Subpopulation best rotation: a modification on PSO

Jorge Barrera Alviar[a*]*, Jorge Peña*[b,] *Roberto Hincapié* [c]

[a]Facultad de Ingeniería. Universidad de los Andes. Carrera 1 N.º 18ª-10 Bogotá, Colombia.

[b]Faculté des sciences sociales et politiques (SSP), Institut de Mathématiques Appliquées (IMA), Université de Lausanne, Suisse Unicentre-CH 1015 Lausanne, Suisse.

[c]Grupo de Investigación, Desarrollo y Aplicación en Telecomunicaciones e Informática (GIDATI). Universidad Pontificia Bolivariana. Circular 1 N.º 70-01. Medellín, Antioquia.

Abstract

This paper deals with a modification on Particle Swarm Optimization (PSO), an original topology whose use can be justified to optimize multimodal functions. The analysis is further verified by some proofs, using different benchmark functions with asymmetric initialization. The method is optimistic and may be a starting point for further discussions.

---------- *Key words*: Particle Swarm Optimization, evolutionary computation, test functions, neighborhood topology.

Rotación de las mejores partículas de las subpoblaciones: una modificación en PSO

Este artículo trata sobre una modificación hecha al método de optimización por enjambre de partículas (PSO), que consiste en una topología original cuyo uso se puede justificar para funciones multimodales. El análisis se verifica con algunas pruebas, usando diferentes funciones de prueba con inicialización asimétrica. El método es optimista y puede ser un punto de partida para futuras discusiones.

---------- *Palabras clave:* optimización por enjambre de partículas, computación evolutiva, funciones de prueba, topología de vecindario.

∗   Autor de correspondencia: teléfono: +57+1+ 339 99 99. Correo electrónico: jorgeb49@yahoo.es (J. Barrera)

Introduction

PSO is an optimization technique inspired by the social behavior of some species and supported by evolutive psychology which suggests that sociocognitive individuals (individuals that know through their own experience and as well as their society experience), must be influenced by their previous behavior and the success of their neighbors [1].

Neighborhood Topologies describe the social structure that makes it possible interaction between individuals within a population. The structure of a social network significantly affects the group performance. In PSO where the individuals or particle behavior can be summarized in three essentials: evaluate, compare and imitate. The method for interactions between particles makes the algorithm work good, poorly or not work at all.

There are two main topologies used in PSO, lbest (ring topology) and gbest (star topology). In a gbest topology each individual knows the performance of all the others, being able to know which one is the best (gbest), on the other hand, in lbest topology, each individual knows the performance of its k topological closer neighbors [1].

Migration is a common technique on genetic algorithms which allow different populations to exchange information by giving the individuals some probability to travel from one to another population. Migration has been widely used for improving genetic algorithms and some others optimization techniques.

The modification suggested is to use some kind of migration in order to create a modified topology. Our proposal is to work with different populations which share no information primarily but achieving interpopulation interaction by the exchange of their best particle.

This paper describes the standard PSO algorithm (SPSO) and the modifications proposed to SPSO are described to facilitate the implementation of the algorithm, further we explain the test functions used to evaluate the algorithm, and the results are exposed and discussed, finally some conclusions are presented.

Standard PSO (SPSO)

PSO explores a D-dimensional space, using a population of particles which are initially provided with random velocity and position in the problem space [2]. Each particle represents a suggested solution and has two kinds of available information, the first kind is about the knowledge of its own experience and the second kind is about the experience of individuals among the whole population [1].

Each particle has a position in the problem space $x_i = (x_{i1}, x_{i2},...,x_{iD})$, velocity $v_i = (v_{i1}, v_{i2},...,v_{iD})$, and a memory with its best previous position $p_i = (p_{i1}, p_{i2},...,p_{iD})$. In every iteration for each population, the i particle, whose p_i obtains a better fitness, is designated as g, and for each iteration, the p_i and p_g vectors are used to modify the position of particle i this way:

$$v_{id} = \text{w}.v_{id}, + c_1.rand().(p_{id} - x_{id}) + c_2.Rand().p_{gl} - x_{id}) \tag{1}$$

$$x_{id} = x_{id} + v_{id} \tag{2}$$

c_1 is known as cognitive factor and $c2$ as social factor, these define the relative influence of the individual and social behavior in the particle movement. w is known as inertial weight and its function is to control the impact of previous velocity on particle movement. Rand() and rand() are two different random numbers between 0 and 1.

It has been found that a group of values that provide the method with great performance in almost all problems is:

- $w = 0.4$

- $c_1 = c_2 = 2$

PSO implementation is as follows [2]:

1. Assign iteration Gc = 1.

2. Initialize population by assigning each parti-cle a random position and velocity like this:

   $x_{id} = xmin + Rand() . (2.xmax)$

   $v_{id} = Rand3() . (2.vmax)$

Where:

- $Rand3()$ is a random number, and

   $-1 \leq Rand3() \leq 1$

- $xmin, xmax$ and $vmax$ depends on the problem to optimize.

3. Evaluate particle fitness.

4. Update all of the p_i.

5. Update g.

6. Change velocity and position for all particles using Eq. 1 and Eq. 2

7. Implement velocity damping for all of the particles:

   If $v_{id} > v_{man}$ then $v_{id} = v_{max}$

   If $v_{id} < -v_{man}$ then $v_{id} = -vmax$

8. Gc = Gc+1

9. If stop criterion is not reached jump to step 3.

Best rotation PSO (BRPSO)

If PSO uses social knowledge to make the system convergent into a solution it would seem unacceptable to separate particles into almost non communicated subpopulations, and that is true if the problem we are dealing is a monomodal function. These are functions with no other minima than the global one, but in multimodal functions the wide knowledge of the whole population performance make the system converge too fast and also increase the probability of stagnation into local minima.

Best rotation is easy to execute and finds very good solutions for multimodal functions optimization. Its implementation consists on a periodically rotation of the best particle of each subpopulation, in order to specify the frequency used to rotate the best individuals of each population. *lc* is used to denote how many iterations there will be between rotation and rotation, *npo* is used to denote the number of subpopulations.

BRPSO can be seen as an extra step between steps 5 and 6 of the algorithm SPSO, like this: If (Gc%lc == 0) Then rotate best individuals

By rotate best individuals it must be understood that i^{th} population must have the best particle of the next population instead of its own original best particle and that the last population must have the best particle of the 1st population instead of its own original best particle.

When best rotation is executed, stagnation on local minima is avoided by forcing populations to move from one local minimum to another one, increasing the exploration of the problem space between different local minima.

Results and discussion

In order to prove the algorithm proposed we used three functions widely used in optimization literature [1], [2], [3].

The function *f1* is the Rosenbrock function:

$$f1\left(\vec{x}\right) = \sum_{d=1}^{D} 100\left(x_{d+1} - x_d^2\right)^2 + (x_d - 1)^2 \qquad (3)$$

The function *f2* is the generalized Rastrigin function:

$$f2\left(\vec{x}\right) = \sum_{d=1}^{D} x_i^2 + 10.[1 - 2.\cos(2\pi.x_d)] \qquad (4)$$

The function *f3* is the generalized Griewank function:

$$f3\left(\vec{x}\right) = \frac{1}{4000} \sum_{d=1}^{D} x_d^2 - \prod_{d=1}^{D} \cos\left(\frac{x_d}{\sqrt{d}}\right) + 1 \qquad (5)$$

For testing each function we used asymmetrical initialization defined by:

For $f1$: $x_i \in (15, 30)$ with $i = 1, 2,..., D$

For $f1$: $x_i \in (2.56, 5.12)$ with $i = 1, 2,..., D$

For $f1$: $x_i \in (300, 600)$ with $i = 1, 2,..., D$

The values used to make velocity damping and define problem space are as follows:

For $f1$: $x\text{max} = v\text{max} = 100$

For $f2$: $x\text{max} = v\text{max} = 10$

For $f3$: $x\text{max} = v\text{max} = 600$

For every function we tested with 10, 20, 30 dimensions, and 20, 40 and 80 particles for each dimension size. All of the values registered in tables show the mean value of 500 trials. For next tables $lc = 50$, $w = 0.4$ and

$$npo = \frac{m}{10}$$

where m is the number of particles, therefore each population has 10 particles and there are as many populations as groups of ten particles are possible.

In tables 1, 2 and 3, the SPSO results are taken from [4].

Table 1 Mean fitness values for the Rosenbrock function

Pop. Size	D	Gmax	SPSO	BRPSO
	10	1.000	44.1374	45.425
20	20	1.500	87.281	139.4881
	30	2.000	132.5973	92.45
	10	1.000	24.3512	9.0649
40	20	1.500	47.7243	48.0761
	30	2.000	66.6341	92.46
	10	1.000	15.3883	3.4762
80	20	1.500	40.6403	17.2439
	30	2.000	63.4453	39.4554

Table 2 Mean fitness values for the Rastrigin function

Pop. Size	D	Gmax	SPSO	BRPSO
	10	1.000	9.9483	7.7346
20	20	1.500	44.3457	40.7703
	30	2.000	97.0551	95.2303
	10	1.000	5.6853	3.5692
40	20	1.500	29.5543	23.9551
	30	2.000	68.0280	56.9706
	10	1.000	3.5988	2.0283
80	20	1.500	20.6500	14.768
	30	2.000	49.344	41.1001
Pop. Size	D	Gmax	SPSO	BRPSO
	10	1.000	0.0203	0.0833
20	20	1.500	0.03174	0.0316
	30	2.000	0.0482	0.0162
	10	1.000	0.07617	0.0479
40	20	1.500	0.02272	0.0097
	30	2.000	0.01527	0.0043
	10	1.000	0.06581	0.0322
80	20	1.500	0.02217	0.0031
	30	2.000	0.01208	0.0009

Table 3 Mean fitness values for the Griewank function

Pop. Size	D	Gmax	SPSO	BRPSO
	10	1.000	0.0203	0.0833
20	20	1.500	0.03174	0.0316
	30	2.000	0.0482	0.0162
	10	1.000	0.07617	0.0479
40	20	1.500	0.02272	0.0097
	30	2.000	0.01527	0.0043
	10	1.000	0.06581	0.0322
80	20	1.500	0.02217	0.0031
	30	2.000	0.01208	0.0009

For these multimodal functions some delay on the convergence increase the capability of the algorithm to optimize the function, and exploration around local minimum helps particles to find even better solutions. It is visible that as more particles are used, the improvement of the best rotation technique is increased.

Conclusion

This paper has explored a new modification on standard PSO, designed to acquire a better performance when optimizing multimodal functions. The best rotation technique delays convergence of the system, prevents local stagnation, and achieves more exploration of the problem space. These characteristics make BRPSO

good for multimodal functions optimizing, but unnecessarily slow for easier testing functions.

References

1. J. Kennedy, R. Eberhart, R. Swarm. *Intelligence*. San Francisco: Morgan Kaufman Publishers. 2001. pp. 287-360.

2. X. Xie, W. Zhang, Z. Yang. "Hybird Particle Swarm Optimizer with Mass Extinction". In: *International Conference on Communications, Circuits and Systems (ICCCAS)*. Chengdu, China. Vol. 2. 2002. pp 1170-1173.

3. A. Carlisle, G. Dozier. "An off-the-Shelf PSO". In: *Proceedings of the workshop on particle swarm optimization*. Indianapolis. Vol. 1. 2001. pp 1-6.

4. X. Xie, W. Zhang, Z. Yang. "Adaptive Particle Swarm Optimization on Individual Level". En: *International Conference on Signal Processing*. Beijing, China. Vol. 4. 2002. pp. 1215-1218.