2010 IEEE Workshop on
Nonlinear Circuit Networks, Tokushima, Dec 3-4, 2010

Particle Swarm Optimization Containing Shared Velocity

Masaki SUGIMOTO
Dept. Electrical and Electronic Eng.,
Tokushima University
Email: sugimoto@ee.tokushima-u.ac.jp

Haruna MATSUSHITA
Dept. Electrical and Electronics Eng.,
Hosei University
Email: haruna.matsushita.pd@k.hosei.ac.jp

Yoshifumi NISHIO
Dept. Electrical and Electronic Eng.,
Tokushima University
Email: nishio@ee.tokushima-u.ac.jp

***Abstract*— In this study, we propose PSO containing shared velocity (PSOV). The important feature of PSOV is that each particle of PSOV shares the same velocity information. We investigate the behavior of PSOV and confirm its efficiency.**

I. Introduction

Particle Swarm Optimization (PSO) [1] is a popular optimization technique for solving objective functions and PSO is an evolutionary algorithm to simulate the movement of flocks of birds toward foods. Due to its simple concept, easy implementation and quick convergence, PSO has attracted attentions and has been widely applied to different fields in recent years. Furthermore, PSO has demonstrated great performances for many problems. However, quick convergence often leads to local optimum problem. It is important for multimodal functions with a lot of local optima to compromise between the quick convergence and being trapped in a local optimum. In order to escape from such local optima and to avoid the premature convergence, the search for a global optimum should be diverse.

In this study, we propose an improved PSO algorithm; PSO containing cooperative particles (called PSOV).

We explain the algorithm of PSOV in detail in Section II. In Section III, we perform basic numerical experiments by using two algorithm methods of PSO and PSOV. Furthermore, we confirm the efficiency of PSOV.

II. PSOV

[PSOV1] (Initialization) Let a generation step $t = 0$. Randomly initialize the particle $i = (1, 2, ..., M)$ position $\boldsymbol{X}_i = (x_{i1}, x_{i2}, ..., x_{iD})$ and its velocity $\boldsymbol{V}_i = (v_{i1}, v_{i2}, ..., v_{iD})$ for all particles i and initialize $\boldsymbol{P}_i = (p_{i1}, p_{i2}, ..., p_{iD})$ with a copy of $\boldsymbol{X}_i$.

[PSOV2] Evaluate the current cost $f(\boldsymbol{X}_i)$. Update the personal best position *pbest* $\boldsymbol{P}_i = (p_{i1}, p_{i2}, ..., p_{iD})$ for each particle i and the global best position *gbest* $\boldsymbol{P}_g = (p_{g_1}, p_{g_2}, ..., p_{g_D})$ among the all particles.

[PSOV3] Update $\boldsymbol{V}_i$ of each particle i depending on its *pbest* and its swarm best *gbest*;

$$v_{id}(t+1) = wv_{id} \quad +c_1 r_1 \{p_{id} - x_{id}(t)\} \\ \qquad\qquad +c_2 r_2 \left\{p_{g_d} - x_{id}(t)\right\}, \tag{1}$$

where r_1 and r_2 are two random variables distributed uniformly on $[0, 1]$, w is an inertia weight of all particles, and c_1 and c_2 are positive acceleration coefficients.

[PSOV4] All the particles shared $\boldsymbol{V}_c = (v_{c_1}, v_{c_2}, ..., v_{c_D})$. Let $\boldsymbol{V}_c$ represent the average velocity of all the particles;

$$v_{c_d} = \frac{1}{M}\sum_{i=1}^{M} v_{id} \tag{2}$$

[PSOV5] Update $\boldsymbol{X}_i$ depending on its $\boldsymbol{V}_c$ and $\boldsymbol{V}_i$;

$$x_{id}(t+1) = x_{id}(t) + c_v v_{c_d}(t+1) + v_{id}(t+1), \tag{3}$$

where c_v is a cooperation coefficients. In other words, the particles combine the action of individual and cooperation.

[PSOV6] Let $t = t + 1$. Go back to [PSOV2], and repeat until $t = T$.

III. Numerical Experiments

In order to confirm the performance of PSOV algorithm, we have performed basic numerical experiments. The problem is finding the optimum (minimum) value of $f(x)$ in the algorithm. Referring to [2], we use the following four bench marks.

1. Sphere function:

$$f_1(x) = \sum_{d=1}^{D-1} x_d^2, \tag{4}$$

where $x \in [-2.048, 2.047]^D$ and the optimum solution x^* are all $[0, 0, \ldots, 0]$.

2. Rosenbrock's function:

$$f_2(x) = \sum_{d=1}^{D-1}(100(x_d^2 - x_{d+1})^2 + (1 - x_d)^2), \tag{5}$$

where $x \in [-2.048, 2.047]^D$ and the optimum solution x^* are all $[1, 1, \ldots, 1]$.

3. Rastrigin's function and its optimum (minimum):

$$f_3(x) = \sum_{d=1}^{D}(x_d^2 - 5\cos(2\pi x_d) + 5), \tag{6}$$

where $x \in [-5.12, 5.12]^D$ and the optimum solution x^* are all $[0, 0, \ldots, 0]$. We consider that an almost optimum value

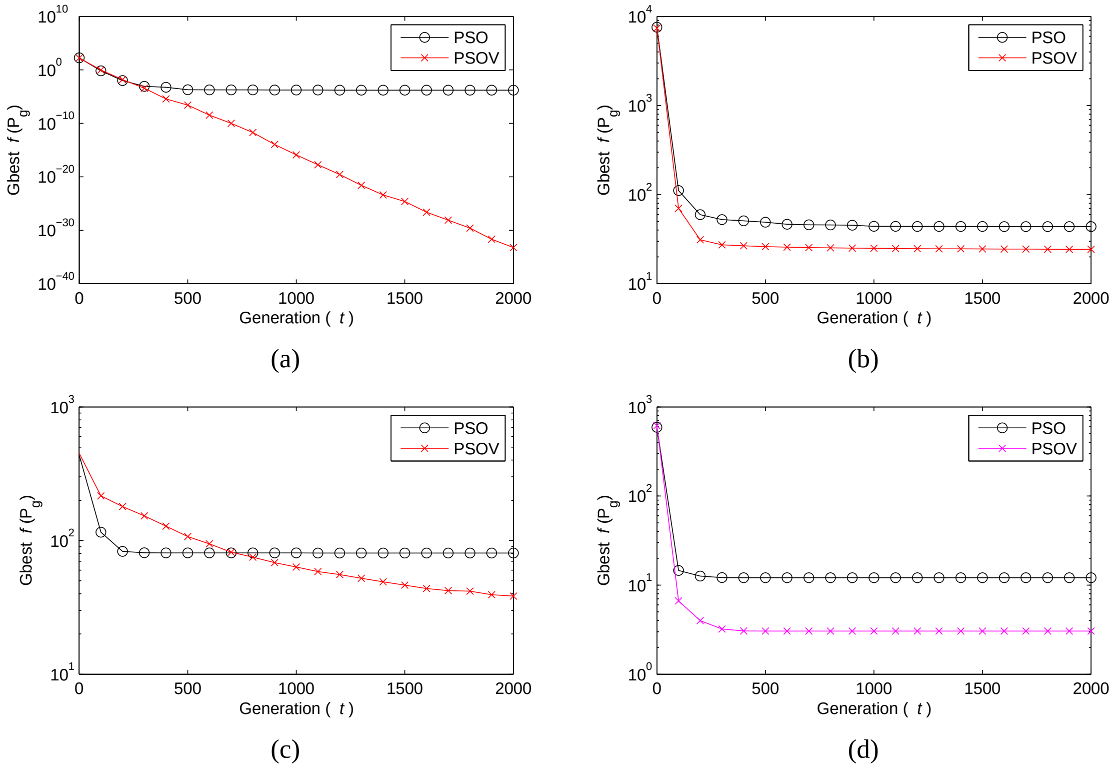

Fig. 2. Mean *gbest* value of every generation for 30-dimensional four functions. (a) Sphere function. (b) Rosenbrock's function. (c) Rastrigin's function. (d) Griewank's function.

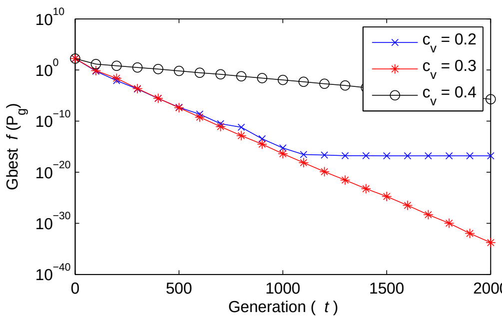

Fig. 1. The c_v of various value is compared. (Sphere function, $D = 30$)

is obtained if the algorithm attains the criterion $f_1(x) = 100$. This criterion is based on [2]. The range of initialization is $-5.12 \leq x_i \leq 5.12$.

4. Griewank's function:

$$f_4(x) = \sum_{d=1}^{D} \frac{x_d^2}{4000} + \prod_{d=1}^{D} \cos\left(\frac{x_d}{\sqrt{d}}\right) + 1, \qquad (7)$$

where $x \in [-600, 600]^D$ and the optimum solution x^* are all $[0, 0, \ldots, 0]$.
The optimum function values $f(x^*)$ of all functions are 0. f_1 and f_2 are unimodal functions, and f_3 and f_4 are multimodal functions with numerous local minima. All the functions have D variables. In this study, D is set to 30 and 100 to investigate the performances in various dimensions. The population size M is set to same the D. The inertia weight is fixed as $w = 0.5$. For PSO and PSOV, the acceleration coefficients are set as $c_1 = c_2 = 1.8$. We experimented in how many influences it has on a simulation by changing the value of c_v We compare PSOV in the c_v, Sphere function was used by D = 30 this time. Because the best result was brought from Figure. 1 at the time of $c_v = 0.3$, $c_v = 0.3$ is used in the following simulation. For PSOV, the cooperation coefficients are set as $c_v = 0.3$.

We carry out the simulation 30 times for all the optimization functions with 2000 generations, namely $T = 2000$. The performances with minimum, maximum and mean function values on four functions with 30-dimension are listed in Table I. We can see that the PSOV can obtain the best

TABLE I

COMPARISON RESULTS PSO AND PSOV ON TEST FUNCTIONS WITH $D = 30$.

f		Avg.	Min.	Max.
f_1	PSO	1.67E-04	3.33E-35	4.73E-03
	PSOV	**5.92E-34**	3.18E-39	1.69E-32
f_2	PSO	43.8	20.2	132.1
	PSOV	**24.3**	21.2	28.4
f_3	PSO	80.6	46.8	117.4
	PSOV	**38.4**	19.9	130.7
f_4	PSO	1.21E+01	3.00E-15	9.09E+01
	PSOV	**3.04**	0	9.07E+01

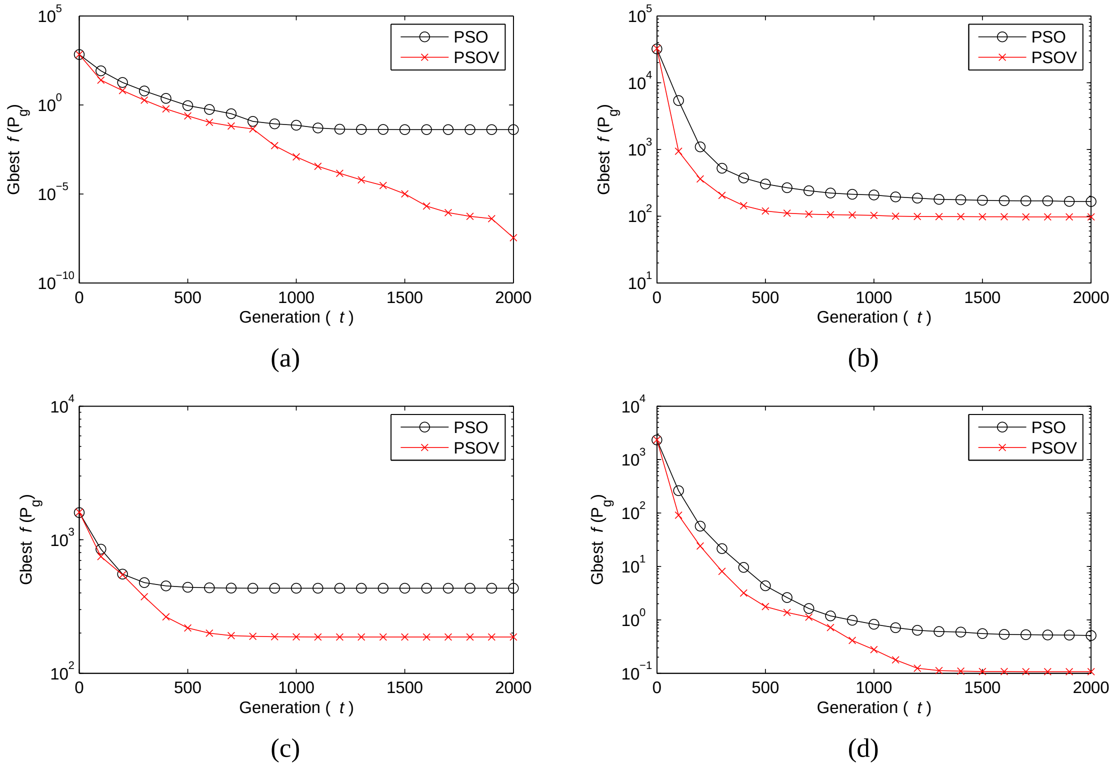

Fig. 3. Mean *gbest* value of every generation for 100-dimensional four functions. (a) Sphere function. (b) Rosenbrock's function. (c) Rastrigin's function. (d) Griewank's function.

TABLE II

COMPARISON RESULTS PSO AND PSOV ON TEST FUNCTIONS WITH D = 100.

f		Avg.	Min.	Max.
	PSO	4.12E-02	4.68E-09	1.18
f_1	PSOV	**3.52E-08**	6.13E-10	3.90E-07
	PSO	165.85	88.67	251.85
f_2	PSOV	**97.60**	92.64	151.41
	PSO	432.84	358.18	603.94
f_3	PSOV	**186.85**	104.47	270.63
	PSO	5.09E-01	2.28E-06	4.13
f_4	PSOV	**1.07E-01**	2.28E-07	1.15

mean values for all the test functions. Next, in case of the performances on the test functions with D = 100 shown in Table II, PSOV can obtain the best mean values for all the test functions.

Figs. 2 shows the mean *gbest* values of every generation over 30 runs for four test functions with 30 dimensional. From these results, we can see, PSOV can obtain the best results. However, the performances of PSO for the Rastrigin's functions f_3 as Figs. 2(c) is more good value until 500 generation. From this results, we have investigated that shared velocity is helping to escape from local minima by the block the quick convergence.

Meanwhile, the mean *gbest* values for 100 dimensional test functions are shown in Fig. 3. In whole generation steps, the performances of PSOV is the good values in all the functions. Because it is difficult for PSO to find the optimum solution, PSO are easily trapped in the local optima and prematurely converge.

From these results, we can confirm that the mean values of PSOV are the best among four problems. Therefore, we can confirm that PSOV algorithm is more effective than standard PSO.

IV. CONCLUSIONS

In this study, we have proposed a new Particle Swarm Optimization algorithm (PSOV) containing the containing shared velocity. We have performed the basic numerical experiments for various dimensions by using the PSO and PSOV. From these results, we have confirmed that PSOV could obtain the effective.

REFERENCES

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," *Proc. IEEE Int. Conf. on Neural Networks*, vol. 4, pp. 1942–1948, 1995.

[2] J. Kennedy and R. Medes, "Population structure and particle swarm performance," *Proc. IEEE Congress on Evolutionary Computation*, vol. 2, pp. 1671–1676, 2002.