2014 International Symposium on Nonlinear Theory and its Applications
NOLTA2014, Luzern, Switzerland, September 14-18, 2014

A Temporal PSO networks for Parallel-Distributed Optimization

Tomoyuki Sasaki†, Hidehiro Nakano†, Arata Miyauchi† and Akira Taguchi†

†Tokyo City University, 1-28-1, Tamazutsumi, Setagaya-ku, Tokyo, 158-8557, Japan
Email: sasaki@ic.cs.tcu.ac.jp

Abstract—In this paper, two particle swarm optimizer networks are proposed. In the proposed algorithms, multiple sub-swarm groups construct a global swarm with a network structure. The first network has deterministic links to each sub-swarm. The second network has temporal links to each sub-swarm. The proposed methods are evaluated by the computer simulations. The solving performances and the number of communications in each method are investigated.

1. Introduction

Recently, systems have been large scale and complicated with the developments of technologies. So, it is difficult to search the optimum statements of the systems that satisfy the constrained conditions within realistic time. Various optimization algorithms called meta-heuristics are developed to obtain high quality solutions speedily for these optimization problems. However, as the optimization problems have been large scale, the evaluation costs for the objective functions have significantly increased. Therefore, reducing the number of evaluations for the objective functions has been a very important subject.

For such a subject, there are methods to parallelize the calculations for the objective functions by using multiple processors, and to reduce them in each processor. In this approach, it is important to satisfy the following requirements: (1) the number of interprocess communications which can be large overhead is reduced, and (2) the parallelized algorithms keep or improve the solving performances as possible, compared with the non-parallelized algorithms.

In this paper, Particle Swarm Optimizers (PSOs) with group-of-group network topologies are proposed. PSO is an optimization algorithm inspired by flocking behavior of living beings [1]. In PSO, particles move to the directions to desired solutions in the search space, by sharing the information of the best solutions in the search process. The basic PSO has high parallelism, since the evaluations of each particle is independent and only the information of the best solutions is exchanged.

In the proposed algorithms, multiple sub-swarms construct a global swarm with network topologies. The sub-swarms exchange the local best solutions of them to each other. Such a group-of-group network topology can provide the following advantages: (1) each sub-swarm group searches solutions independently, and the diversity of the solutions can be generated, (2) the evaluation values for the objective functions are calculated in each sub-swarm group independently, and the parallel-distributed computing with high parallelism can be realized, and (3) only the information of the best solutions is exchanged, and the overhead of the interprocess communications can be reduced as possible.

The various PSO algorithms using multiple sub-swarms have been proposed [2]-[5]. However, it has not been sufficiently considered how the network topologies between the sub-swarms affect the solving performances. On the other hand, the various PSO algorithms with network topologies have been proposed [6]-[11]. However, the network topologies are introduced to the couplings between each particle; they are not multiple sub-swarm models but global swarm models.

In this paper, the following two algorithms using group-of-group network topologies are proposed. (1) PSO networks with Deterministic couplings (PSON-D). (2) PSO networks with Temporal couplings (PSON-T). The proposed methods are evaluated by the computer simulations. Then, the solving performances and the number of communications in each method are investigated. It is shown that the proposed algorithms can significantly improve the solving performances in various benchmark functions with very small interprocess communications. Our algorithms are suitable for implementation on parallel-distributed computer systems.

2. PSO networks with deterministic couplings (PSON-D)

In this section, the PSO Networks with Deterministic couplings (PSON-D) are explained. Figure 1 shows the example of PSON-D. In this figure, the swarm is divided into multiple sub-swarm groups, and each group is connected to the other neighbor groups. A particle i in a group $g(i)$ has the personal best solution $pbest_i$, and the group $g(i)$ has the local best solution $lbest_{g(i)}$. Each group has the neighbor groups characterized by a constant Degree Between each Group (DBG) and each group has the group-local best solution ($glbest$) shared by their groups. The network topology can be changed by adjusting DBG. When a particle i in a group $g(i)$ updates $lbest_{g(i)}$, the group $g(i)$ transmits the information of $lbest_{g(i)}$ to the neighbor groups. The group $g(i)$ and its neighbor groups update $glbest$ which each group independently has if the transmitted $lbest_{g(i)}$ is better than the

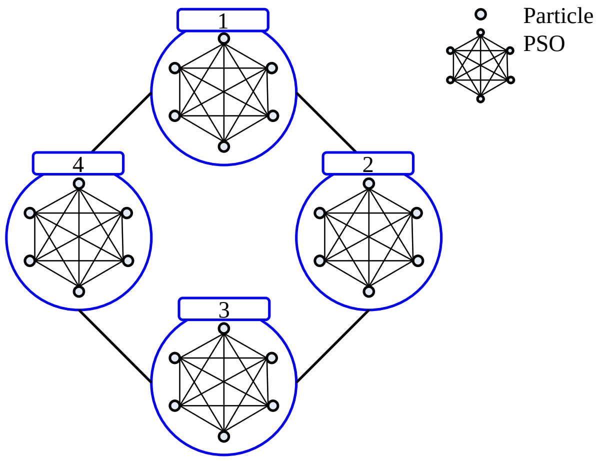

Figure 1: An example of a PSO Network (PSON-D)

current *glbest*. The ith particle updates the position vector and velocity vector by the following equation.

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (pbest_i^t - x_i^t) + c_2 r_2 (lbest_{g(i)}^t - x_i^t) + c_3 r_3 (glbest_{g(i)}^t - x_i^t) \quad (1)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (2)$$

where x_i^t and x_i^{t+1} denote the current and next position vectors of the ith particle, v_i^t and v_i^{t+1} denote the current and next velocity vectors of the ith particle, and t denotes the search iteration. ω is an inertia coefficient, c_1, c_2 and c_3 are acceleration coefficients, and r_1, r_2 and r_3 are uniform random numbers from 0 to 1. In PSON-D, even if one group traps into local minimum, the group can escape from the local minimum by the effect of $glbest_{g(i)}$. The algorithm of PSON are described by the following procedures.

Step1: Let $t = 0$. For all i, the vectors x_i^0 and v_i^0 are initialized at random.

Step2: For all i, the particle i updates $pbest_i^t$.

Step3: For all $g(i)$, the group $g(i)$ updates $lbest_{g(i)}^t$.

Step4: If a particle i in the $g(i)$th group updates $lbest_{g(i)}^t$, the group $g(i)$ communicates to the neighbor groups and their groups update each $glbest^t$.

Step5: For all i, the vectors x_i^{t+1} and v_i^{t+1} are updated by Equations (1) and (2).

Step6: Let $t = t + 1$. Step 2 to 5 are repeated until $t = t_{max}$.

If the number of groups is 1 and *glbest* is not used, PSON-D corresponds to the standard PSO. If *DBG* decreases, the propagation of *glbest* becomes slower. While *DBG* increases, the propagation of *glbest* becomes faster.

3. PSO networks with temporal couplings (PSON-T)

In this section, temporal couplings are introduced to our PSO network. This algorithm is referred to as the PSO Network with Temporal couplings (PSON-T).

In PSON-T, each particle does not have the group-local best solution (*glbest*) in the neighbor groups but has the global best solution (*gbest*) in all the groups. In addition, each group temporally refers to *gbest*. If a group does not refer to *gbest*, the group operates as same as the independent PSO of the single swarm. PSON-T is described by the following procedures.

Step1: Let $t = 0$. For all i, the vectors x_i^0 and v_i^0 are initialized at random.

Step2: For all i, the particle i updates $pbest_i^t$.

Step3: For all $g(i)$, the group $g(i)$ updates $lbest_{g(i)}^t$.

Step4: For all $g(i)$, the random number $r_{g(i)}$ is obtained. If $r_{g(i)} \leq C$, the group $g(i)$ can update $gbest^t$. If $r_{g(i)} > C$, the group $g(i)$ does not update $gbest^t$.

Step5: For all i, the vectors x_i^{t+1} and v_i^{t+1} are updated as follows.

$$v_i^{t+1} = \begin{cases} \omega v_i^t + c_1 r_1 (pbest_i^t - x_i^t) + c_2 r_2 (lbest_{g(i)}^t - x_i^t) + c_3 r_3 (gbest^t - x_i^t) & r_{g(i)} \leq C \\ \omega v_i^t + c_1 r_1 (pbest_i^t - x_i^t) + c_2 r_2 (lbest_{g(i)}^t - x_i^t) & r_{g(i)} > C \end{cases} \quad (3)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (4)$$

where C is the communication rate parameter.

Step6: Let $t = t + 1$. Step 2 to 5 are repeated until $t = t_{max}$.

If the parameter C is small, the number of communications can be reduced. In addition, it is expected that the diversity of the solutions can be generated; the search performances can be improved.

4. Numerical experiments

In order to confirm the effectiveness of the proposed PSON-D and PSON-T, the numerical experiments are performed.

The four benchmark functions shown in Table 1 are used in the experiments. F_2 (Sphere function) and F_3 (Rosenbrock function) are unimodal functions. F_1 (Rastrigin function) and F_4 (Griewank function) are multimodal functions with numerous local minimum. The optimum solution of F_3 is $\mathbf{x} = (1, 1, 1, \cdots, 1)$, and the optimum solutions of the others functions are $\mathbf{x} = (0, 0, 0, \cdots, 0)$. The optimum evaluation values of all the functions are $F(\mathbf{x}) = 0$.

Each algorithm is applied to the above four functions whose dimensions are 30. For all the experiments, the number of iterations $t_{max}$is 30000 and the number of trials with

Table 1: Benchmark Functions.

Function	Range
$F_1(\mathbf{x}) = 10D + \sum_{d=1}^{n} \left((x_d)^2 - 10\cos(2\pi x_d) \right)$	$x_d \in [-5.12 : 5.12]$
$F_2(\mathbf{x}) = \sum_{d=1}^{n} x_d^2$	$x_d \in [-5.12 : 5.12]$
$F_3(\mathbf{x}) = \sum_{d=1}^{n} \left(100(x_{d+1} - x_d^2)^2 + (1 - x_d)^2 \right)$	$x_d \in [-2.048 : 2.048]$
$F_4(\mathbf{x}) = 1 + \frac{1}{4000} \sum_{d=1}^{n} x_d^2 - \prod_{d=1}^{n} \cos\left(\frac{x_d}{\sqrt{d}} \right)$	$x_d \in [-600 : 600]$

Table 2: Parameters in PSO-D

	PSON-D
No. of groups	8
No. of particles	20
DBG	2, 4, 7
ω	0.729
c_1	1.4955
c_2	1.4955
c_3	0.1955

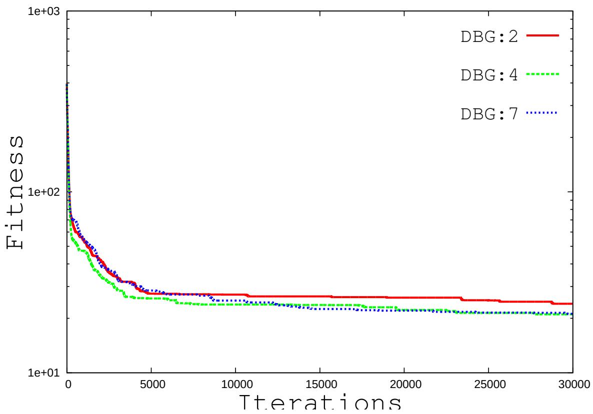

Figure 2: Simulation results for PSON-D (Function: F_1).

different random initial values is 32. For each algorithm, 30 experimental values are obtained by removing the the best value and the worst value from 32 trials.

First, the number of groups is fixed to 8, and DBG is varied to 2, 4, and 7. $DBG = 2$ and $DBG = 7$ correspond to the ring topology and star topology, respectively. Then, the solving performances of PSON-D are investigated, and the number of communications is evaluated. Table 2 shows the parameter in the experiments. Figure 2 and Table 3 show the simulation results. In Figure 2, the horizontal axis is the number of iterations, and the vertical axis is the average experimental values. In Table 3, "Best" means the best experimental value, "Worst" means the worst experimental value, "Ave." means the average experimental value. "Comm." means the total number of communications.

Table 3: Simulation results for PSON-D.

DBG	Best	Worst	Ave.	Comm.
		F_1		
2	1.39×10^{1}	3.48×10^{1}	2.41×10^{1}	1.02×10^{4}
4	1.19×10^{1}	2.98×10^{1}	2.11×10^{1}	1.30×10^{4}
7	1.19×10^{1}	2.79×10^{1}	2.12×10^{1}	1.71×10^{4}
		F_2		
2	0	0	0	1.06×10^{5}
4	0	0	0	1.94×10^{5}
7	0	0	0	3.29×10^{5}
		F_3		
2	4.33×10^{-17}	1.20×10^{-14}	1.27×10^{-15}	3.71×10^{5}
4	4.40×10^{-18}	3.72×10^{-15}	2.93×10^{-16}	1.94×10^{5}
7	2.64×10^{-19}	4.05×10^{-15}	2.49×10^{-16}	1.46×10^{6}
		F_4		
2	0	1.11×10^{-16}	3.70×10^{-18}	6.72×10^{3}
4	0	0	0	1.04×10^{4}
7	0	0	0	1.65×10^{4}

Table 4: Parameters in each algorithm

	SPSO	PSON-D	PSON-D	PSON-T
No. of groups	1	8	8	8
No. of particles	160	20	20	20
DBG	–	2	7	–
ω	0.729			
c_1	1.4955			
c_2	1.4955			
c_3	–	0.1955		1.9955
C	–			0.01

Table 5: Simulation results for each algorithm

	Best	Worst	Ave.	Comm.
		F_1		
SPSO	4.88×10^{1}	1.23×10^{2}	8.69×10^{1}	–
PSON-T	0	9.94×10^{-1}	3.30×10^{-2}	2.20×10^{3}
		F_2		
SPSO	0	0	0	–
PSON-T	0	1.98×10^{-323}	0	7.26×10^{3}
		F_3		
SPSO	7.13×10^{-11}	1.25×10^{2}	1.68×10^{1}	–
PSON-T	4.58×10^{-14}	5.55×10^{-7}	3.08×10^{-8}	7.13×10^{3}
		F_4		
SPSO	0	4.40×10^{-2}	9.85×10^{-3}	–
PSON-T	0	0	0	5.65×10^{2}

As shown in these results, when the parameter DBG increases, the number of communications increases. However, the search performances are improved a little. From these results, DBG should be small value.

Next, PSON-T is compared with SPSO, PSON-D ($DBG = 2$) and PSON-D ($DBG = 7$). Table 4 shows the parameters in the experiments. Table 5 show the simulation results for SPSO and PSON-T. The simulation results for PSON-D can be found in Table 3. Figure 3 shows the

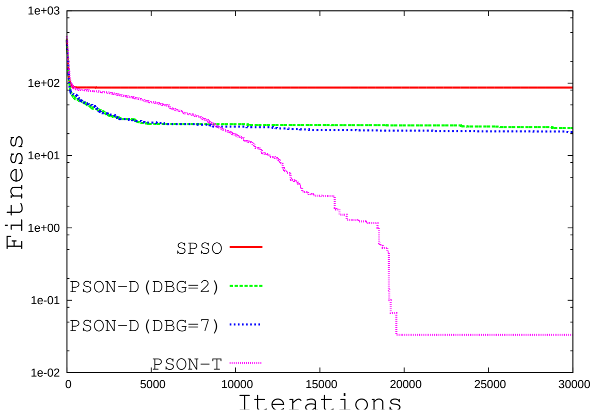

Figure 3: Simulation results for each algorithm (Function: F_1.)

comparison results for the function F_1. As shown in these results, PSON-T has better solving performances than the other algorithms. In the Function F_1, the solutions of SPSO and PSON-D are trapped into the local minima. However, PSON-T can escape from the local minima by temporally referring to *gbest*. In PSON-T, the communication rate parameter C is fixed 0.01. Then, each sub-swarm group independently searches solutions in the most iterations. When a sub-swarm group is trapped into the local minima, the sub-swarm group can move from the local minima by temporally referring to *gbest*. Moreover, since each sub-swarm group does not frequently communicate to each other, the number of communications can be reduced. The above results show that PSON-T has the good solving performances with reducing the overhead for interprocess communications.

5. Conclusions

This paper has proposed two PSO networks (PSON-D and PSON-T). In order to confirm the effectiveness of our new algorithm, the numerical experiments have been performed. First, in PSON-D, the parameter of *DBG* has been varied. In this simulation, the smaller *DBG* leads to reduce the number of communications. However, it is necessary to investigate effects to solving performances by changing *DBG*, in more detail. Next, PSON-T has been compared with a standard PSO, and PSON-D. In this simulation, PSON-T shows better performances than the other algorithms. Also, PSON-T can reduce the number of interprocess communications. This algorithm can effectively solve various optimization problems by using multiple processor systems.

Future problems include the more improvements of solving performances, and the implementation on real multiple processor systems.

References

[1] J.Kennedy and R.Eberhart, "Particle Swarm Optimization", Proc. IEEE Int. Conf. Neural Networks, pp. 1942-1948, 1995 near Theory and its Applications, pp. 197-200, 2010.

[2] X.Lai and G.Tan, "Studies on Migration Strategies of Multiple Population Parallel Particle Swarm Optimization", ICNC, pp. 798-802, 2012

[3] H.Nakano, Y.Taguchi, Y.Kanamori, A.Utani, A.Miyauchi, H.Yamamoto, "A Competitive Particle Swarm Optimizer and its Application to Wireless Sensor Networks", IEEJ Trans., 7, S1, pp. 52-58, 2012

[4] G.G.Yen and M.Daneshyari, "Diversity-based Information Exchange among Multiple Swarm in Particle Swarm Optimization", Proc. IEEE CEC, pp. 6150-6157, 2006

[5] M.Iwamatsu, "Multi-Species Particle Swarm Optimizer for Multimodal Function Optimization", IEICE Trans. Inf. & Syst., vol. E89-D, no. 3, 2006

[6] T.Tsujimoto, T.Shindo, T.Kimura, K.Jin'no, "A Relationship between Network Topology and Search Performance of PSO", Proc. of IEEE CEC, pp. 1526-1531, 2012

[7] E.Miyagawa and T.Saito, "Particle Swarm Optimizers with Growing Tree Topology", IEICE Trans. Fundamentals, E92-A, pp.2275-2282, 2009

[8] S.B.Akat and V.Gazi, "Particle Swarm Optimization with Dynamic Neighborhood Topology:Three Neighborhood Strategies and Preliminary Results", Proc. of IEEE Swarm Intelligence Symposium, pp. 1-8, 2008

[9] J.Lane, A.Engelbrecht and J.Gain, "Particle Swarm Optimization with Spatially Meaningful Neighbors", Proc. of IEEE Swarm Intelligence Symposium, pp. 1-8, 2008

[10] R.Sano, T.Shindo, K.Jin'no, T.Saito, "Particle Swarm Optimization with Switched Topology", Proc. of IEEE SMC, pp. 530-535, 2012

[11] H.Matsushita, Y.Nishio and T.Saito, "Particle Swarm Optimization with Novel Concept of Complex Network", in Proc. of Int. Symposium on Nonli