Jiuyong Li \(Ed.\), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010

Hybrid Particle Swarm Optimisation Algorithms Based on Differential Evolution and Local Search

Wenlong Fu[1], Mark Johnston[1], and Mengjie Zhang[2]

[1] School of Mathematics, Statistics and Operations Research
[2] School of Engineering and Computer Science
Victoria University of Wellington, P.O. Box 600, Wellington, New Zealand

Abstract. Particle Swarm Optimisation (PSO) is an intelligent search method based on swarm intelligence and has been widely used in many fields. However it is also easily trapped in local optima. In this paper, we propose two hybrid PSO algorithms: one uses a Differential Evolution (DE) operator to replace the standard PSO method for updating a particle's position; and the other integrates both the DE operator and a simple local search. Seven benchmark multi-modal, high-dimensional functions are used to test the performance of the proposed methods. The results demonstrate that both algorithms perform well in quickly finding global solutions which other hybrid PSO algorithms are unable to find.

Keywords: Particle Swarm Optimisation, Differential Evolution.

1 Introduction

Particle Swarm Optimisation (PSO) is a stochastic global optimisation method which originated from the simulation of the social behaviour of birds within a flock, as developed by Kennedy and Eberhart in 1995 [1]. It is widely used in function optimisation [2], object detection [3], optimisation of wireless sensor networks [4], and many other applications [5,6].

The global optimisation of multi-modal functions is an important topic in scientific and engineering research since many real situations can be modelled as nonlinear optimisation problems. The standard PSO has difficulty with consistently converging to global optima, especially for multi-modal, high-dimensional functions. For escaping from local optima, Bratton and Blackwell [2] proposed a simplified recombinant PSO for function optimisation. Also, classical Differential Evolution (DE) operators have been integrated into hybrid PSO algorithms for global optimisation [8,9]. However, these PSO variants still have problems finding global solutions for some benchmark multi-modal, high-dimensional functions.

The goal of this paper is to investigate new hybrid PSO techniques for finding globally optimal solutions of multi-modal, high-dimensional functions. Instead of using the standard PSO method, we aim to use hybrid PSO techniques for updating a particle's position. We will consider two hybrid PSO approaches: using PSO with a DE operator (called HybridPSO1) rather than strongly depending on the currently global best and local best positions; and integrating

J. Li (Ed.): AI 2010, LNAI 6464, pp. 313–322, 2010.

Jiuyong Li \(Ed.\), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010

the DE operator with a local search (called HybridPSO2) to do a small amount of additional searching for a better position about the current position. Both approaches will be examined and compared with some existing PSO methods on seven benchmark multi-modal, high-dimensional functions. We will focus on whether the new approaches *can* find the global solutions for these functions, and investigate the performance of these approaches in converging to a global solution.

The goal here is to determine whether hybrid PSO can find global optima which allude other PSO-based methods. This should give some idea as to whether it would subsequently be worthwhile applying the proposed hybrid PSO methods to other benchmark sets of test problems including read-world problems. If we are able to demonstrate effectiveness then a serious comparison with state-of-the-art algorithms would subsequently be needed.

In the remainder of this paper, Section 2 briefly describes background on PSO and DE, and Section 3 describes both hybrid algorithms in detail. After presenting the experimental design in Section 4, Section 5 discusses the experimental results. Finally, Section 6 gives conclusions and future work directions.

2 Background

This section briefly describes necessary background information on Particle Swarm Optimisation and Differential Evolution.

2.1 Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) is a stochastic method for optimising without explicit knowledge of the gradient of the nonlinear function. PSO maintains a population of candidate solutions (called *particles*) and moves these particles around the search space. Each particle "flies" in a D-dimensional space according to the historical experiences of its own and its colleagues. Particle i has both a position, x_i, and a velocity v_i, which in "standard" PSO (SPSO), are updated as follows [10]:

$$v_{ik}^{t+1} = w \times v_{ik}^{t} + \phi_1 \times rand() \times (p_{ik}^{t} - x_{ik}^{t}) + \phi_2 \times rand() \times (g_k^t - x_{ik}^t) \quad (1)$$

$$x_{ik}^{t+1} = x_{ik}^{t} + v_{ik}^{t+1} \quad (2)$$

for component $k = 1, \ldots, D$. Here w is inertia weight; ϕ_1 and ϕ_2 are acceleration constants; $rand()$ are random values between 0 and 1; v_{ik}^t is the dimension k of the ith particle's velocity in generation t, v_i^t is the ith particle's velocity in generation t, and $v_i^t = [v_{i1}^t, v_{i2}^t, \ldots, v_{iD}^t]$; x_{ik}^t is the dimension k of the ith particle's position in generation t, x_i^t is the ith particle's position in generation t, and $x_i^t = [x_{i1}^t, x_{i2}^t, \ldots, x_{iD}^t]$; $p_i^t = [p_{i1}^t, p_{i2}^t, \ldots, p_{iD}^t]$ is the best position of the ith particle up to generation t, and $g^t = [g_1^t, g_2^t, \ldots, g_D^t]$ is the global best position of particles up to generation t. When termination criteria are satisfied, such as t being equal to the maximum generation, the global best position is taken as the solution to the problem.

Jiuyong Li \(Ed.\), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010

2.2 Differential Evolution

Differential Evolution (DE) is also a population-based optimisation algorithm. It has been applied to classical optimisation and multi-objective optimisation [7]. DE creates new candidate solutions by combining existing ones, via three evolutionary operators: mutation, crossover and selection. The classical DE (crossover) operator is given as:

$$v_i^t = x_{l1}^t + F(x_{l2}^t - x_{l3}^t) \tag{3}$$

$$x_{i,j}^{t+1} = \begin{cases} v_{i,j}^t & rand() < p_{cr} \\ x_{i,j}^t & \text{otherwise} \end{cases} \tag{4}$$

where $x_{l1}^t, x_{l2}^t, x_{l3}^t$ represent the position of three individual particles (candidate solutions) from the population at the tth generation; $x_{i,j}^t$ is the jth element (dimension) value of the ith individual in the population at the tth generation; F is the so-called scaling factor ($F \in [0, 2]$), and p_{cr} is called the crossover probability ($p_{cr} \in [0, 1]$). DE is similar to PSO in that they both feature interaction among individuals.

2.3 Related Work of Hybrid PSO for Multi-modal Functions

In recent years, researchers have proposed hybrid PSO variants to optimise multi-modal functions. Zhang et al [9] and Xin et al [8] both combined PSO and DE operators to search for global solutions of multi-modal functions. Akbari and Ziarati [11] introduced stochastic local search in PSO for multi-modal function optimisation. In those methods, their purposes were to improve particles' exploration ability. Xin et al [8] used a probability to select standard PSO or DE operators to control particle movement and maintain population diversity in case all particles plunge into a local optima. However, those methods still have problems with some multi-modal functions, such as the Generalised Rastrigin function (see formula (6) in Section 4). Since the global best position affects all particles in PSO and it is easily trapped in one of the local optima of the Generalised Rastrigin function, all particles tend to prematurely converge near the current best position (not necessarily the true global optima). There is still an issue of how to improve particles' ability of exploring the search space for multi-modal function optimisation.

Since PSO has difficulty escaping from a locally optimal position in multi-modal function optimisation problems, Bratton and Blackwell [2] proposed a simple model, removing the effect of the global best position, that has better performance than standard PSO (SPSO). Therefore, one strategy for improving the searching ability of PSO is changing the way to update particle positions and weakening the effect of the global best position.

3 New Hybrid PSO Algorithms

Since DE has similarities with PSO, we propose to replace the position update method used in SPSO with a DE operator. For exploring a better position in the

Jiuyong Li \(Ed.\), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010

Algorithm 1. HybridPSO1

1: Initialise the particles, the local best positions and the global best position.
2: Use the DE operator to update each particle's position using (3) and (4).
3: Update the current local best positions and the global best position.
4: If the maximum generation is reached, go to step 5; otherwise return to step 2.
5: Output the global best position particle as the solution.

neighbourhood of the current position, a local search operator is also introduced after a particle updates its position. The two hybrid algorithms are described in detail as follows.

3.1 Hybrid PSO Algorithm Based on Differential Evolution

The classical DE operator is introduced to update particle positions in this algorithm. The hybrid algorithm is called HybridPSO1. The particles positions are updated by (3) and (4), not by (1) and (2). In the algorithm HybridPSO1, we use three local best positions to construct new positions and then let each particle fly to the related new position if the new position is better than the current position. We save the global position in the memory and update its position if one new position is better than it. The global position will be returned as the global solution when the maximum generation is reached. The whole HybridPSO1 is described in Algorithm 1.

3.2 Hybrid PSO Algorithm Based on Differential Evolution and Local Search

In any PSO method, the behaviour of particles moving to the next position is discontinuous. It is possible that one particle cannot hit a better position in the current region. Based on HybridPSO1, we introduce a simple local search to better explore the neighbourhood of a local optima. The new hybrid PSO algorithm is called HybridPSO2. The local search is described in Algorithm 2, where x_i^k is the kth generation particle i. Particles are selected to update their position by the local search after arriving at a new position when the DE operation finishes. The local search is integrated in the hybrid PSO algorithm and the whole hybrid algorithm is described in Algorithm 3, where in step 3, only some particles arriving at new positions are selected to use the local search (with probability p_{local}).

3.3 Discussion

Both hybrid PSO techniques weaken the effect of the global best position on all particles. The way of using the DE operator here, in both hybrid PSO algorithms, is different from other hybrid PSO techniques based on DE [8,9]. Unlike these methods of alternating the use of standard PSO method and DE operators, both hybrid PSO techniques here directly replace the standard PSO method

Jiuyong Li \(Ed.\), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010

Algorithm 2. Local Search

1: Let $\Delta x_i^0 = x_i^{k+1} - x_i^k$ and $itr = 0$.
2: If $f(x_i^{k+1}) > f(x_i^k)$, let $\Delta x_i^0 = -\Delta x_i^0$.
3: If $itr < n_{itr}$, go to step 4; otherwise go to step 7.
4: $x_i' = x_i^{k+1} + \Delta x_i^{itr}$.
5: If $f(x_i') > f(x_i^{k+1})$, let $\Delta x_i^{itr+1} = \Delta x_i^{itr}/2$, otherwise $x_i^{k+1} = x_i'$;
6: $itr = itr + 1$; go to step 3.
7: Finish the local search.

Algorithm 3. HybridPSO2

1: Initialise the particles, the local best positions and the global best position.
2: Use the DE operator to update each particle's position using (3) and (4).
3: Randomly select some particles and perform local search on them using Algorithm 2.
4: Update the current local best positions and the global best position.
5: If the maximum generation is reached, go to step 6; otherwise return to step 2.
6: Output the global best position particle as the solution.

by the DE operator. Weakening the influence from the global best position is similar to the simplified recombinant PSO [2], but the method for updating particle positions is different. In [2], each particle is affected by its history and its neighbour's history, however, each particle in HybridPSO1 and HybridPSO2 is affected by all particles.

4 Experimental Design

We now describe the test functions and parameter settings for our experiments.

4.1 Multi-modal and High-Dimensional Functions

In practical optimal design problems, objective functions often lead to multi-modal domains. Multi-modal, high-dimensional functions often contain many local minima and a single global optimum. As Section 2 mentioned, the Generalised Rastrigin function contains many local minima (Figure 1 shows the two-dimensional Rastrigin function, i.e., with $D = 2$). These local optima make many PSO variants *fail* to find the global solution [2,8,9,11].

A standard set of seven benchmark multi-modal functions are employed to show the global optimisation performance of the proposed HybridPSO1 and HybridPSO2. These problems each contain many local minima and a single global optimum.

1. *Generalised Schwefel 2.6*

$$f_1 = -\sum_{i=1}^{D} x_i \sin(\sqrt{|x_i|}) \quad x_i \in [-500, 500] \tag{5}$$

Jiuyong Li \(Ed.\), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010

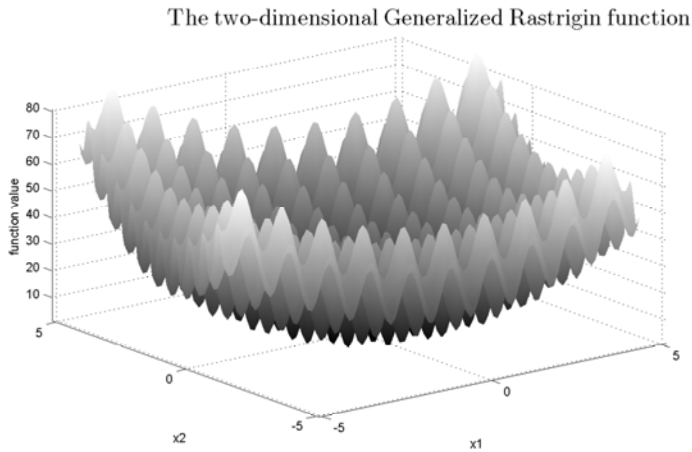


Fig. 1. The two-dimensional Generalized Rastrigin function

2. *Generalised Rastrigin*

$$f_2 = \sum_{i=1}^{D} \{x_i^2 - 10\cos(2\pi x_i) + 10\} \quad x_i \in [-5.12, 5.12] \tag{6}$$

3. *Ackley*

$$f_3 = -20\exp\left\{-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right\} - \exp\left\{\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right\} + 20 + e \quad x_i \in [-32, 32] \tag{7}$$

4. *Generalised Griewank*

$$f_4 = \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}}) + 1 \quad x_i \in [-600, 600] \tag{8}$$

5. *Penalised function P8*

$$f_5 = \frac{\pi}{D}\{10\sin^2(\pi y_1) + \sum_{i=1}^{D-1}\{1 + 10\sin^2(\pi y_{i+1}\} + (y_d - 1)^2\} + \sum_{i=1}^{D}\mu(x_i, 10, 100, 4)$$

$$\text{where} \quad y_i = 1 + \tfrac{1}{4}(x_i + 1) \quad x_i \in [-50, 50] \tag{9}$$

6. *Penalised function P16*

$$f_6 = 0.1\{10\sin^2(3\pi x_1) + \sum_{i=1}^{D-1}(x_i - 1)^2\{1 + 10\sin^2(3\pi x_{i+1})\} +$$

$$(x_d - 1)^2\{1 + \sin^2(2\pi x_D)\}\} + \sum_{i=1}^{D}\mu(x_i, 5, 100, 4) \quad x_i \in [-50, 50] \tag{10}$$

Jiuyong Li \(Ed.\), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010

where

$$\mu(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a \le x_i \le a \\ k(-x_i - a)^m & x_i < a \end{cases}$$

7. *Shifted Rastrigin*

$$f_7 = \sum_{i=1}^{D} \{z_i^2 - 10\cos(2\pi z_i) + 10\} \quad x_i \in [-5.12, 5.12], z_i = x_i - o_i \qquad (11)$$

where o_i is a random value in $[-5.12, 5.12]$.

The global optimal value of Generalized Schwefel 2.6 function is −12569.5, and the global optimal values of all of the other six functions are zero.

4.2 Parameter Setting

In the literature, $D = 30$ and $D = 100$ are usually chosen to test algorithm performance for solving multi-modal, high-dimensional functions, with $D = 30$ being the most popular setting. To evaluate the performance of the new hybrid algorithms and compare with results reported for DM3-PSO [2], DEPSO [8] and DE [8], $D = 30$ is selected. To compare with DM3-PSO [2], the number of function evaluations in HybridPSO1 and HybridPSO2 are limited to 300000. Table 1 shows the parameter values in the HybridPSO1 and HybridPSO2. We run each function test 100 times randomly and independently. These parameter values were chosen based on the literature.

5 Experimental Results and Discussion

We study the results from both hybrid algorithms and compare them with results reported for DM3-PSO [2], DEPSO [8] and DE [8]. Table 2 shows the comparison. The results for both SPSO and DM3-PSO come from [2]. Simply, values less than 10^{-11} have been round to 0.0. For DEPSO and DE we only list the results available from [8], which does not give the number of function evaluations.

Table 1. Parameter settings

Parameter	Value
Population size	60
D (dimension)	30
n_{itr}	4
p_{local}	0.05
F	1.2

Jiuyong Li \(Ed.\), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010

Table 2. Results for solving seven multi-modal functions: mean ± standard deviation of best function values found from 100 replications. Here '±0.0' means that all replications found the global optima.

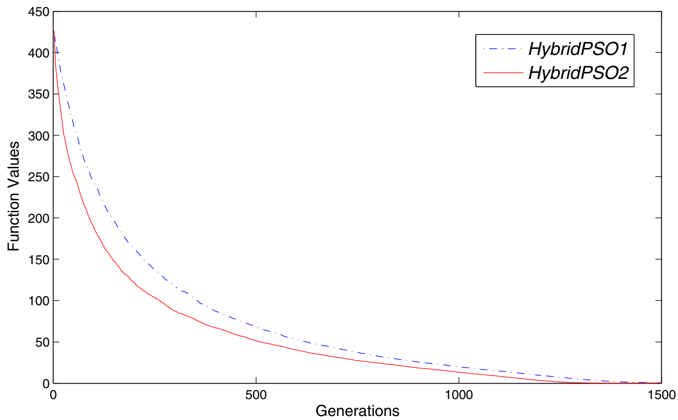
	HybridPSO1	HybridPSO2	SPSO[2]	DM3-PSO[2]	DEPSO[8]	DE[8]
f_1	-12569.5 ± 0.0	-12569.5 ± 0.0	3522 ± 32	1830 ± 46	-12569.5 ± 0.0	-9639.5 ± 190
f_2	0.0 ± 0.0	0.0 ± 0.0	140.156 ± 5.87	9.88 ± 0.86	0.0 ± 0.0	2.2 ± 1.8
f_3	0.0 ± 0.0	0.0 ± 0.0	12.93 ± 1.59	0.0 ± 0.0	0.0 ± 0.0	$(1.1 \pm 0.2)10^{-5}$
f_4	0.0 ± 0.0	0.0 ± 0.0	0.019 ± 0.004	0.0 ± 0.0	—	—
f_5	0.0 ± 0.0	0.0 ± 0.0	0.15 ± 0.05	0.0 ± 0.0	—	—
f_6	0.0 ± 0.0	0.0 ± 0.0	0.003 ± 0.001	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
f_7	1.6 ± 1.5	0.14 ± 0.40	—	—	55.0 ± 4.4	47.0 ± 7.2

From Table 2, it is found that DM3-PSO, HybridPSO1 and HybridPSO2 can successfully converge to the global solutions of functions f_3, f_4, f_5 and f_6, but DM3-PSO has difficulty with solving functions f_1 and f_2. HybridPSO1 and HybridPSO2 solve six of the multi-modal functions successfully, and gives a good solution to f_7. Especially for f_2 (Generalised Rastrigin function), these other PSO variants cannot even find the global solution [11]. In [8], DEPSO found the global optima in 7.7 seconds on a 2.8GHz CPU. In HybridPSO1, it only takes 1 second for 300 000 function evaluations with 2.1GHz CPU. HybridPSO1 finds the global value far less than 300 000 function evaluations (see From Figure 2). DEPSO [8] took 10.8 ± 0.2 seconds for solving function f_3 and HybridPSO1 only took about 1 second for finding the global optima. For finishing 300000 function evaluations, HybridPSO2 takes about 1 second. As a final test, the hybrid PSO methods perform considerably better on f_7 than DEPSO and DE. Based on the analysis and comparison, HybridPSO1 and HybridPSO2 have excellent performance in solving multi-modal and high-dimensional function optimisation problems.

Figure 2 plots the best value against each generation (top figure) and against the count of function evaluations (bottom figure), when using HybridPSO1 and HybridPSO2 to optimise function f_2. In both plots, the vertical axis represents the best value seen so far, averaging over 100 independent trails. Their evolution progress demonstrates HybridPSO2 converges faster to the global position in the early stages based on the population generation. HybridPSO2 has a slightly faster convergence speed to the global position in the early stages based on the number of function evaluations, but its speed is slower than HybridPSO1 after about 2.0×10^4 evaluations.

Since we weaken the global best position effect, the whole population does not appear to get trapped in one local optimal position, and both hybrid PSO algorithms keep good population diversity. The local search appears to help particles explore local optima, therefore HybridPSO2 accelerates the progress of particles evolution in the early stage but at the expense of more function evaluations.

Jiuyong Li (Ed.), AI 2010: Advances in Artificial Intelligence
3rd Australasian Joint Conference Adelaide,Australia,December 2010 Proceedings
Springer-Verlag Berlin Heidelberg 2010



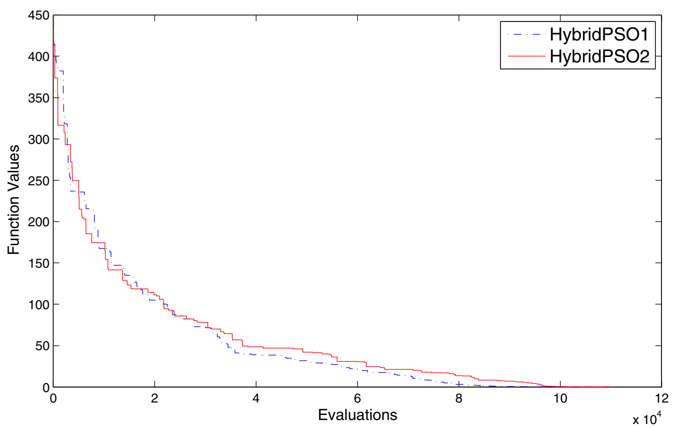


Fig. 2. The best function value for the 30-dimensional Generalised Rastrigin function in each generation (top figure) and each count of function evaluations (bottom figure) using HybridPSO1 and HybridPSO2

6 Conclusions

The goal of this paper was to investigate hybrid PSO approaches to optimise multi-modal functions. The goal was successfully achieved by using a DE operator and integrating a local search. In both hybrid algorithms, the convergence to local optima was successfully avoided, and the HybridPSO2 can converge faster to global solutions than the HybridPSO1 in the early stages.

Two hybrid PSO algorithms were developed in this paper. HybridPSO1 replaces the method in standard PSO with one DE operator and uses it to update particles. HybridPSO2 integrates one local search operator based on HybridPSO1, explores the local optimal position in particles region. Both hybrid PSO algorithms are effective to find the global solutions of the seven benchmark multi-modal and high-dimensional functions. In future work, we will investigate further ways to use different local search operators to help particles fly to the global best position.

References

1. Kennedy, J., Eberhart, R.C.: Particle Swarm Optimization. In: Proceedings of IEEE International Conference on Neural Networks IV, pp. 1942–1948 (1995)
2. Bratton, D., Blackwell, T.: A simplified recombinant PSO. Journal of Artificial Evolution and Applications (2008)
3. Setayesh, M., Zhang, M., Johnston, M.: A new homogeneity-based approach to edge detection using PSO. In: Proceedings of the 24th International Conference on Image and Vision Computing New Zealand (IVCNZ), pp. 231–236. IEEE Press, Wellington (2009)
4. Aziz, N., Mohemmed, A.W., Zhang, M.: Particle swarm optimization for coverage maximization and energy conservation in wireless sensor networks. In: Di Chio, C., Brabazon, A., Di Caro, G.A., Ebner, M., Farooq, M., Fink, A., Grahl, J., Greenfield, G., Machado, P., O'Neill, M., Tarantino, E., Urquhart, N. (eds.) Applications of Evolutionary Computation. LNCS, vol. 6025, pp. 51–60. Springer, Heidelberg (2010)
5. Poli, R., Kennedy, J., Blackwell, T., Freitas, A.: Particle swarms: the second decade. Journal of Artificial Evolution and Applications (2008)
6. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization: an overview. Swarm Intelligence 1(1), 33–57 (2007)
7. Price, K., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization. Springer, Heidelberg (2005)
8. Xin, B., Chen, J., Peng, Z., Pan, F.: An adaptive hybrid optimizer based on particle swarm and differential evolution for global optimization. Science China Information Sciences 53(5), 980–989 (2010)
9. Zhang, W., Xie, X.: DEPSO: hybrid particle swarm with differential evolution operator. In: IEEE International Conference on Systems, Man & Cybernetics (SMCC), Washington DC, USA, pp. 3816–3821 (2003)
10. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: Proceedings of IEEE World Congress on Computational Intelligence, pp. 69–73 (1998)
11. Akbari, R., Ziarati, K.: Combination of particle swarm optimization and stochastic local search for multimodal function optimization. In: IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application (PACAIIA), pp. 388–392 (2008)