A Hybrid PSO-BFGS Strategy for Global Optimization of Multimodal Functions

Shutao Li, *Member, IEEE*, Mingkui Tan, Ivor W. Tsang, and James Tin-Yau Kwok

***Abstract*—Particle swarm optimizer (PSO) is a powerful optimization algorithm that has been applied to a variety of problems. It can, however, suffer from premature convergence and slow convergence rate. Motivated by these two problems, a hybrid global optimization strategy combining PSOs with a modified Broyden-Fletcher-Goldfarb-Shanno (BFGS) method is presented in this paper. The modified BFGS method is integrated into the context of the PSOs to improve the particles' local search ability. In addition, in conjunction with the territory technique, a reposition technique to maintain the diversity of particles is proposed to improve the global search ability of PSOs. One advantage of the hybrid strategy is that it can effectively find multiple local solutions or global solutions to the multimodal functions in a box-constrained space. Based on these local solutions, a reconstruction technique can be adopted to further estimate better solutions. The proposed method is compared with several recently developed optimization algorithms on a set of 20 standard benchmark problems. Experimental results demonstrate that the proposed approach can obtain high-quality solutions on multimodal function optimization problems.**

***Index Terms*—Local diversity, particle swarm optimizer (PSO), reconstruction technique, territory.**

I. Introduction

PARTICLE swarm optimizer (PSO), which was proposed by Kennedy and Eberhart in 1995 [1], is a population-based stochastic optimization technique inspired by the social behavior of bird flocking or fish schooling for finding an optimal solution in complex search spaces. Due to its effectiveness and simple implementation in solving multidimensional problems, PSO and its variants have been applied in many areas.

However, one drawback of the canonical PSO is that it suffers from premature convergence and slow convergence rate [2], [3]. To address this problem, many improvements of the PSO algorithms have been proposed. Traditional improved variants can be generally categorized into three groups [3]. The first category adjusts parameters to trade off the global and local search abilities of PSO [4], [5]. The second category designs efficient population utilization strategy or dynamic multiple swarms to improve the global search ability [6]–[8]. In the third category, a hybrid mechanism combining PSO with other evolutionary algorithms is explored to keep the population diversity and improve the local convergence rate [9]–[11].

Another drawback of the canonical PSO and the traditional variants is that it is difficult for them to find multiple optima due to an intrinsic restriction that all particles must converge to only one point at the final step [12]. To address this problem, a multigrouped particle swarm optimization technique was proposed in [12]. It allows particles to converge to multiple points rather than to only one point, and thus, it can find multiple local optima. However, it has the limitation that each local optimum needs to be supported by an independent swarm [12]. Parsopoulos and Vrahatis [13] introduced a repulsion technique as well as deflection and stretching techniques into PSO to compute all the global optima. This is an efficient algorithm that has the ability to detect all global minimizers of a function, under the assumption that the global optimum was known *a priori*. However, this assumption does not hold for most problems in real problems.

Recently, improving the performance of evolutionary algorithms by introducing the local search method into the evolutionary algorithms has attracted much attention [14]–[16]. Based on the estimation of distribution algorithm, Zhang *et al.* [17] introduced a hybrid evolutionary algorithm for continuous global optimization problems where the simplex method was introduced to implement the local search. To improve the local search ability of genetic algorithm (GA), a large collection of methods, named as memetic algorithm (MA), has been thoroughly studied in recent years [18]–[20]. In particular, in [19], a dynamical approach is proposed to start the local search and determine the local search intensity. However, this strategy may lead to too many local searches. As for PSO, Liang and Suganthan developed a hybrid strategy combining a dynamic multiswarm (DMS) PSO with a local search technique to maintain the particles' diversity as well as local search ability [2].

In addition, Fan and Zahara [21] also proposed to integrate the simplex search method into the PSO iterations for unconstrained optimizations. There are also some other combination strategies [22], [23]. For example, Coelho and Mariani [23] recently developed a novel chaotic PSO combined with an implicit filtering local search method to solve economic dispatch problems.

Manuscript received January 9, 2010; revised August 9, 2010 and November 21, 2010; accepted December 15, 2010. Date of publication January 28, 2011; date of current version July 20, 2011. This work was supported in part by the National Natural Science Foundation of China under Grant 60871096 and Grant 60835004 and in part by the Ministry of Education of China through the Ph.D. Programs Foundation under Grant 200805320006. This paper was recommended by Associate Editor Q. Zhang.

S. Li is with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China (e-mail: shutao_li@yahoo.com.cn).

M. Tan was with the College of Electrical and Information Engineering, Hunan University, Changsha 410082, China. He is now with the School of Computer Engineering, Nanyang Technological University, Singapore 639798 (e-mail: tanmingkui@gmail.com).

I. W. Tsang is with the School of Computer Engineering, Nanyang Technological University, Singapore 639798.

J. T.-Y. Kwok is with the Department of Computer Science and Engineering, Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCB.2010.2103055

The above methods have shown great improvements to the local convergence of the population-based methods. However, the backscattering mechanism and the potentials of the hybrid strategy need to be investigated further. One important issue is how to prevent particles from being trapped in a local optimum in the local search. In this paper, an innovative framework is proposed to integrate the deterministic optimization methods into PSO algorithms. The main objectives of this work are to alleviate the premature convergence of PSO and improve its convergence rate.

In conclusion, the main contributions of this paper are listed as follows: 1) in the proposed method, rather than periodically invoked, the local search is dynamically started by using a proposed local diversity index (LDI); 2) a reposition technique in conjunction with a territory technique is proposed to maintain the diversity of the particles, which can efficiently improve the global search ability and prevent particles from being trapped in a local optima; and 3) a reconstruction operator is conducted to learn the global optimum or better local solutions from the obtained multiple local optima. In addition, our method is helpful for finding the multiple solutions to the multimodal functions in a more efficient way.

The rest of this paper is organized as follows. In Section II, the hybrid PSO-BFGS strategy as well as the related techniques are presented. Experiments on the benchmark functions and discussions are illustrated in Section III. The conclusions of this paper are finally discussed in Section IV.

II. PSO-BFGS Strategy

A. Canonical Particle Swarm Optimization

In the canonical PSO algorithm, each individual can be seen as a particle in a D-dimensional space. The PSO exploits potential solutions through a population and detects the optimal solution based on the cooperation and competition among particles.

The evolution mechanism of a single particle in the canonical PSO can be described as follows:

$$V_{id} = w \times V_{id} + c_1 \times r_1 \times (pbest_{id} - X_{id}) + c_2 \times r_2 \times (gbest_d - X_{id}) \quad (1)$$

$$X_{id} = X_{id} + V_{id} \quad (2)$$

where V_{id} and X_{id} represent the velocity and position of particle i in the dth dimension, respectively, w is the inertial weight that makes a tradeoff between the global and local search abilities [4], c_1 and c_2 are acceleration constants, r_1 and r_2 are random numbers in the range [0, 1], $pbest_{id}$ is the best position found so far regarding to particle i in the dth dimension, and $gbest_d$ is the globally best position that has been visited so far by all the particles.

B. Premature Convergence and Population Diversity

The main deficiencies of the canonical PSOs are the premature convergence and the slow convergence rate. Therefore, why is the performance of PSO limited? Generally speaking, we can divide the multimodal function optimization into two substages: the first stage is to find the optimality basin, and the second stage is to reach the local or global optimum [24]. Here, the optimality basin means a small neighborhood around a local minimum x^*, from any point x of which one can reach to x^* smoothly and monotonically [19]. In the traditional PSOs, both stages are fulfilled by the cooperation and competition of particles, which unavoidably weakens the global search ability of the particles at the final iterations [12]. Therefore, to maintain high diversity is important for PSOs to avoid premature convergence. Multiswarming is one possible way to maintain large diversity [2]. However, it may decrease the local convergence rate.

The diversity of the population can be a good measure to the global search ability. Then how do we measure the population diversity? In this paper, we propose the use of LDI to measure the local as well as global diversity of the population. Here, we use three nearest particles to represent the local neighborhood structure of the population. Let X_0^k be the particle with the best fitness value and X_{01}^k and X_{02}^k be the two particles closest to X_0^k, where k is the iteration index. Then, LDI at iteration k is defined as

$$LDI^k = \frac{\sum_{i=1}^{2} \left\| X_0^k - X_{0i}^k \right\|}{2\sqrt{\sum_{j=1}^{D} (Ub_j - Lb_j)^2}} \quad (3)$$

where Ub_j and Lb_j are the upper and lower bounds for the dimension j of the search space, respectively, and D is the dimensionality of the problem. For simplicity, we hereafter drop the superscript k for LDI^k. There are several considerations to use LDI. At first, the local neighborhood is better to describe the structure of particles in the local basins. Second, this definition is also suitable for multiswarm systems where the leading swarm may only contain a small number of particles.

Obviously, LDI can also present the global diversity of the population. For a given swarm system, the larger the LDI is, the less likely will the population get stuck in premature convergence. The faster the LDI value decreases, the faster PSO converges and the more likely it is to be trapped in premature convergence. Hence, we can roughly determine whether the particles enter an optimality basin or not by using LDI. if LDI is small enough (e.g., smaller than a predefined LDI_0), we can assume that the particles have entered an optimality basin. In conclusion, we can divide the particles' search behavior into the global search and the local search by using LDI. That is, if $LDI > LDI_0$, then the population is doing the global search. Otherwise, the population will perform the local search. Here, LDI_0 can be also considered as a *coarse stopping criterion* on the PSO algorithms and can be directly adopted as the termination criterion for the traditional PSOs.

C. General Ideas

As below, we will start to present our new hybrid scheme that integrates the local search into PSO iterations for multimodal function optimization. In our method, we use a modified BFGS method as the local search technique. Several critical issues of such integration remain to be addressed. The first is when to start the local search and how to efficiently use the local search. The second is how to find and hold multiple local optima and prevent intruding a local optimum (or local optimality basin) in the local search. The third is how to efficiently keep the diversity of the population. The last issue is how to reuse the obtained multiple local optima to estimate the global or better solutions, if possible.

We address the first problem by means of LDI, as shown in Section II-B. To approach the latter problems, several operators are proposed: a *territory* technique in Section II-E to hold the multiple local optima while *reposition* operator in Section II-F to keep the diversity of particles. Meanwhile, a *reconstruction* algorithm is adopted in Section II-G to reconstruct solutions. Finally, the general scheme that integrates these terms will be presented in Section II-H.

D. Local Search With a Modified BFGS Method

In the proposed strategy, the local search of PSO is implemented by a modified BFGS method. BFGS is an effective quasi-Newton method in solving unconstrained nonlinear optimization problems. In the BFGS method, only the first derivative needs to be calculated. However, there is no guarantee that it can converge on nonconvex or ill-conditioned problems. Hence, some modifications should be made. Let $\nabla f(x)$ be the gradient or subgradient of a function $f(x)$ at point x and d^k be the search direction at iteration k.

1) Given an optimization problem with constraint set Ω, a minimizer may lie either in the interior or on the boundary. Hence, besides $\|\nabla f(x^k)\| < \epsilon$, two other stopping criteria, i.e., $\|\nabla f(x^k)\|/\|\nabla f(x^0)\| < \epsilon$ and $|f(x^{k+1}) - f(x^k)| < \epsilon$, are adopted when solving nonsmooth or nonconvex problems, where $\nabla f(x^0)$ is the gradient of the initial point x^0. These two conditions are very important when the point is on the constraint bounds or BFGS cannot converge. For those points without the definition of gradient, we can simply treat these points as the local optima. Note that in the real-world applications, some problems may not be differentiable. In these cases, we can use the ***numerical gradients*** instead [25]. The feasibility as well as the convergence property of the BFGS method using numerical gradient was discussed in [25].
2) The magnitude of the search direction d^k can be very large in the early iterations, and this may move the particle far beyond the search space. Then, a projection strategy is adopted to ensure that the particles always stay inside the bound. That is, if x^{k+1} is outside the search space, it will be projected back by

$$x^{k+1} = P_\Omega(x^{k+1}) \tag{4}$$

where P_Ω is a projection operator on Ω defined as follows [26]:

$$P_\Omega(x, L_b, U_b)_i = \begin{cases} L_{bi} & x_i < L_{bi} \\ x_i & L_{bi} \le x_i \le U_{bi} \\ U_{bi} & x_i > U_{bi}. \end{cases}$$

E. Territory of Particles

With the local search, we can easily find multiple solutions. To hold those solutions and avoid intruding in the same basin, the term *territory* is used. In animal behaviors, territory is a fixed area from which an animal or a group of animals exclude other members of the same species. In light of this function, the territory can be naturally introduced into PSOs to prevent the particles being trapped in a local optimality basin.

In this paper, a territory is represented as a hyperball consisting of the following three parts: 1) the local solution L; 2) the radius of the territory R; and 3) the local optimal value $f(L)$. It is presented as $O(L, R, f(L))$. If a particle finds a local solution L, it will exclude others from intruding. If a new local optimum is found, a new territory is added to territory set T (which is initially empty).

For a given local optimum L, the radius R of a territory can be approximated by $R_s = \|x_s - L\|_2$. However, $\|x_s - L\|_2$ can be too large for some cases, and it may overlay some potential solutions. Then, we should constrain the radius using an upper bound $R_{\max}$. In our method, we use LDI to determine the particles' status. Then, $R_{\max}$ should be smaller than the sum of the distance of X_0, X_{01} and X_{02}. Hence, we can approximate $R_{\max}$ by $R_{\max} \approx LDI_0\sqrt{\sum_{n=1}^{D}(Ub_n - Lb_n)^2}$, where D is the variable dimensions. Obviously, different search scopes may result in different $R_{\max}$ values. To avoid this, we use the following alternative metric:

$$R_{\max} \approx LDI_0\sqrt{D}. \tag{5}$$

Finally, we confirm the radius R by $\min\{R_{\max}, R_s\}$. On the other hand, if another particle of local search is trapped in an existing territory, we can update the radius dynamically if possible. That is, once we obtain a new R_{new}, we update R with R_{new} if $R_{\text{new}} > R$. In such a way, we can quit the local search in advance for saving computations. Note that if a particle is trapped in multiple territories, it is necessary to confirm which territory the particle is trapped in. This can be easily performed by

$$j = \arg\max_i (\cos(\beta_i)) \tag{6}$$

where $\cos(\beta_i)$ is the cosine of the angle β_i formed by the search direction of BFGS and the direction of the particle to each territory. The territory mentioned above can be seen as an approximation to the local optimality basin. However, the real local optimality basin may be much more complex with complex shapes, while the territory is defined as a hyper ball for simplicity.

F. Reposition

We use a *Reposition* operator to dispatch particles, which can efficiently maintain the diversity of the population. Once a territory is found, there is no need to do the local search within this territory. The particle in a local search as well as its two neighboring particles can then be repulsed to explore other solutions. To explore larger space, we also repulse the $p_r \times ps$ particles with better fitness values. Here, ps denotes the number of particles, and p_r denotes the portion of particles that should be repulsed. On the other hand, if the PSOs cannot converge, we reduce the search scope of the $p_r \times ps$ particles with inferior fitness values and drag them to the $pbest$ of someone else. The repulsed or dragged particles are called freed particles. The

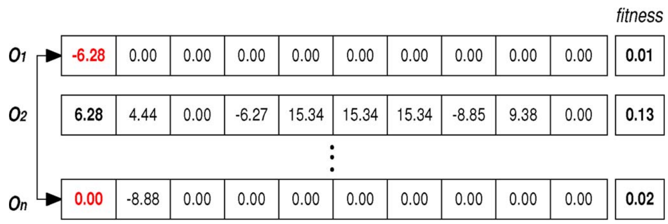


Fig. 1. Local territories of particles on Griewanks's function [3]. $Fitness$ denotes the function value of different territories, and $[0, 0, \ldots, 0]^{10}$ is the global optimum.

proposed reposition technique keeps the freed particles with unchanged velocities and updates their positions as follows:

$$X_{ik} = pbest_{\bullet k} + \lambda_k \times (Ub_k - Lb_k) \times N(0, 1) \tag{7}$$

where $N(0, 1)$ is the normal distribution with mean 0 and standard deviation 1, k is the dimension index, $pbest_{\bullet k}$ is the randomly selected particle's $pbest$, and λ_k is a scalar that confines the distribution of the new particle in the kth dimension. λ_k decreases linearly with the standard variation δ_k of the swarm as

$$\lambda_k = \lambda_{k\,\max} - \frac{\lambda_{k\,\max} - \lambda_{k\,\min}}{\sigma_{k\,\max} - \sigma_{k\,\min}}(\sigma_k - \sigma_{k\,\min}) \tag{8}$$

where $\sigma_{k\,\max} = \sqrt{(Ub_k - Lb_k)^2/12}$ is the standard variation of the uniform distribution $U(Lb_k, Ub_k)$, $\lambda_{k\,\max} = (\sigma_{k\,\max}/Ub_k - Lb_k)$ is the maximum value of λ_k, $\sigma_{k\,\min}$ is the minimum standard variation that the population in the kth dimension can achieve, and $\lambda_{k\,\min}$ is the minimum value of λ_k. Since the local search will be started when $LDI < LDI_0$, we set $\lambda_{k\,\min} = \sigma_{k\,\min} = LDI_0$. According to (8), if the population in the kth dimension has a large diversity, λ_k will be small, and particle i will be dragged to $pbest_{\bullet k}$. Conversely, when λ_k is large, then particle i will be repulsed. Therefore, the reposition technique can efficiently keep the population's diversity and simultaneously reduce the search scopes of the particles when the PSO cannot converge. Hence, the premature convergence problem is avoided.

G. Reconstruction

When multiple local optima are obtained, can we further estimate the global optimum or a relatively better solution from these optima? The answer is possible, which we will show by the *Reconstruction* technique.

Note that two local solutions in a territory set usually have differences only in some locations. Then, a better solution can be estimated by exchanging the different locations using a cooperative learning strategy [27]. For example, in Fig. 1, there are n territories (local solutions) obtained on Griewanks's function (no shift and no rotation) with ten dimensions. None of them is the global solution, and the global optimum is $[0, 0, \ldots, 0]^D$. However, they contain some information about the global optimum. And we can easily estimate the global optimum based on these local optima. For example, if we choose O_1 as the context vector and O_n as the learning vector. Then, the global solution can be obtained by exchanging the *different* components of O_1 with O_n at the first dimension.

In the above example, only one exchange step is required to get the global optimum. However, we can also change multiple positions, i.e., learning steps, in one time. More generally, to handle the nonseparable function, rather than exchanging fixed learning steps in the original cooperative learning [27], we can use varied learning steps, referred to as the variable-step-length cooperative learning (VSLCL) strategy. Given two local solutions, suppose that the one $O_c = [L_c, R_c, f(L_c)]$ with smaller fitness values may contain more information about the global optimum, we choose it as the context vector, and the other one, $O_l = [L_l, R_l, f(L_l)]$, is called the learning vector. In VSLCL, let m_l be the maximum number of locations that can be exchanged each time and l_s be the learning step in some iteration. The VSLCL between the two vectors is performed as follows:

Algorithm 1. VSLCL algorithm

0) Given two local solutions L_c and L_l.

1) Find the locations with different values under some precision ε, counting the number as d_n. If $d_n < m_l$, set $m_l = d_n$. Initialize $l_s = m_l$.

2) Replace l_s locations in L_c that are different in L_l with the counterparts in L_l by order, resulting in a new vector L_{new} with fitness $f(L_{\mathrm{new}})$. Let $L_c = L_{\mathrm{new}}$ if $f(L_{\mathrm{new}}) < f(L_c)$.

3) Let $l_s = ls - 1$. If $l_s > 0$, go to step 2; otherwise, output the new $O_c = [L_c, R_c, f(L_c)]$.

The learning strategy in VSLCL is very similar to the guided mutation used in discrete GAs [28]. The difference lies in that in guided mutation, the swap is performed with some probabilities, while in VSLCL, the swap is performed when there is an improvement for the fitness value. When there are more than two local solutions, the VSLCL can be easily extended to the multiple local optima case. Let N_T be the number of territories and all better reconstructed solutions are stored in a new territory set T_{new}. Then, the reconstruction algorithm iteratively proceeds as follows:

Algorithm 2. Reconstruction algorithm

0) Given a territory set T and new territory set $T_{\mathrm{new}} = [\ ]$. Let n_t be the size of T.

1) Find the territory from T with the minimum fitness value as a context vector O_c. Select another territory in turn as the learning territory O_l. Perform VSLCL between O_l and O_c, and obtain a new territory O_{newc}.

2) If $O_{\mathrm{newc}} = O_c$, it indicates that O_c is not changed, and that there is no need to continue to update this O_c. Add O_c to T_{new} and delete O_c from T. Let $n_t = n_t - 1$.

3) If $n_t < 2$ and O_c to T_{new} and go to step 4); otherwise, go to step 1).

4) Choose the solution with the best fitness value from T_{new} as the estimated global optimum.

H. General Framework of the Hybrid Strategy

With all problems solved, we now present the implementation scheme of the proposed hybrid strategy, as shown in Fig. 2. Most PSO algorithms can be adopted to implement the hybrid strategy. To better illustrate the hybrid strategy, a particle flag $pflag$ is introduced to denote the state of the particles. Based on our hybrid strategy, there are three possible states, denoted by 0, 1, and 2, for each particle. State 0 denotes that the particle

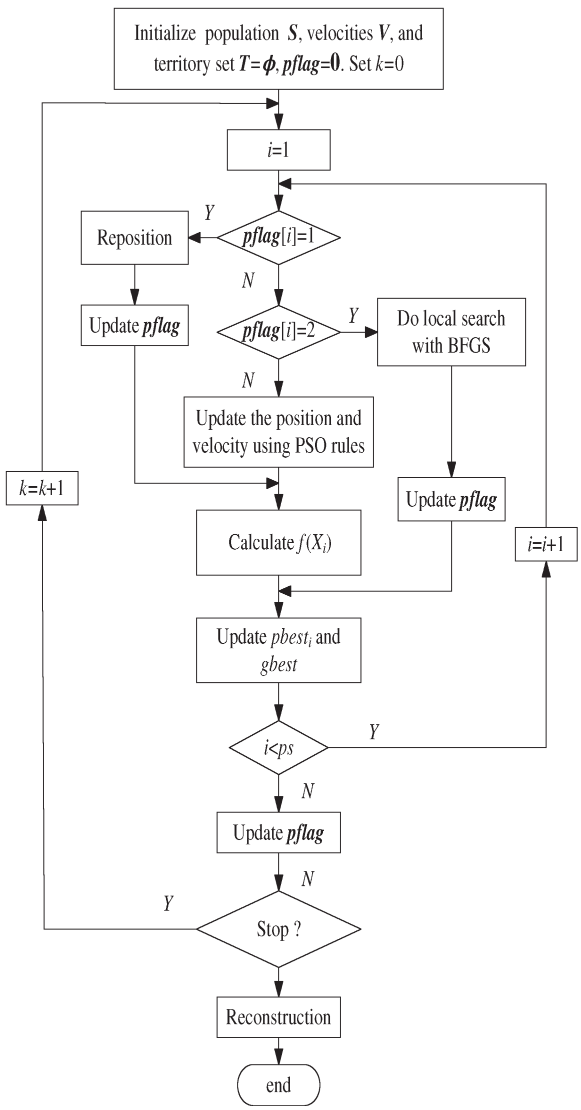


Fig. 2. General framework of the PSO-BFGS strategy.

is normal, and its position and velocity are updated according to PSO rules. State 1 denotes that the particle is free, and it should be updated using the reposition technique. State 2 denotes the particle is in an optimality basin and the local search should be invoked. A condition transition diagram to describe the particle status is shown in Fig. 3. During initialization, the flag for each particle is set to 0. The state updating rules are summarized as follows.

Condition I: If the $LDI < LDI_0$ holds, then the flag of the best particle is set to 2, and the flags of its two nearest particles as well as those $p_r \times ps$ particles with the best fitness values are set to 1.

Condition II: If the above condition is not satisfied, the local search is enforced in every K iterations. In other words, if $k = mK$, where m is an integer, the flag of the best particle is set to 2, and those $p_r \times ps$ particles with the lowest fitness values are set to 1.

Condition III: If the local search for a particle i is done, set its flag to 1.

Condition IV: If the reposition process of the particle i is done, then its flag is set to 0.

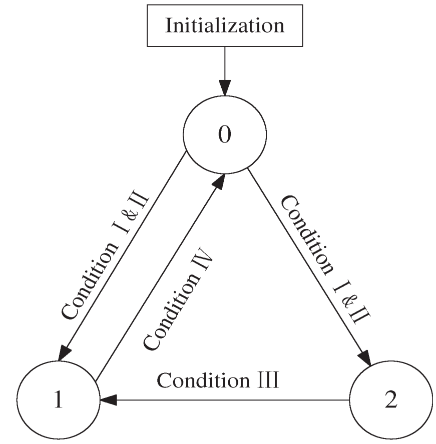


Fig. 3. Condition transition diagram of particles in PSO-BFGS algorithm. The state variable $pflag$ in Fig. 2 switches among status 0, 1, and 2 in the update step.

TABLE I
GLOBAL LOCAL OPTIMA, SEARCH RANGES, AND INITIALIZATION RANGES OF THE TEST FUNCTIONS

f	$f(x^*)$	Search Range	Initialization Range	Function Name
f_1	0	$[-2.048, 2.048]^D$	$[-100, 50]^D$	Shifted Rosenbrock Function [3]
f_2	0	$[-32.768, 32.768]^D$	$[-32.768, 16]^D$	Shifted Ackley Function [3]
f_3	0	$[-600, 600]^D$	$[-600, 200]^D$	Shifted Griewanks Function [3]
f_4	0	$[-0.5, 0.5]^D$	$[-0.5, 0.2]^D$	Shifted Weierstrass Function [3]
f_5	0	$[-5.12, 5.12]^D$	$[-5.12, 2]^D$	Shifted Rastrigin Function [3]
f_6	0	$[-500, 500]^D$	$[-500, 500]^D$	Schwefel Function [3]
f_7	0	$[-10, 10]^D$	$[-10, 5]^D$	Shifted Levy Function [30]
f_8	0	$[-50, 50]^D$	$[-50, 20]^D$	Shifted Penalized Function P8 [30]
f_9	0	$[-2.048, 2.048]^D$	$[-100, 50]^D$	Shifted Rotated Rosenbrock Function [3]
f_{10}	0	$[-32.768, 32.768]^D$	$[-32.768, 16]^D$	Shifted Rotated Ackley Function [3]
f_{11}	0	$[-600, 600]^D$	$[-600, 200]^D$	Rotated Griewank Function [3]
f_{12}	0	$[-0.5, 0.5]^D$	$[-0.5, 0.2]^D$	Shifted Rotated Weierstrass Function [3]
f_{13}	0	$[-5.12, 5.12]^D$	$[-5.12, 2]^D$	Shifted Rotated Rastrigin Function [3]
f_{14}	0	$[-500, 500]^D$	$[-500, 500]^D$	Rotated Schwefel Function [3]
f_{15}	0	$[-10, 10]^D$	$[-10, 5]^D$	Shifted Rotated Levy Function [30]
f_{16}	0	$[-50, 50]^D$	$[-50, 20]^D$	Shifted Rotated Penalized Function P8 [30]
f_{17}	0	$[-100, 100]^D$	$[-100, 100]^D$	Shifted Rotated Expanded Scaffer F6 [29]
f_{18}	0	$[-500, 500]^D$	$[-100, 100]^D$	Schwefel's Problem 2.6 [29]
f_{19}	0	$[-\pi, \pi]^D$	$[-\pi, \pi]^D$	Schwefel's Problem 2.13 [29]
f_{20}	0	$[-3, 1]^D$	$[-3, 1]^D$	Expanded Extended F8F2 [29]

By means of the $pflag$ and the LDI, the local search and the global search can be performed separately. Accordingly, part of the particles can be freed to go on the global search and maintain a relatively high diversity. Therefore, the premature convergence can be avoided. Finally, if multilocal optima are obtained, we can optionally reconstruct or estimate the global optimum or a better solution based on the VSLCL method. In the evolutionary algorithms, the maximum number of iterations $\max\_iter$ and the maximum number of fitness evaluations are commonly used as termination conditions. For the proposed strategy, in addition to these two conditions, the number of territories can be also adopted as a stopping criterion. This criterion is very useful when dealing with multiple global optimization problems.

III. BENCHMARK TESTS AND DISCUSSIONS

A. Benchmark Functions

Twenty multimodal benchmark functions are chosen to evaluate the proposed strategy. These functions, except for f_6, f_{14}, and $f_{17} - f_{20}$, are the shifted or shifted rotated versions of several basic multimodal functions using the rules discussed in [29]. Note for the last four functions, we omit their f_{bias} in [29]. Table I shows the global optimal fitness value $f(x^*)$, the search ranges $[Lb, Ub]^D$, and the initialization range of each function.

TABLE II
AVERAGE RATIO OF TIME SPENT ON GRAD EVALUATIONS TO FUNCTION EVALUATIONS. tg/tf STANDS FOR THE AVERAGE RATIO OF 2000 EXPERIMENTS, AND Tg/Tf STANDS FOR THE RATIO ADOPTED IN THE EXPERIMENTS

Dim	f_1		f_2		f_3		f_4	
	tg/tf	Tg/Tf	tg/tf	Tg/Tf	tg/tf	Tg/Tf	tg/tf	Tg/Tf
10	2.88	3	1.29	1.5	1.09	1.5	1.58	2
30	3.64	4	1.47	1.5	1.01	1.5	1.47	1.5
Dim	f_5		f_6		f_7		f_8	
	tg/tf	Tg/Tf	tg/tf	Tg/Tf	tg/tf	Tg/Tf	tg/tf	Tg/Tf
10	2.88	3	1.29	1.5	1.09	1.5	1.58	2
30	3.64	4	1.47	1.5	1.01	1.5	1.47	1.5

B. Experimental Settings

In our experiments, we use both the numerical gradient and the analytical gradient of the test functions to form the search direction in the BFGS method. The PSO with inertia weight (PSO-w) [4] and the comprehensive learning particle swarm optimizer (CLPSO) [3] are chosen as the two context algorithms. With analytical gradients, it results in two new algorithms, PSO-w-BFGS and CLPSO-BFGS. Meanwhile, we use the notation of PSO-w-NBFGS and CLPSO-NBFGS for numerical gradients, where NBFGS denotes BFGS method with numerical gradients. For full comparison, we also use an adaptive simulated annealing (ASA) method [31] as the local search. Here, we use the DMS and the CLPSO as the context PSOs, resulting in two new methods, namely, DMS-L-ASA and CLPSO-ASA. They are compared with other six algorithms, i.e., GA, MA, PSO-w, CLPSO, DMS-L-PSO [2], and a random started BFGS method (Rand-BFGS), on the 20 test functions with 10 and 30 dimensions, respectively. As for Rand-BFGS, we iteratively initialize BFGS with random starting points and keep track of the best solution found over all the runs. We use the Genetic Algorithms for Optimization Toolbox (GAOT) to implement the GA method [32], and the code of MA is from the authors [19].

In our experiments, when counting the number of fitness evaluations, the time spent on the gradient calculation should be considered. For numerical gradient calculation, we use the two-point estimation [27]. Hence, one gradient calculation needs D fitness evaluations. For analytical gradient, Table II lists the average ratio of time spent on derivative evaluations to the function evaluations (denoted by tg/tf) for unrotated problems by averaging 2000 independent experiments with 10 and 30 dimensions. The time ratio adopted in the experiments is denoted by Tg/Tf. When fixing Tg/Tf, we let it always be greater than tg/tf, as shown in Table II. Further, we let the ratios of the rotated problems be the same as their unrotated counterparts. For $f(17) - f(20)$, only numerical gradients are considered.

All the experiments are performed 50 times. The mean and variance of the final function value, and the successful times of finding the global optimum on different problems (referred as hit rate), are used to compare the various algorithms. The hit rate shows the whole performance of the algorithms, while the median convergence elucidates their convergence behaviors. The median function value is obtained as follows: If the hit rate of an algorithm for a particular function is zero, then the median function values of 50 times are recorded; otherwise, only the success cases of finding the global optima are recorded. In addition, a t-test is performed between the best results of our methods, and the best results of others to determine whether the results obtained by the proposed method are statistically different from others. The values 1 and -1 denote that the results obtained by the proposed method are statistically better and worse than the best among the rest of the methods with a 5% significance level, respectively, whereas the value 0 denotes that the results are not statistically different. To make a fair comparison, the maximum number of fitness evaluations is set to 30 000 for the 10-D problems and 180 000 for the 30-D problems. For our method, $0.05 \times \max\_func$ fitness evaluations are left for the reconstruction process. Parameter m_l is the maximum learning steps in the reconstruction process, and 5 is usually large enough. Parameter p_r is the portion of freed particles, which is similar to the mutation probability in GAs [19]. In general, if p_r is too large, the swarm may lose the history search information but can obtain better global search ability. On the other hand, if p_r is too small, the swarm tends to be trapped in the same optimality basin. LDI_0 is the threshold value of the local diversity that can adaptively start the local search in the proposed method. Generally speaking, a small LDI_0 can be set to ensure enough evolutions, and a large LDI_0 can be set to obtain multiple local optimal solutions. The hybrid method will implement the local search periodically at every K iterations when the context PSOs cannot converge where the condition $LDI < LDI_0$ cannot achieve. A large K is favorable. The sensitivity study of the parameters will be further studied in the third experiment. The final parameter settings of the proposed method in the experiments are shown in Table III. The parameters of PSO-w and CLPSO are kept the same as in [3]. Except for DMS-L-PSO, the swarm size or population size is set to 10 for 10-D functions and 30 for 30-D functions for all methods. The same parameter settings of DMS-L-PSO are used as in [2], where the swarms' number is 20 and each swarm's population size is 3. Hence, the total population for DMS-L-PSO size is 60. Except for population size, we also keep the default parameter settings for GA and MA as they are in the toolbox. In MA, the local search is also implemented by the BFGS method.

TABLE III
PARAMETER SETTINGS OF THE PSO-BFGS ALGORITHMS. m_l, LDI_0, p_r, AND K STAND FOR THE MAXIMUM LEARNING STEPS IN RECONSTRUCTION, THE THRESHOLD VALUE OF THE LOCAL DIVERSITY, THE PORTION OF FREED PARTICLES, AND THE SEARCH PERIOD, RESPECTIVELY

	m_l	LDI_0	p_r	K
PSO-w-BFGS	5	0.01	0.25	$30D$
CLPSO-BFGS	5	0.01	0.25	$30D$

C. Experimental Results and Discussions

1) Results of 10-D Problems: In this experiment, all the algorithms are performed on the 20 test functions with ten dimensions. The hit rate (denoted by hit), the mean and variance of the final function values of various algorithms (denoted by $mean \pm variance$), and the t-test results are recorded in Table IV. The number in brackets in the table for PSO-w-(N)BFGS and CLPSO-(N)BFGS represents the number of global optima obtained by the reconstruction technique. Fig. 4 shows the median convergence graphs of the different algorithms, where we do not include the results of DMS-L-ASA and CLPSO-ASA for the former 16 functions to avoid crowding

TABLE IV
RESULTS OF 20 BENCHMARK FUNCTIONS ON TEN DIMENSIONS. *hit* STANDS FOR THE SUCCESSFUL TIMES OF FINDING THE GLOBAL OPTIMUM, WHILE $mean \pm variance$ STANDS FOR THE MEAN AND VARIANCE OF THE FINAL FUNCTION VALUE, RESPECTIVELY

	f_1		f_2		f_3		f_4	
	hit	*mean* ± *variance*	*hit*	*mean* ± *variance*	*hit*	*mean* ± *variance*	*hit*	*mean* ± *variance*
GA	0	5.96e+00±1.08e+00	0	9.00e-02±4.00e-02	0	1.30e-01±5.00e-02	0	2.00e-02±2.00e-02
MA	1	3.32e+00±2.52e+00	50	5.88e-016±0.00e+00	14	1.62e-01±2.07e-01	0	6.36e-01±6.20e-01
PSO-w	0	2.99e+00±1.57e+00	50	1.00e-14±7.00e-15	0	9.44e-02±4.43e-02	25	9.51e-01±1.32e+00
CLPSO	0	3.09e+00±1.99e+00	50	5.00e-15±2.00e-15	27	5.49e-03±7.11e-03	49	3.00e-02±2.12e-01
Rand-BFGS	50	1.18e-11±1.61e-11	0	1.77e+01±1.06e+00	0	1.68e+00±1.48e+00	0	8.25e+00±1.03e+00
DMS-L-PSO	50	9.43e-11±2.46e-11	50	1.22e-08±1.20e-08	2	2.05e-02±1.45e-02	50	8.67e-05±3.40e-05
PSO-w-BFGS	50	4.78e-13±1.31e-13	50	1.04e-14±4.02e-15	44(18)	1.53e-03±5.65e-03	29(9)	8.46e-01±1.14e+00
CLPSO-BFGS	50	4.30e-14±1.34e-13	50	8.99e-15±3.52e-15	50	0.00+00±5.40e-14	50(15)	0.00e+00±1.00e-15
PSO-w-NBFGS	50	1.46e-11±5.78e-13	50	1.06e-10±2.41e-10	29(8)	4.50e-03±6.40e-03	28(7)	1.20+00±1.39e+00
CLPSO-NBFGS	50	1.04e-14±1.61e-14	50	1.84e-13±1.73e-13	50	1.64e-12±1.26e-12	50(17)	6.47e-15±4.46e-15
DMS-L-ASA	0	5.74e+00±8.20e-01	50	0.00e+00±0.00e+00	0	5.00e-02±2.00e-02	50	0.00e+00±0.00e+00
CLPSO-ASA	0	4.23e+00±2.08e+00	50	0.00e+00±0.00e+00	0	2.00e-02±1.00e-02	0	2.00e-02±1.40e-01
t-test value	1		0		1		1	
	f_5		f_6		f_7		f_8	
	hit	*mean* ± *variance*	*hit*	*mean* ± *variance*	*hit*	*mean* ± *variance*	*hit*	*mean* ± *variance*
GA	0	1.60e-01±3.30e-01	0	1.30e-01±1.20e-01	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
MA	50	0.00e+00±0.00e+00	1	1.63e+04±3.02e+03	-	-	-	-
PSO-w	1	4.50e+00±2.77e+00	0	3.77e+02±1.56e+02	50	0.00e+00±0.00e+00	46	8.79e-04±3.01e-03
CLPSO	44	1.39e-01±4.03e-01	25	7.82e+01±8.83e+01	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
Rand-BFGS	10	3.08e+01±1.97e+01	0	1.10e+03±2.08e+02	50	7.01e-12±7.54e-12	0	8.48e+00±3.98e+00
DMS-L-PSO	24	6.72e-01±8.16e-01	9	1.74e+02±1.31e+02	50	0.00e+00±0.00e+00	50	0.00e+00±1.00e-15
PSO-w-BFGS	42	8.34e-01±2.20e+00	0	5.97e+02±2.09e+02	50	6.11e-13±4.69e-13	50	6.97e-13±5.54e-13
CLPSO-BFGS	50	2.61e-13±2.42e-13	50	4.18e-13±4.58e-13	50	6.52e-13±5.72e-13	50	7.60e-13±5.99e-13
PSO-w-NBFGS	40(3)	7.45e-01±1.84e+00	0	3.76e+02±2.45e+02	50	1.36e-10±1.39e-10	50	2.79e-10±5.83e-10
CLPSO-NBFGS	50	3.55e-09±3.17e-09	50	3.22e-05±2.78e-05	50	6.33e-13±5.88e-13	50	8.70e-13±2.13e-12
DMS-L-ASA	0	1.81e+00±9.90e-01	0	7.19e+00±2.70e+01	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
CLPSO-ASA	0	5.40e-01±8.10e-01	0	2.84e+01±5.11e+01	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
t-test value	1		1		-1		-1	
	f_9		f_{10}		f_{11}		f_{12}	
	hit	*mean* ± *variance*	*hit*	*mean* ± *variance*	*hit*	*mean* ± *variance*	*hit*	*mean* ± *variance*
GA	0	8.13e+00±1.15e+00	0	9.10e-01±7.80e-01	0	2.80e-01±1.500e-01	0	2.12e+00±8.50e-01
MA	0	7.20e+00±4.8e-01	46	3.72e+00±7.45e+00	7	1.27e-02±1.227e-02	0	2.66e+00±1.55e+00
PSO-w	0	4.59e+00±1.33e+00	38	3.24e-01±6.00e-01	0	1.55e-01±8.05e-02	1	1.51e+00±1.36e+00
CLPSO	0	5.20e+00±1.78e+00	50	3.24e-07±2.09e-06	0	4.47e-02±2.44e-02	0	3.59e-01±4.20e-01
Rand-BFGS	50	1.20e-11±1.76e-11	0	1.80e+01±8.35e-01	0	1.63e+00±1.71e+00	0	8.36e+00±8.83e-01
DMS-L-PSO	50	1.60e-10±8.39e-11	50	2.26e-08±1.49e-08	1	2.25e-02±1.51e-02	0	2.01e-02±7.97e-02
PSO-w-BFGS	50	1.09e-13±3.82e-13	50	9.70e-15±4.32e-15	32	4.29e-03±6.21e-03	26(12)	7.40e-01±1.15e+00
CLPSO-BFGS	50	4.20e-14±1.68e-13	50	1.04e-14±4.06e-15	50	6.31e-13±5.81e-13	34(8)	2.43e-01±4.98e-01
PSO-w-NBFGS	50	3.42e-11±1.09e-10	50	6.53e-11±1.43e-10	26	5.90e-03±7.60e-03	17(7)	1.28e-00±1.41e+00
CLPSO-NBFGS	50	2.07e-14±9.43e-14	50	1.88e-12±1.33e-13	50	4.43e-08±4.54e-08	28(10)	2.45e-01±5.37e-01
DMS-L-ASA	0	6.09e+00±9.107e-01	50	0.00e+00±0.00e+00	0	1.00e-01±4.00e-02	0	5.00e-02±1.60e-01
CLPSO-ASA	0	5.18e+00±1.88e+00	50	0.00e+00±0.00e+00	0	6.00e-02±4.00e-02	0	3.60e-01±5.40e-01
t-test value	1		0	1	1		-1	
	f_{13}		f_{14}		f_{15}		f_{16}	
	hit	*mean* ± *variance*	*hit*	*mean* ± *variance*	*hit*	*mean* ± *variance*	*hit*	*mean* ± *variance*
GA	0	1.71e+01±8.42e+00	0	5.03e+02±8.68e+01	50	0.00e+00±0.00e+00	0	1.00e-02±1.00e-02
MA	0	1.2e+01±6.14e+01	0	-	-	-	-	-
PSO-w	0	1.16e+01±5.00e+00	1	5.82e+02±3.16e+02	50	0.00e+00±0.00e+00	47	6.59e-04±2.64e-03
CLPSO	0	6.72e+00±2.35e+00	0	5.57e+02±2.34e+02	50	3.11e-12±1.55e-11	50	1.19e-07±1.86e-07
Rand-BFGS	0	2.96e+01±1.84e+01	0	1.38e+03±3.27e+02	50	5.23e-11±3.07e-10	0	8.27e+00±3.94e+00
DMS-L-PSO	0	4.42e+00±1.32e+00	0	3.93e+02±1.13e+02	50	0.00e+00±0.00e+00	50	3.85e-12±4.35e-12
PSO-w-BFGS	39	2.45e+00±5.07e+00	0	8.49e+02±3.29e+02	50	7.38e-13±5.12e-13	50	6.39e-13±5.79e-13
CLPSO-BFGS	47	2.08e-01±9.05e-01	8(4)	2.01e+02±1.80e+02	50	7.09e-13±6.04e-13	50	5.57e-13±4.41e-13
PSO-w-NBFGS	0	7.58e+00±4.87e+00	0	6.00e+02±3.00e+02	50	1.52e-10±1.32e-10	50	5.01e-08±2.62e-07
CLPSO-NBFGS	0	2.62e-00±1.02e-00	1	3.76e+02±1.79e+02	50	4.43e-13±3.76e-13	50	3.58e-7±2.52e-06
DMS-L-ASA	0	8.21e+00±1.89e+00	0	3.00e+02±2.10e+02	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
CLPSO-ASA	0	8.42e+00±3.38e+00	0	4.50e+02±2.48e+02	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
t-test value	1		0		-1		1	
	f_{17}		f_{18}		f_{19}		f_{20}	
GA	0	3.36e+00±0.00e+00	0	3.75e+03±0.00e+00	0	5.48e+02±0.00e+00	0	3.60e-01±0.00e+00
MA	0	3.66e+00±3.13e-01	25	3.15e+02±6.30e+02	0	4.19e+03±4.75e+03	0	8.49e-01±2.74e-01
PSO-w	0	3.41e+00±3.50e-01	0	1.81e+03±1.92e+03	0	1.27e+03±1.78e+03	0	8.50e-01±5.40e-01
CLPSO	0	3.40e+00±2.40e-01	0	1.84e+03±1.22e+03	0	2.61e+02±2.36e+02	0	2.60e-01±2.00e-01
Rand-BFGS	0	4.22e+00±2.00e-01	0	5.83e+03±1.49e+03	0	9.21e+00±7.44e+00	0	1.55e+01±6.81e+00
DMS-L-PSO	0	3.35e+00±4.00e-01	0	1.27e+03±1.23e+03	14	8.18e+01±3.47e+02	0	4.40e-01±1.90e-01
PSO-w-NBFGS	0	3.15e+00±4.40e-01	0	2.09e+03±2.11e+03	0	1.20e+02±3.05e+02	0	7.80e-01±2.90e-01
CLPSO-NBFGS	0	3.48e+00±3.40e-01	0	2.01e+03±1.34e+03	0	2.2e+02±5.27e+02	0	2.00e-01±1.60e-01
DMS-L-ASA	0	3.43e+00±2.90e-01	0	9.27e+02±3.65e+02	0	6.65e+02±4.91e+02	0	8.90e-01±1.40e-01
CLPSO-ASA	0	3.45e+00±3.20e-01	0	1.15e+03±9.79e+02	0	3.68e+02±5.89e+02	0	2.90e-01±2.40e-01
t-test value	0		-1		-1		0	

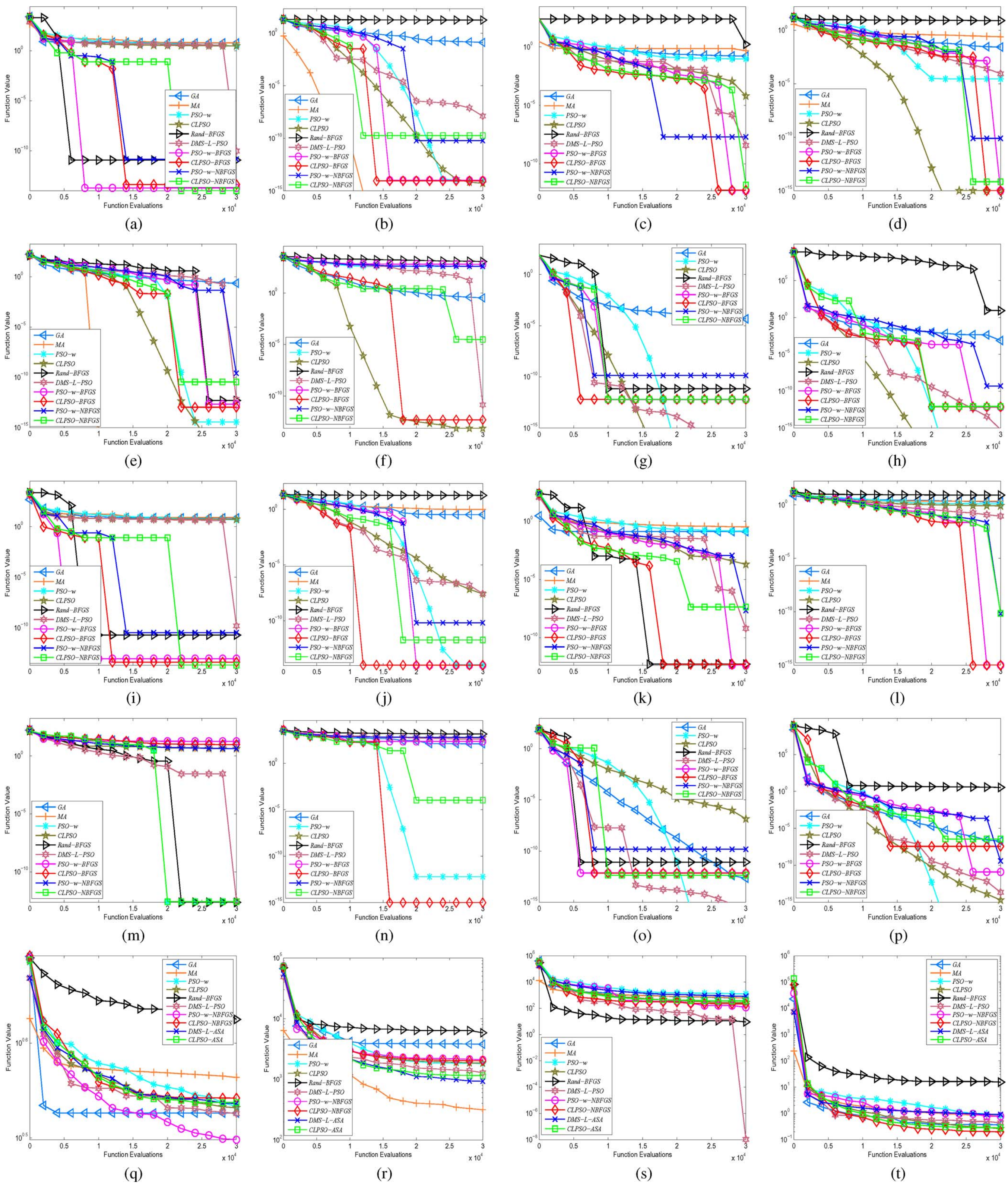


Fig. 4. Median convergence graphs of different algorithms on the 20 benchmark functions with ten dimensions. The figures record the mean value of the median function value of the benchmark functions. (a) f_1. (b) f_2. (c) f_3. (d) f_4. (e) f_5. (f) f_6. (g) f_7. (h) f_8. (i) f_9. (j) f_{10}. (k) f_{11}. (l) f_{12}. (m) f_{13}. (n) f_{14}. (o) f_{15}. (p) f_{16}. (q) f_{17}. (r) f_{18}. (s) f_{19}. (t) f_{20}.

in the figures. For the last four functions, we only use the numerical gradients. Hence, there are no results for PSO-w-BFGS and CLPSO-BFGS while the results for DMS-L-ASA and CLPSO-ASA are included.

In Table IV and Fig. 4, we can see that PSO-w-(N)BFGS and CLPSO-(N)BFGS have significantly improved their counterparts, i.e., PSO-w and CLPSO, respectively, both on the hit rate and convergence rate. In general, the performance of PSO-w-NBFGS and CLPSO-NBFGS is inferior to their analytical counterparts, namely, PSO-w-BFGS and CLPSO-BFGS, particularly on problem f_{13}. Two factors account for this problem. On one hand, the numerical gradient is generally less accurate than the analytical gradient, which may bring bias to the direction calculation in the BFGS method. For example, on the shifted rotated Rastrigin's function (f_{13}), the numerical gradient is not accurate because of the rotation. Then, the CLPSO-NBFGS and PSO-w-NBFGS methods will fail to identify the global optimum even the global optimality basin has been detected. Different from the shifted rotated Rastrigin's function, the CLPSO-NBFGS and PSO-w-NBFGS methods can obtain good performance on its shift counterpart (f_5). On the other hand, the time spent on the numerical gradient is much more than that spent on the analytical gradient. In Table II, we can see that the time spent on the analytical gradient calculation can be at most two times than that spent on the fitness function evaluation. However, the time spent on numerical gradient is D times than that spent on fitness function evaluation. Hence, given fixed function evaluations, there are fewer function evaluations left for particle evolutions in CLPSO-NBFGS and PSO-w-NBFGS.

PSO-w-BFGS outperforms PSO-w, particularly on f_1, f_3, f_5, f_9, f_{10}, f_{11}, f_{12}, and f_{13}. PSO-w-NBFGS significantly outperforms PSO-w on f_1, f_3, f_5, f_9, f_{10}, f_{11}, and f_{12} but shows no considerable improvement on f_{13}. For those functions that PSO-w has successfully found the global optima, PSO-w-(N)BFGS shows a faster convergence rate, as shown in Fig. 4. As to CLPSO-BFGS, it shows great improvements to CLPSO on hit rates on functions f_{10}, f_{11}, f_{12}, and f_{13}. However, CLPSO-NBFGS shows little improvement on f_{13}. In Fig. 4, we can see that CLPSO-(N)BFGS also converges faster than CLPSO for those functions on which the hit rates are 50. Generally speaking, CLPSO-(N)BFGS shows better performance than PSO-w-(N)BFGS, which is caused by the fact that the CLPSO has better global search ability [3].

Except for f_{14}, CLPSO-BFGS performs better than DMS-L-PSO on hit rates, particularly on functions f_3, f_5, f_6, f_{11}, f_{12}, and f_{13}. Except for f_{13}, CLPSO-NBFGS also shows great improvement compared with DMS-L-PSO. PSO-w-BFGS and PSO-w-NBFGS also show competitive results compared with DMS-L-PSO. According to the t-test results, the proposed hybrid strategy can obtain improved performance on hit rates on most functions.

In addition, because the accuracy of the hybrid strategy is mainly controlled by the BFGS method, we can change it to obtain results of different accuracy. This treatment will not improve the complexity of the hybrid method too much because of the territory technique, which prevents the particles from detecting the detected local optima. On function f_{12}, although the proposed strategy shows inferior t-test performance to DMS-L-PSO, it has a higher probability of finding the global optimum.

Among the 12 methods, Rand-BFGS can achieve good results on simple problems, such as the shifted Rosenbrock's function but performs poorly on most problems. Finally, in Table IV, we can observe that all the algorithms do not work well on function f_{14} as well as the last four algorithms. We can also conclude from the results that a simple GA may not be suitable for complicated problems while the MA is better. MA can obtain the best result on f_{18}, while DMS-L-PSO can obtain the best result on f_{19}. However, on the whole, our methods can obtain comparable performance compared with other methods.

For the hybrid methods with the (N)BFGS method, the number in brackets in Table IV represents the number of global optima obtained by the reconstruction technique. Take PSO-w-BFGS as example, 18 of the 44 global optima on function f_3 and 12 of the 26 global optima on function f_{12} are obtained by the reconstruction technique. Note that f_{12} is a nonseparable function that indicates that the multiple local optima obtained by the hybrid strategy, indeed, contain a wealth of information about the global optimum, and the reconstruction technique can be very useful to estimate the global solutions to both separable and nonseparable functions. However, if we use a nondeterministic optimization method, such as SA, as the local search method, we are not likely to obtain accurate enough local solutions. In such a case, the reconstruction operator may be not useful. However, SA may be useful for highly noised problems where the deterministic optimization methods absolutely cannot work.

2) Results of 30-D Problems: In the second experiment, all the algorithms are performed 50 times with 30-D on the 20 test functions. The hit rate (denoted by hit), the mean and variance of the final function values of various algorithms (denoted by $mean \pm variance$), and the t-test results are shown in Table V. The numbers in brackets of the hybrid methods are the results obtained by the reconstruction technique. Due to space limitation, the median convergence graphs of the different algorithms are not presented in this paper. In Table V, CLPSO-NBFGS and PSO-w-NBFGS also show the inferior performance compared with their counterparts, i.e., CLPSO-BFGS and PSO-w-BFGS, due to the same reason discussed in 10-D experiments. However, CLPSO-NBFGS and PSO-w-NBFGS achieve competitive performance compared with other methods on most problems. For CLPSO-NBFGS, except for functions f_{12}, f_{13}, and f_{14}, it shows very good performance on identifying the global optimum. As to PSO-w-NBFGS, it is not so good as CLPSO-NBFGS. However, it still outperforms PSO-w on the hit rate for f_1, f_3, f_8, f_9, f_{10}, and f_{11}. In conclusion, the hybrid method shows much improved performance compared with the context PSOs. For DMS-L-PSO, it performs the best on f_{12} but fails on f_5 and f_6 compared with CLPSO-BFGS and CLPSO-NBFGS. The t-test results for 30-D problems also show the competitive performance of the proposed method. On the last four functions, our methods are also comparable. Similar to the 10-D problem, the simple GA is also not so good on 30-D problems. In addition, the MA methods may fail on some problems. One possible reason is that too many local searches are invoked in the MA method.

Parameter Sensitivity Study: In the third experiment, three parameters, namely, the threshold LDI_0, the iteration K, and the number of the freed particles p_r, are studied. Here, we just take CLPSO-BFGS on function $f_1 - f_8$ with 10-D for the case study. When studying one parameter, we keep other parameters the same as in Table III. All the experiments are performed 50 times for each value, and the hit rates of each possible

TABLE V

RESULTS OF 20 BENCHMARK FUNCTIONS ON 30 DIMENSIONS. *hit* STANDS FOR THE SUCCESSFUL TIMES OF FINDING THE GLOBAL OPTIMUM, WHILE *mean* ± *variance* STANDS FOR THE MEAN AND VARIANCE OF THE FINAL FUNCTION VALUE, RESPECTIVELY

	f_1		f_2		f_3		f_4	
	hit rate	mean ± variance	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*
GA	*0*	2.77e+01±7.71e+00	0	2.80e-01±8.00e-02	0	*6.50e-01±2.20e-01*	0	4.20e-01±1.10e-01
MA	0	3.06e+01±2.25e+01	50	2.74e-008±5.47e-008	29	1.62e-01±2.07e-01	0	9.26e+00±3.44e+00
PSO-w	0	2.67e+01±1.71e+01	50	4.00e-14±1.40e-14	20	1.49e-02±1.95e-02	9	3.35e+00±2.60e+00
CLPSO	0	1.81e+01±3.98e+00	50	7.83e-13±3.71e-13	50	8.20e-14±3.54e-13	50	0.00e+00±0.00e+00
Rand-BFGS	50	3.73e-11±3.42e-11	0	1.91e+01±2.39e-01	50	5.99e-13±5.94e-13	0	3.53e+01±1.47e+00
DMS-L-PSO	50	1.16e-10±7.31e-11	50	2.80e-14±1.00e-14	50	0.00e+00±0.00e+00	48	4.23e-02±2.10e-01
PSO-w-BFGS	49	7.97e-02±5.64e-01	50(1)	2.64e-10±0.00e+00	50	6.75e-13±6.24e-13	17(5)	2.77e+00±1.80e+00
CLPSO-BFGS	50	5.56e-14±5.64e-14	50	2.31e-14±7.03e-15	50	7.43e-13±6.40e-13	50(39)	0.00e+00±0.00e+00
PSO-w-NBFGS	50	2.40e-14±2.67e-14	50(19)	8.07e-11±4.00e-10	50	1.08e-07±8.39e-08	5(5)	2.83e+00±2.01e+00
CLPSO-NBFGS	50	5.52e-15±3.31e-15	50	1.96e-12±6.49e-12	50	9.77e-08±6.63e-08	50(41)	7.42e-10±5.30e+00
DMS-L-ASA	0	2.41e+01±2.10e-01	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
CLPSO-ASA	0	2.04e+01±5.04e+00	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
t-test value	1		1		-1		0	

	f_5		f_6		f_7		f_8	
	hit rate	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*
GA	0	1.00e+00±5.30e-01	0	4.25e+00±3.34e+00	50	0.00e+00±0.00e+00	0	6.00e-02±4.00e-02
MA	50	0.00e+00±0.00e+00	0	7.94e+03±2.79e+03	-	-	-	-
PSO-w	0	2.40e+01±8.24e+00	0	1.12e+03±3.13e+02	50	0.00e+00±0.00e+00	45	1.10e-03±3.33e-03
CLPSO	50	0.00e+00±1.00e-15	50	1.09e-13±4.36e-13	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
Rand-BFGS	0	2.83e+02±4.55e+01	0	4.62e+03±4.30e+02	50	3.83e-11±3.96e-11	0	2.42e+01±5.56e+00
DMS-L-PSO	0	1.85e+01±3.62e+00	0	1.68e+03±8.09e+02	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
PSO-w-BFGS	0	3.20e+01±8.44e+00	0	2.08e+03±4.55e+02	50	7.25e-13±6.25e-13	50	0.00e+00±0.00e+00
CLPSO-BFGS	50	3.48e-13±5.29e-13	50	2.91e-13±6.74e-13	50	6.69e-13±5.83e-13	50	6.59e-13±6.58e-13
PSO-w-NBFGS	1	3.40e+01±9.19e+00	0	2.17e+03±5.21e+02	50	2.01e-15±1.77e-15	49(5)	3.3e-03±1.55e-02
CLPSO-NBFGS	50	1.23e-08±2.37e-08	50	2.16e-05±3.55e-05	50	5.39e-13±6.04e-13	50(1)	2.02e-11±1.39e-10
DMS-L-ASA	0	2.69e+01±3.49e+00	16	1.38e+02±1.43e+02	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
CLPSO-ASA	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
t-test value	1		0		-1		0	

	f_9		f_{10}		f_{11}		f_{12}	
	hit rate	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*
GA	0	2.90e+01±6.40e-01	0	2.62e+00±2.46e+00	0	7.10e-01±1.30e-01	0	1.62e+01±2.02e+00
MA	0	1.17e+02±1.46e+01	0	3.64e+00±7.25e+00	29	3.42e-02±6.36e-02	0	2.10e+01±2.84e+00
PSO-w	0	2.79e+01±1.39e+01	8	1.42e+00±7.33e-01	16	1.57e-02±1.80e-02	0	7.87e+00±2.85e+00
CLPSO	0	2.31e+01±2.09e+00	50	8.66e-06±2.40e-05	50	8.31e-07±2.34e-06	0	3.31e+00±1.39e+00
Rand-BFGS	50	8.06e-11±1.47e-10	0	1.91e+01±1.92e-01	50	4.76e-13±4.91e-13	0	3.47e+01±1.98e+00
DMS-L-PSO	50	1.81e-10±9.91e-11	50	3.40e-14±1.50e-14	50	5.60e-14±3.95e-13	37	3.37e-01±6.93e-01
PSO-w-BFGS	50	3.53e-14±3.71e-14	50(1)	6.16e-08±4.36e-07	50	5.46e-13±4.68e-13	0	4.69e+00±2.40e+00
CLPSO-BFGS	50	5.51e-14±5.97e-14	50	2.31e-14±7.17e-15	50	6.14e-13±5.65e-13	10(3)	1.03e+00±1.46e+00
PSO-w-NBFGS	49	7.97e-02±5.64e-01	42(1)	1.55e-01±4.31e-01	50	8.27e-08±4.59e-08	0	5.41e+00±2.31e+00
CLPSO-NBFGS	50	1.41e-14±1.34e-14	50	1.97e-12±1.01e-13	50	1.96e-07±1.33e-07	1	2.20e+00±1.90e+00
DMS-L-ASA	0	2.50e+01±1.80e-01	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00	0	6.10e-01±1.09e+00
DMS-L-ASA	0	2.54e+01±1.00e+00	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00	0	4.43e+00±2.21e+00
t-test value	1		1		-1		-1	

	f_{13}		f_{14}		f_{15}		f_{16}	
	hit rate	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*
GA	0	9.30e+01±3.79e+01	0	2.70e+03±7.54e+02	50	0.00e+00±0.00e+00	17	1.70e-01±8.00e-02
MA	0	1.03e+02±2.07e+01	-	-	-	-	-	-
PSO-w	0	5.43e+01±1.48e+01	0	2.33e+03±6.21e+02	50	6.13e-10±0.00e+00	43	1.54e-03±3.85e-03
CLPSO	0	3.45e+01±5.81e+00	0	2.05e+03±3.14e+02	50	1.05e-11±5.14e-11	50	0.00e+00±0.00e+00
Rand-BFGS	0	2.86e+02±3.92e+01	0	5.62e+03±4.21e+02	50	4.99e-11±4.54e-11	0	2.46e+01±6.26e+00
DMS-L-PSO	0	3.19e+01±5.47e+00	0	3.28e+03±5.92e+02	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
PSO-w-BFGS	0	5.00e+01±1.20e+01	0	3.59e+03±7.13e+02	50	6.21e-13±5.64e-13	50	5.54e-13±5.81e-13
CLPSO-BFGS	18	1.30e+01±1.24e+01	0	2.42e+03±4.00e+02	50	6.17e-13±5.59e-13	50	7.61e-13±6.17e-13
PSO-w-NBFGS	0	4.49e+01±1.10e+01	0	2.93e+03±7.60e+02	50	1.77e-10±1.62e-13	44	3.00e-03±1.40e-02
CLPSO-NBFGS	0	2.78e+01±5.63e+00	0	2.02e+03±3.55e+02	50	4.89e-13±4.83e-13	50	5.96e-13±5.99e-13
DMS-L-ASA	0	4.63e+01±5.32e+00		2.65e+03±6.74e+02	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
CLPSO-ASA	0	3.16e+01±7.33e+00		1.93e+03±3.58e+02	50	0.00e+00±0.00e+00	50	0.00e+00±0.00e+00
t-test value	1		0		-1		-1	

	f_{17}		f_{18}		f_{19}		f_{20}	
	hit rate	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*	*hit rate*	*mean* ± *variance*
GA	0	1.32+e01±0.00e+00	0	4.24e+04±0.00e+00	0	5.91e+04±0.00e+00	0	1.93e+00±0.00e+00
MA	0	1.08e+01±4.28e-01	0	1.00e+04±7.07e+02	0	9.33e+04±3.25e+04	0	5.26e+01±2.61e+01
PSO-w	0	1.29e+01±5.20e-01	0	1.37e+04±4.27e+03	0	5.55e+04±4.77e+04	0	4.57e+00±1.94e+00
CLPSO	0	1.30e+01±2.10e-01	0	1.41e+04±2.17e+03	0	2.13e+04±3.97e+03	0	1.01e+00±0.37e+00
Rand-BFGS	0	1.37e+01±3.00e-01	0	4.27e+04±3.72e+03	0	2.54e+03±1.11e+03	0	2.22e+02±8.38e+01
DMS-L-PSO	0	1.24e+01±2.30e-01	0	7.53e+03±1.70e+03	3	7.61e+03±5.71e+03	0	2.55e+00±7.40e-01
PSO-w-NBFGS	0	1.24e+01±3.90e-01	0	9.88e+03±3.53e+03	0	1.60e+03±2.54e+03	0	3.33e+00±5.20e-01
CLPSO-NBFGS	0	1.29e+01±2.20e-01	0	1.27e+04±3.62e+03	0	4.53e+03±3.35e+03	0	9.10e-01±3.70e-01
DMS-L-ASA	0	1.27e+01±2.70e-01	0	9.99e+03±1.26e+03	0	2.54e+04±9.28e+03	0	5.57e+00±4.40e-01
CLPSO-ASA	0	1.27e+01±3.80e-01	0	1.07e+04±2.63e+03	0	1.50e+04±1.00e+04	0	9.60e-01±3.40e-01
t-test value	0		0		0		0	

value are recorded. The final results are shown in Fig. 5. From Fig. 5, the hit rate, by varying LDI_0, has a wider range than the results of K and p_r. In general, the proposed strategy is more sensitive to LDI_0 than to K and p_r. For LDI_0, the best results are obtained at 0.01. And we know that if it is too small, the local search will be started mainly by K, which usually

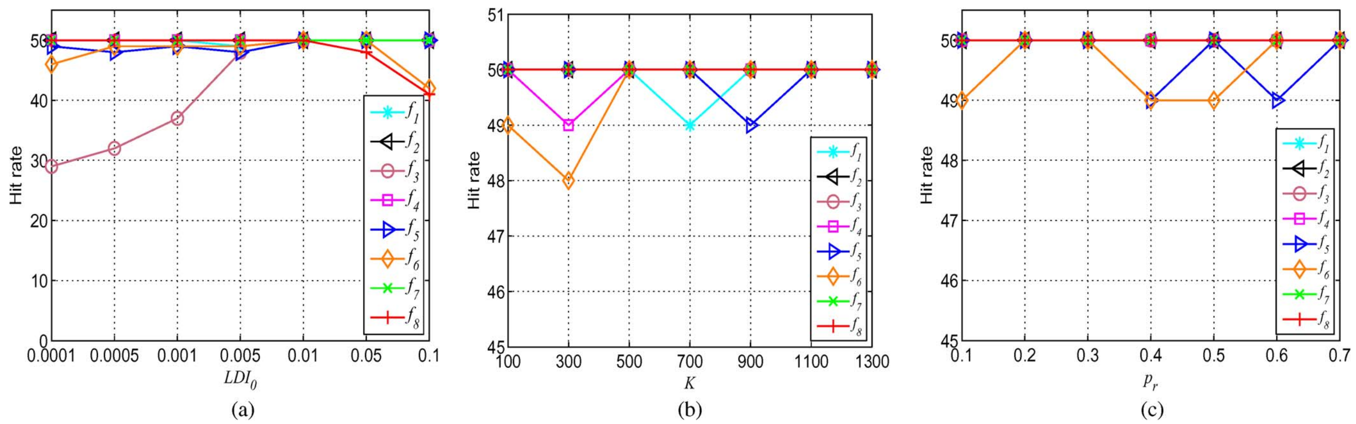


Fig. 5. Influences of the parameters of CLPSO-BFGS on eight benchmark functions $(f_1 - f_8)$. The figures record the successful times (hit rate) of 50 experiments with varying values on parameters LDI_0, K, and p_r, respectively. (a) Hit rate with various LDI_0. (b) Hit rate with various K. (c) Hit rate with various p_r.

does not show good performance, as shown in Fig. 5(a). On the other hand, if it is too large, the local search will be started too frequently, and the performance also degrades. Generally, we suggest that LDI_0 should be smaller than 0.05. From Fig. 5(b), for the iteration period K, CLPSO-BFGS obtains comparable results with all possible values, which indicate that the hybrid strategy is not sensitive to K. Actually, there is no need for K if the context PSO has a good local convergence property. However, a too small K may lead to too frequent local searches. Therefore, a large K would be favorable (even $K = \max\_iter$ is acceptable). For p_r, from Fig. 5(c), all the possible values can produce competitive results. However, p_r's from 0.2 to 0.3 perform the best.

IV. Conclusion

Premature convergence and slow convergence rate are two main deficiencies of PSOs. Meanwhile, deterministic optimization methods, such as the BFGS method, are known for their fast convergence but are quite sensitive to the starting point when solving nonconvex problems. In this paper, we proposed a new hybrid PSO-BFGS strategy for the global optimization of multimodal functions. To make the combination more efficient, an LDI is proposed to dynamically start the local search, and a reposition technique is proposed to keep the diversity of particles, which can effectively avoid the premature convergence problem. In addition, by adopting a territory technique, the proposed strategy can efficiently find multiple local (or global) optima using a small population. The benchmark test results demonstrate improved performance compared with other methods, particularly on the lower dimensional problems. The implementation of the hybrid strategy is straightforward and most of the informed PSO algorithms can be adopted as the context PSOs. However, different context algorithms may result in different performance. In the experiments, the CLPSO-(N)BFGS methods usually outperforms PSO-w-(N)BFGS in both the hit rate and convergence rate. This is mainly caused by the fact that CLPSO possesses a more complicated and effective particle learning structure than PSO-w, making CLPSO better than PSO-w in the use of the freed particles [3]. Therefore, to design more efficient rules to reuse the freed particles is a future direction.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions, which have considerably improved this paper. The authors would also like to thank Prof. P. N. Suganthan (Nanyang Technological University) for providing PSO codes and advice.

References

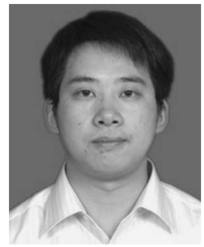
[1] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, 1995, pp. 1942–1948.
[2] J. J. Liang and P. N. Suganthan, "Dynamic multi-swarm particle swarm optimizer with local search," in *Proc. IEEE Congr. Evol. Comput.*, 2005, pp. 522–528.
[3] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, Jun. 2006.
[4] Y. Shi and R. C. Eberhart, "A modified particle swarm optimizer," in *Proc. IEEE Congr. Evol. Comput.*, 1998, pp. 69–73.
[5] M. Clerc and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, vol. 6, no. 1, pp. 58–73, Feb. 2002.
[6] Z. Zhan, J. Zhang, Y. Li, and H. Chung, "Adaptive particle swarm optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 6, pp. 1362–1381, Dec. 2009.
[7] S. Hsieh, T. Sun, C. Liu, and S. Tsai, "Efficient population utilization strategy for particle swarm optimizer," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 2, pp. 444–456, Apr. 2009.
[8] G. Yen and W. Leong, "Dynamic multiple swarms in multiobjective particle swarm optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 4, pp. 890–911, Jul. 2009.
[9] C. Juang, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 34, no. 2, pp. 997–1006, Apr. 2004.
[10] S. H. Ling, H. H. C. Iu, K. Y. Chan, H. K. Lam, B. C. W. Yeung, and F. H. Leung, "Hybrid particle swarm optimization with wavelet mutation and its industrial applications," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 3, pp. 743–763, Jun. 2008.
[11] Y. Chen, W. Peng, and M. Jian, "Particle swarm optimization with recombination and dynamic linkage discovery," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 6, pp. 1460–1470, Dec. 2007.
[12] J. Seo, C. Im, C. Heo, J. Kim, H. Jung, and C. Lee, "Multimodal function optimization based on particle swarm optimization," *IEEE Trans. Magn.*, vol. 42, no. 4, pp. 1095–1098, Apr. 2006.
[13] K. E. Parsopoulos and M. N. Vrahatis, "On the computation of all global minimizers through particle swarm optimization," *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 211–224, Jun. 2004.
[14] Y. W. Leung and Y. P. Wang, "An orthogonal genetic algorithm with quantization for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 5, no. 1, pp. 41–53, Feb. 2001.
[15] N. Noman and H. Iba, "Accelerating differential evolution using an adaptive local search," *IEEE Trans. Evol. Comput.*, vol. 12, no. 1, pp. 107–125, Feb. 2008.

[16] D. Liu, K. Tan, C. Goh, and W. Ho, “A multiobjective memetic algorithm based on particle swarm optimization,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 1, pp. 42–50, Feb. 2007.
[17] Q. Zhang, J. Sun, E. Tsang, and J. Ford, “Hybrid estimation of distribution algorithm for global optimization,” *Eng. Comput.*, vol. 21, no. 1, pp. 91–107, 2004.
[18] Y. S. Ong, M. H. Lim, N. Zhu, and K. W. Wong, “Classification of adaptive memetic algorithms: A comparative study,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 1, pp. 141–152, Feb. 2006.
[19] Q. H. Nguyen, Y. S. Ong, and M. H. Lim, “A probabilistic memetic framework,” *IEEE Trans. Evol. Comput.*, vol. 13, no. 3, pp. 604–623, Jun. 2009.
[20] H. Soh, Y. S. Ong, Q. C. Nguyen, Q. H. Nguyen, M. S. Habibullah, T. Hung, and J.-L. Kuo, “Discovering unique, low-energy pure water isomers: Memetic exploration, optimization and landscape analysis,” *IEEE Trans. Evol. Comput.*, vol. 14, no. 3, pp. 419–437, Jun. 2010.
[21] S. K. S. Fan and E. Zahara, “A hybrid simplex search and particle swarm optimization for unconstrained optimization,” *Eur. J. Oper. Res.*, vol. 181, no. 2, pp. 527–548, Sep. 2007.
[22] Z. Z. Zhou, Y. S. Ong, P. B. Nair, A. J. Keane, and K. Y. Lum, “Combining global and local surrogate models to accelerate evolutionary optimization,” *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 37, no. 1, pp. 66–76, Jan. 2007.
[23] L. S. Coelho and V. C. Mariani, “A novel chaotic particle swarm optimization approach using Hénon map and implicit filtering local search for economic load dispatch,” *Chaos Solitons Fractals*, vol. 39, no. 2, pp. 510–518, Jan. 2009.
[24] F. van den Bergh and A. P. Engelbrecht, “A cooperative approach to particle swarm optimization,” *IEEE Trans. Evol. Comput.*, vol. 8, no. 3, pp. 225–239, Jun. 2004.
[25] J. Bräuninger, “A variable metric algorithm for unconstrained minimization without evaluation of derivatives,” *Numer. Math.*, vol. 36, no. 4, pp. 359–373, Dec. 1981.
[26] R. H. Byrd, P. Lu, and J. Nocedal, “A limited memory algorithm for bound constrained optimization,” *SIAM J. Sci. Stat. Comput.*, vol. 16, no. 5, pp. 1190–1208, Sep. 1995.
[27] H. G. Cobb, “Is the genetic algorithm a cooperative learner?,” in *Foundations of Genetic Algorithms 2*. San Mateo, CA: Morgan Kaufmann, 1992.
[28] Q. Zhang, J. Sun, and E. Tsang, “Evolutionary algorithm with the guided mutation for the maximum clique problem,” *IEEE Trans. Evol. Comput.*, vol. 9, no. 2, pp. 192–200, Apr. 2005.
[29] P. N. Suganthan, N. Hansen, J. J. Liang, K. Deb, Y.-P. Chen, A. Auger, and S. Tiwari, “Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization,” Nanyang Technol. Univ., Singapore, Rep. 2005005, May 2005.
[30] D. Bratton and J. Kennedy, “Defining a standard for particle swarm optimization,” in *Proc. Swarm Intell. Symp.*, 2007, pp. 120–127.
[31] L. Ingber, “Adaptive simulated annealing (ASA),” Lester Ingber Research, Tech. Rep., 2008.
[32] C. Houck, J. Joines, and M. Kay, “A genetic algorithm for function optimization: A Matlab implementation,” North Carolina State Univ., Raleigh, NC, Tech. Rep., 1995.

Shutao Li (M’07) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Hunan University, Changsha, China, in 1995, 1997, and 2001, respectively.

He was with the College of Electrical and Information Engineering, Hunan University, in 2001. He was a Research Associate in the Department of Computer Science, Hong Kong University of Science and Technology (HKUST), Clear Water Bay, Hong Kong, from May 2001 to October 2001. From November 2002 to November 2003, he was a Postdoctoral Fellow at Royal Holloway College, University of London, London, U.K., working with Prof. J. Shawe-Taylor. From April 2005 to June 2005, he was with the Department of Computer Science, HKUST, as a Visiting Professor. He is currently a Full Professor in the College of Electrical and Information Engineering, Hunan University. He is the author or coauthor of more than 100 refereed papers. His professional interests include computational intelligence, information fusion, pattern recognition, and image processing.

Dr. Li served as a Member of the Neural Networks Technical Committee from 2007 to 2008. He received two Second-Grade National Awards at the Science and Technology Progress of China in 2004 and 2006.



Mingkui Tan received the B.S. and M.S. degrees from Hunan University, Changsha, China, in 2006 and 2009, respectively. He is currently working toward the Ph.D. degree at Nanyang Technological University, Singapore.

His technical interests include particle swarm optimization and large-scale machine learning.

Ivor W. Tsang received the Ph.D. degree in computer science from Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong, in 2007.

He is currently an Assistant Professor in the School of Computer Engineering, Nanyang Technological University (NTU), Singapore. He is also the Deputy Director of the Center for Computational Intelligence, NTU. His current research interests include machine learning, kernel methods, large-scale optimization and its applications to data mining, and pattern recognitions.

Dr. Tsang received the prestigious IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding Paper Award for 2004 in 2006. He clinched the second-class prize of the National Natural Science Award from the Ministry of Education, China, for 2008 in 2009. His work on transfer learning for visual event recognition was awarded the Best Student Paper Prize at the 22nd IEEE Computer Society Conference on Computer Vision and Pattern Recognition in 2010. His work on speech adaptation earned him the Best Paper Award from the IEEE Hong Kong Chapter of Signal Processing Postgraduate Forum in 2006. He also received the Microsoft Fellowship in 2005.

James Tin-Yau Kwok received the Ph.D. degree in computer science from Hong Kong University of Science and Technology (HKUST), Clear Water Bay, Hong Kong, in 1996.

He then joined the Department of Computer Science, Hong Kong Baptist University, Kowloon, Hong Kong, as an Assistant Professor. He returned to HKUST in 2000 and is currently an Associate Professor in the Department of Computer Science and Engineering. He is an Associate Editor for *Neurocomputing*. His research interests include kernel methods, machine learning, pattern recognition, and artificial neural networks.

Dr. Kwok is an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS. He also received the IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding 2004 Paper Award in 2006.