase the parameter by just 0.1, the performance of the standard PSO becomes drastically worse as especially. In other words, the performance of PSO is sensitive to the parameters, however, the performance of PSOMS is stable. From these results, the proposed PSOMS is more effective and the parametrical dependence is not stronger than PSO.

6. Conclusions

This study has proposed a Particle Swarm Optimization with Map Structure (PSOMS) which is the simple method as the standard PSO. All particles of PSOMS are connected to adjacent particles by neighborhood relation, which dictates the topology, of the 2-dimensional map. Each particle is updated depending on the neighborhood distance between it and

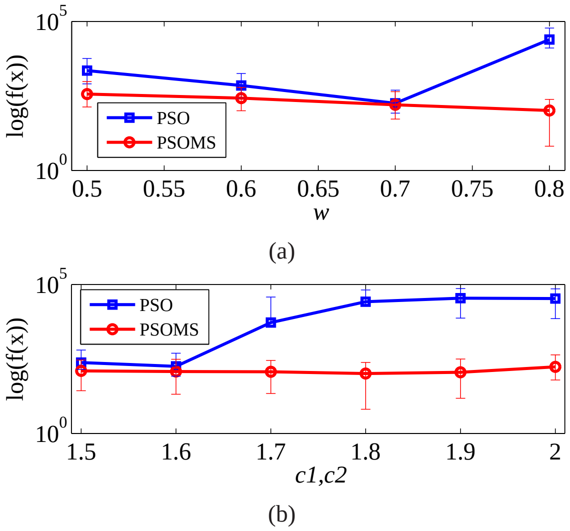


Figure 3: Results for Rosenbrock function f_1 with different parameters. (a) Using different w. $c_1(= c_2)$ is fixed as 1.6 for PSO and as 1.8 for PSOMS. (b) Using different $c_1(= c_2)$. w is fixed as 0.7 for PSO and as 0.8 for PSOMS.

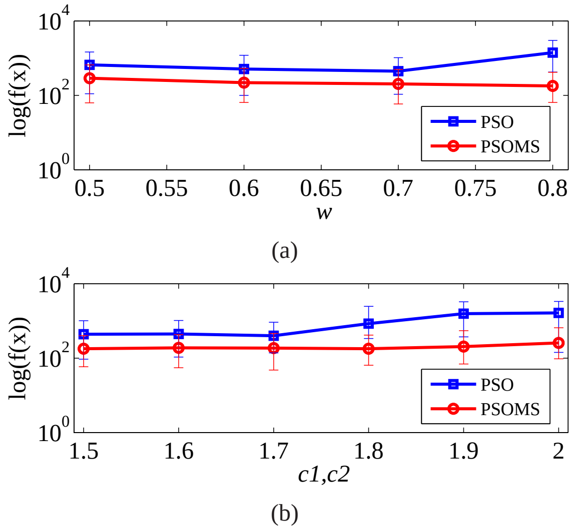


Figure 4: Results for Rastrigin function f_2 with different parameters. (a) Using different w. $c_1 = c_2$ is fixed as 1.6 for PSO and as 1.8 for PSOMS. (b) Using different $c_1(= c_2)$. w is fixed as 0.7 for PSO and as 0.8 for PSOMS.

a winner, whose function value is best among all particles. In the simulation results, the searching efficiency of PSOMS is better than PSO. Furthermore, we have confirmed that the parametrical dependence of PSOMS is not stronger than PSO.

References

[1] J. Kennedy and R. C. Eberhart, “Particle swarm optimization,” in *Proc. IEEE. Int. Conf. Neural Netw.*, pp. 1942–1948, 1995.

[2] J. Kennedy, “Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance,” in *Proc. of Cong. on Evolut. Comput.*, pp. 1931–1938, 1999.

[3] J. Kennedy and R. Mendes, “Population structure and particle swarm performance,” in *Proc. of Cong. on Evolut. Comput.*, pp. 1671–1676, 2002.

[4] R. Mendes, J. Kennedy and J. Neves, “The Fully Informed Particle Swarm: Simpler, Maybe Better,” in *IEEE Trans. Evolut. Comput.*, vol. 8, no.3, pp. 204–210, June 2004.

[5] J. Lane, A. Engelbrecht and J. Gain, “Particle Swarm Optimization with Spatially Meaningful Neighbours” in *Proc. of IEEE Swarm Intelligence Symposium*, 2008.

[6] S. B. Akat and V. Gazi, “Particle Swarm Optimization with Dynamic Neighborhood Topology: Three Neighborhood Strategies and Preliminary Results” in *Proc. of IEEE Swarm Intelligence Symposium*, 2008.