2008 IEEE Workshop on
Nonlinear Circuit Networks, Tokushima, Dec 5-6, 2008

Particle Swarm Optimization with Map Structure

Haruna Matsushita and Yoshifumi Nishio
Department of Electrical and Electronic Engineering,
Tokushima University
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.**

I. 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.

In this study, we propose a new 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. 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. Simulation results and comparisons with the standard PSO show that the proposed PSOMS can effectively enhance the searching efficiency.

II. 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.

$$c = \arg\min_i \{f(\boldsymbol{X}_i(t))\}. \tag{1}$$

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*, *gbest* and the distance on the map between i and the winner c, according to

$$\begin{aligned} v_{id}(t+1) &= w v_{id}(t) + c_1 \mathrm{rand}(\cdot)\,(p_{id} - x_{id}(t)) \\ &\qquad + c_2 h_{c,i}\,(x_{cd} - x_{id}(t))\,, \\ x_{id}(t+1) &= \ x_{id}(t) + v_{id}(t+1), \end{aligned} \tag{2}$$

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 $c_1 = c_2$. $\mathrm{rand}(\cdot)$ is an uniform random numbers samples from $U(0,1)$. $h_{c,i}$ is the fixed neighborhood function defined by

$$h_{c,i} = \exp\left(-\frac{\|\boldsymbol{r}_i - \boldsymbol{r}_c\|^2}{2\sigma^2}\right), \tag{3}$$

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).

III. Experimental Results

In order to evaluate the performance of PSOMS, we use two benchmark optimization problems. One is the Rosenbrock function f_1 as Eq. (4) and the other is the Rastrigin function

TABLE I

COMPARISON RESULTS OF PSO AND PSOMS FOR f_1 AND f_2.

f	Method	Mean	Minimum	Maximum
f_1	PSO	178.5590	95.0156	312.9265
	PSOMS	102.9688	96.4675	138.9978
f_2	PSO	444.3451	337.3337	584.0390
	PSOMS	178.9360	114.3723	236.4654

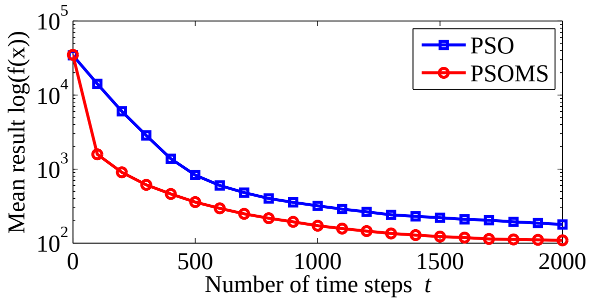

Fig. 1. Mean Rosenbrock function value of every generation.

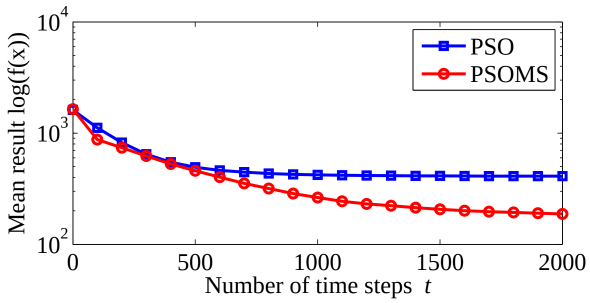

Fig. 2. Mean Rastrigin function value of every generation.

f_2 as Eq. (5).

$$f_1(x) = \sum_{d=1}^{D-1} \left(100 \left(x_d^2 - x_{d+1}\right)^2 + (1 - x_d)^2\right), \quad x \in [-2.048, 2.047]^D \tag{4}$$

$$f_2(x) = \sum_{d=1}^{D} \left(x_d^2 - 10 \cos\left(2\pi x_d\right) + 10\right), \quad x \in [-5.12, 5.12]^D \tag{5}$$

For both two functions, we use $D = 100$ dimensions. 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 $\sigma = 1.0$.

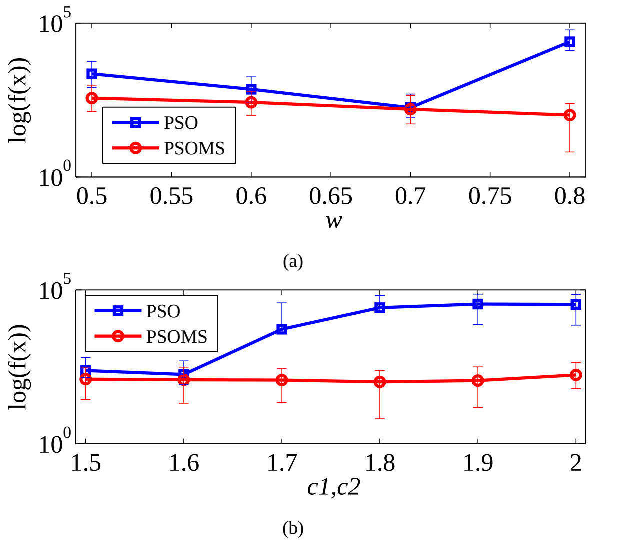

Fig. 3. Results for Rosenbrock function f_1 with different parameters. (a) Using different w. $c_1 (= c_2)$ is fixed as 1.6 for PSO and as 1.8 for PSOMS. (b) Using different $c_1 (= c_2)$. w is fixed as 0.7 for PSO and as 0.8 for PSOMS.

We carry out the simulations repeated 30 times for each optimization function with 2000 time steps. The performance of PSO and PSOMS with their corresponding minimum and mean function values are listed in Table I. Figures 1 and 2 show the mean value of the best function value of every generation over 30 runs for f_1 and f_2 function, respectively. From these results, 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.

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. The proposed PSOMS is more effective and the parametrical dependence is not stronger than PSO. The performance of PSO is sensitive to the parameters, however, the performance of PSOMS is stable.

IV. CONCLUSIONS

This study has proposed 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

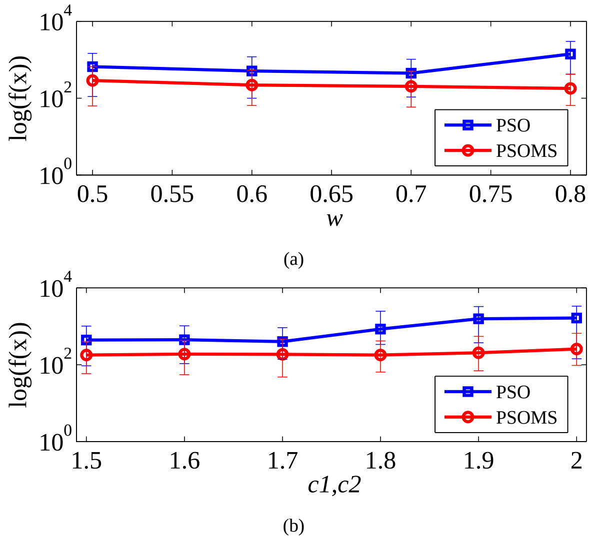

Fig. 4. Results for Rastrigin function f_2 with different parameters. (a) Using different w. $c_1 = c_2$ is fixed as 1.6 for PSO and as 1.8 for PSOMS. (b) Using different $c_1(= c_2)$. w is fixed as 0.7 for PSO and as 0.8 for PSOMS.

and 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.