2009 International Symposium on Nonlinear Theory and its Applications
NOLTA'09, Sapporo, Japan, October 18-21, 2009

Network-Structured Particle Swarm Optimizer with Small-World Topology

Haruna Matsushita† and Yoshifumi Nishio†

†Tokushima University,
2–1 Minami-Josanjima, Tokushima, Japan.
Email: {haruna, nishio}@ee.tokushima-u.ac.jp

Abstract—This study proposes Network-Structured Particle Swarm Optimizer (NS-PSO) with Small-World topology. All particles are connected to adjacent particles depending on the small-world network. The directly connected particles share their own best position. Each particle is updated depending on the neighborhood distance on the network between it and a winner, whose function value is best among all particles. We apply NS-PSO with small-world topology to various optimization problems and confirm the effectiveness of the proposed model.

1. Introduction

Particle Swarm Optimization (PSO) [1] is an 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. 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. Due to these features, the standard PSO greatly depends on its parameters and converge prematurely in case of solving complex problems which have local optima.

On the other hand, the Self-Organizing Map (SOM) [2] is an unsupervised learning algorithm. SOM consists of neurons located on 2-dimensional network. The neurons self-organize statistical features of the input data according to the neighborhood relationship of the map structure.

Various topological neighborhoods for PSO have been considered by researches [3]–[7]. Each particle shares its best position among neighboring particles on the network. However, the information of each particle is not updated depending on the neighborhood distance on the network.

In our past study, we have applied the concept of SOM to PSO and have proposed Network-Structured Particle Swarm Optimizer considering neighborhood relationships (NS-PSO) [8][9]. All particles of NS-PSO are connected to adjacent particles by a neighborhood relation, which dictates the topology of the networks. The connected particles, namely neighboring particles on the network, share the information of their own past best position. In every generation, we find a winner particle, whose function value is the best among all particles, as SOM algorithm, and each particle is updated depending on the neighborhood distance between it and the winner on the network. We applied NS-PSO to the various network topology as rectangular, hexagonal, cylinder and toroidal. From comparison results, we find that the circular-topology is effective for the simple unimodal functions and the hexagonal-topology is appropriate for the complex multimodal functions.

In this study, we apply NS-PSO to well-known network; "small-world network" suggested by Watts and Strogatz [10]. Simulation results and comparisons with the previous PSOs show that the proposed NS-PSO with small-world topology can effectively enhance the searching efficiency.

2. Network-Structured Particle Swarm Optimizer with Small-World Topology Considering Neighborhood Relationships (NS-PSO)

In the algorithm of the standard PSO, multiple solutions called "particles" coexist. At each time step, the particle flies toward 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 $x_{id} \in [x_{\min}, x_{\max}]$.

The algorithm of NS-PSO is based on both two structures; the standard PSO and SOM. NS-PSO has following three key features.

1. All particles are connected to adjacent particles by a neighborhood relation, which dictates the topology of the network.

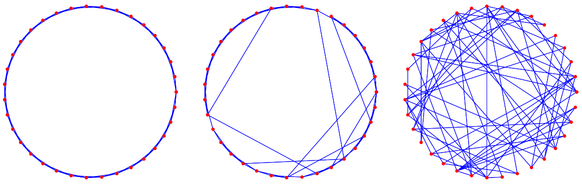
2. The particles share the local best position between the neighborhood particles directly connected.

3. In every generation, we find a winner particle with best function value among all particle as SOM learning.

By these features, each particle of NS-PSO is updated depending on its own best position, the position of the winner and the neighborhood distance between it and the winner on the network.

2.1. Small-World Model

In this study, we apply NS-PSO to the 1-dimensional (1-D) small-world topology shown in Fig. 1(b). The 1-D



(a) (b) (c)

Figure 1: 1-dimensional network topology with 36 particles. $k = 2$. (a) 1-dimensional lattice ($p = 0$). (b) Small-World topology ($p = 0.04$). (c) Random graph ($p = 1$).

small-world topology is defined on a lattice with M particles and periodic boundary conditions. The 1-D small-world topology is generated by following algorithm;

(1) Connect each particle i to its k neighbor particles according to the topology of 1-dimensional lattice.

(2) Rewire each particle i to another particle chosen at random with probability p.

When $p = 0$, the network topology is 1-dimensional lattice, and when $p = 1$, it is a random graph.

2.2. Algorithm of NS-PSO with Small-World Topology

The algorithm of NS-PSO with the small-world topology is same as the conventional NS-PSO except its topology and a neighborhood function as Eq. (3).

(NS-PSO1) (Initialization) Let a generation step $t = 0$. Randomly initialize the particle position $\boldsymbol{X}_i$, initialize its velocity $\boldsymbol{V}_i$ for each particle i to zero, 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 all the particles. Define g as the winner c. Find $\boldsymbol{L}_i = (l_{i1}, l_{i2}, \cdots, l_{iD})$ with the best function value among the directly connected particles, namely own neighbors. Connect all the particles to adjacent particles according to the subsection 2.1.

(NS-PSO2) Evaluate the fitness $f(\boldsymbol{X}_i)$ and find a winner particle c with the best fitness among the all particles at current time t;

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

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 $\boldsymbol{X}_c = (x_{c1}, x_{c2}, \cdots, x_{cD})$ is the position of the winner c.

(NS-PSO3) Find each local best position (called *lbest*) $\boldsymbol{L}_i$ among the particle i and its neighborhoods, which are directly connected with i on the network, so far. For each particle i, update *lbest* $\boldsymbol{L}_i$, if needed.

(NS-PSO4) Update $\boldsymbol{V}_i$ and $\boldsymbol{X}_i$ of each particle i depending on its *lbest*, position of the winner $\boldsymbol{X}_c$ and the distance on the network between i and the winner c, according to

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

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 number sample from $U(0,1)$. $h_{c,i}$ is the fixed neighborhood function defined by

$$h_{c,i} = \exp\left(-\frac{dis(c,i)}{\sigma^2}\right), \tag{3}$$

where $dis(c,i)$ is the shortest-path distance between particles c and i on the network and is called *neighborhood distance*. The fixed parameter σ corresponds to the width of the neighborhood function. Therefore, large σ strengthens particles' spreading force to the whole space, and small σ strengthens their convergent force toward the winner.

(NS-PSO5) Let $t = t + 1$ and go back to (NS-PSO2).

3. Simulation

In order to evaluate the performance of NS-PSO with small-world topology, we use six benchmark optimization problems summarized in Table 1. f_1, f_2 and f_3 are unimodal functions, and f_4, f_5 and f_6 are multimodal functions with numerous local minima. The optimum solution x^* of Rosenbrock's function f_2 is $[1, 1, \ldots, 1]$, and x^* of the other functions are all $[0, 0, \ldots, 0]$. The optimum value $f(x^*)$ of all the functions is 0. All the functions have D variables. In this study, D is set to 30 and 50 to investigate the performances in various dimensions. The landscape maps of benchmark functions with two variables are shown in Fig. 2.

We compare NS-PSO with small-world topology to the standard PSO, PSO with small-world topology and NS-PSO with rectangular-topology. Features of each algorithm are follows:

PSO: This is the standard PSO with NO neighborhood relationship. Each particles is updated depending on its *pbest* and *gbest*.

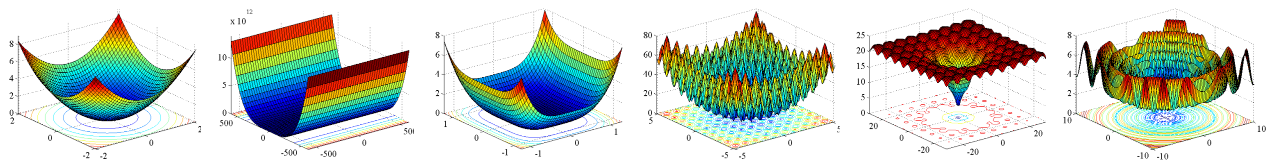
PSO with small-world topology: Its particles are connected to other particles according to small-world network and share their *lbest* with directly connected particles. Each particles is updated depending on its *lbest* and *gbest*. The winner does NOT exist, and the neighborhood distance is NOT considered.

NS-PSO with various topology: Its particles are connected to other particles according to the topology of the network and share their *lbest* with directly connected particles. Each particles is updated depending on its *lbest*, winner's position and the neighborhood distance.

The population size M is set to 36 in PSO, and the network sizes are 36 in NS-PSO with small-world topology

Table 1: Six Test Functions.

Function name	Test Function	Initialization Space
Sphere function;	$f_1(x) = \sum_{d=1}^{D-1} x_d^2,$	$x \in [-2.048, 2.048]^D$
Rosenbrock's function;	$f_2(x) = \sum_{d=1}^{D-1} \left(100\left(x_d^2 - x_{d+1}\right)^2 + (1 - x_d)^2\right),$	$x \in [-2.048, 2.048]^D$
4th De Jong's function;	$f_3(x) = \sum_{d=1}^{D} d x_d^4,$	$x \in [-1.28, 1.28]^D$
Rastrigin's function;	$f_4(x) = \sum_{d=1}^{D} \left(x_d^2 - 10\cos(2\pi x_d) + 10\right),$	$x \in [-5.12, 5.12]^D$
Ackley's function;	$f_5(x) = \sum_{d=1}^{D-1} \Big(20 + e - 20e^{-0.2\sqrt{0.5(x_d^2 + x_{d+1}^2)}} - e^{0.5(\cos(2\pi x_d) + \cos(2\pi x_{d+1}))}\Big),$	$x \in [-30, 30]^D$
Stretched V sine wave function;	$f_6(x) = \sum_{d=1}^{D-1} (x_d^2 + x_{d+1}^2)^{0.25} \left(1 + \sin^2(50(x_d^2 + x_{d+1}^2)^{0.1})\right),$	$x \in [-10, 10]^D$



(a) (b) (c) (d) (e) (f)

Figure 2: Six 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) Sphere function. (b) Rosenbrock's function. (c) 4th De Jong's function. (d) Rastrigin's function. (e) Ackley's function. (f) Stretched V sine wave function.

and 6×6 in NS-PSO with rectangular-topology. For all PSOs, the parameters are set as $w = 0.7$ and $c_1 = c_2 = 1.6$. The neighborhood radius σ of NS-PSO with small-world topology and with rectangular-topology are 1.6 and 1.5, respectively. To generate the small-world topology, the probability p is chosen 0.04, and the neighborhood parameter k is set to 2. We carry out the simulations repeated 30 times for all the optimization functions with 3000 generations.

3.1. Experimental Results

The performances with the minimum and mean function values over 30 independent runs on six functions are listed in Table 2. The best results of the mean values among all the algorithms are shown in bold.

On both dimension, the standard PSO, PSO with small-world topology, NS-PSO with rectangular-topology and with small-world topology achieve the best values 0, 1, 2 and 9 times, respectively. It is interesting to note that PSO with small-world topology can not obtain better results than the standard PSO with no connection, and in fact, it dramatically degrades the performance on almost functions. However, two kinds of NS-PSOs evidently surpass not only the standard PSO but also PSO with small-world topology, on almost functions for both dimensions. These results mean that it is not important to share *lbest* among neighbors, and updating with considering the neighborhood distance produces the effective results. In the updating of NS-PSO, the neighborhood gaussian function is used, then, the particles move according to the neighborhood distance between the winner and them. The roles of the NS-PSO particles are determined by the connection relationship, and they produce the diversity of the particles. These effects avert the premature convergence, and the particles of NS-PSO can easily escape from the local optima.

In contrast to PSO, NS-PSO with small-world topology obtain significantly-improved results over rectangular-topology. In particular, NS-PSO with small-world topology tends to upgrade the performance on unimodal functions as f_1, f_2 and f_3. It is considered that NS-PSO with small-world topology has advantages of both the particle-diversity and the ease of exchanging of the particle information.

From these results, we can say that NS-PSO, whose par-

Table 2: Comparison results of PSO and NS-PSO with small-world topology on 6 test functions.

D	f		PSO		NS-PSO	
			No connection	Small-World	Rectangular	Small-World
30	f_1	Mean	9.10e-51	4.53e-37	6.28e-55	**2.16e-75**
		Minimum	3.16e-56	6.91e-41	1.23e-58	3.66e-85
	f_2	Mean	16.87	19.57	14.74	**5.64**
		Minimum	2.17	0.70	5.37	1.16
	f_3	Mean	4.86e-78	2.19e-58	1.22e-89	**1.04e-124**
		Minimum	3.78e-85	2.62e-65	2.07e-96	2.67e-149
	f_4	Mean	64.34	82.88	42.48	**30.99**
		Minimum	44.77	41.79	23.88	11.94
	f_5	Mean	70.81	47.58	**45.75**	66.22
		Minimum	5.16	2.58e-14	5.16	5.16
	f_6	Mean	22.18	22.06	**9.29**	11.83
		Minimum	10.44	11.84	2.4	5.05
50	f_1	Mean	2.29e-20	9.35e-17	8.22e-25	**1.16e-33**
		Minimum	4.09e-27	7.31e-20	1.51e-29	4.30e-42
	f_2	Mean	55.24	66.34	43.61	**36.89**
		Minimum	36.74	25.38	38.48	19.14
	f_3	Mean	1.58e-35	1.81e-26	1.51e-41	**6.52e-60**
		Minimum	9.86e-42	3.47e-29	7.96e-47	4.35e-69
	f_4	Mean	148.31	188.41	92.8	**61.29**
		Minimum	94.52	123.37	52.73	28.85
	f_5	Mean	249.67	**125.55**	159.62	225.34
		Minimum	97.84	19.22	67.60	143.31
	f_6	Mean	65.62	68.72	41.35	**39.20**
		Minimum	39.36	44.58	21.95	23.02

ticles are updated depending on the neighborhood distance, is more effective than the standard PSO, which has no neighborhood relationship, and PSO with network topology which does not use the neighborhood distance. Furthermore, although the results slightly depend on the problems, NS-PSO with small-world topology is suitable for optimization, especially on unimodal functions.

4. Conclusions

In this study, we have proposed Network-Structured Particle Swarm Optimizer (NS-PSO) with Small-World topology. All particles of NS-PSO are connected to adjacent particles by a neighborhood relation of small-world network, and their information are updated depending on the neighborhood topology. We have applied NS-PSO with small-world topology to optimization problems. We have confirmed that NS-PSO, whose particles are updated depending on the neighborhood distance, is more effective than the standard PSO, which does not use the neighborhood distance. Furthermore, NS-PSO with small-world topology is suitable for optimization, especially on unimodal functions.

References

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

[2] T. Kohonen, *Self-organizing Maps*, Berlin, Springer, 1995.

[3] 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.

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

[5] 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.

[6] J. Lane, A. Engelbrecht and J. Gain, “Particle Swarm Optimization with Spatially Meaningful Neighbours,” in *Proc. of IEEE Swarm Intelligence Symposium*, pp. 1–8, 2008.

[7] 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*, pp. 1–8, 2008.

[8] H. Matsushita and Y. Nishio, “Network-Structured Particle Swarm Optimizer Considering Neighborhood Relationships,” in *Proc. of IEEE. Int. Joint Conf. on Neural Netw.*, pp. 2038–2044, 2009.

[9] H. Matsushita and Y. Nishio, “Network-Structured Particle Swarm Optimizer with Various Topology and its Behaviors,” in *Lecture Notes in Computer Science*, vol. 5629, pp. 163–171, 2009.

[10] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’ networks,” Nature 393, pp.440–442, 1998.