

Comprehensive Learning Particle Swarm Optimization with Tabu Operator Based on Ripple Neighborhood for Global Optimization

Jin Qi, Bin Xu, Kun Wang

School of Internet of Things

Nanjing University of Posts and Telecommunications

Nanjing, China

qijin@njupt.edu.cn, xubin.njupt@foxmail.com,

kwang@njupt.edu.cn

Xi Yin, Xiaoxuan Hu, Yanfei Sun

School of Automation

Nanjing University of Posts and Telecommunications

Nanjing, China

yinximm2009@163.com, comichu88@qq.com,

sunyanfei@njupt.edu.cn

Abstract—For the weak convergence at the latter stage of the comprehensive learning particle swarm optimizer (CLPSO), we put forward a new CLPSO based on Tabu search to enhance the performance. Inspired by the phenomenon of water waves, a Ripple Neighborhood (RP) structure based on the Gaussian distribution is proposed to construct a new adaptive neighborhood structure to guide the selection of candidate solutions in Tabu search, which solves the problem of low convergence and improves the quality of the solution in CLPSO. Experimental results on the standard 26 test functions show that the proposed algorithm achieves a better performance compared with CLPSO.

Keywords—comprehensive learning particle swarm optimizer (CLPSO); Tabu search; Gaussian distribution; parameter adaptive

I. INTRODUCTION

Many real world problems can be modeled as the optimization problem. Equation (1) is to express a D-dimensional unconstrained optimization problem for minimization:

$$\min f(x), x \in R^n \quad (1)$$

For the complex optimization problems which are high-dimensional, non-convex and non-differentiability, the traditional approaches are usually unable to optimize by using the gradient or sub-gradient information of the objective function. Therefore, the solving effect is not ideal. With many advantages such as strong global search ability, fast convergence speed, high search efficiency and robustness, swarm intelligence algorithms perform well on complex function optimization problems and thus provide a new way to solve this kind of optimization problems.

After comprehensive learning particle swarm optimizer (CLPSO) first proposed, many experts have had corresponding researches in enhancing CLPSO's performance and extending its application. Liang and Suganthan [1] proposed the adaptive CLPSO based on history learning. In this algorithm, the particle learning probability changed adaptively along with the

difference of biggest improvements obtained from past several generations. At the same time, the improvement direction of history is added to the update of particle velocity[2]. Ismail [3] proposed SACLPSO to make some adaptive changes on related parameters in CLPSO. In order to improve low precision in CLPSO, Xiang Yu [4] put forward the ECLPSO which introduced the disturbance term in the velocity updating of particle to obtain a higher search capabilities. In ECLPSO, the learning probability of particle not only depended on the optimal fitness value of particle but also relied on the exploitation progress state of particle, which promotes the convergence. Hasanzadeh. M [5] introduced automatic learning machine into CLPSO.

There has been a number of improvements of CLPSO and thus the performance of CLPSO has also been improved accordingly, the CLPSO and its variants have better effect on solving some of the multi-peak problems (such as the complex asymmetric problem). As for uni-modal problems and other multi-peak problems, the effect is not satisfied. The convergence speed becomes slow in the later stage of the algorithm and the precision of the solution is not good enough.

Thus, we put forward a new CLPSO based on Tabu search to enhance the local search ability. Inspired by the phenomenon of water waves, a Ripple Neighborhood (RP) structure based on the Gaussian distribution is proposed to guide the selection of candidate solutions, which improve the low convergence and the quality of the solution in CLPSO to some extent.

The remaining parts are organized as follows: Section II gives a specific statement about CLPSO and Tabu search. Section III expounds the CLPSO+Tabu algorithms in detail. Section IV verifies the performance of the algorithm on the benchmark functions, and compares the CLPSO+Tabu algorithms with CLPSO algorithm. Section V makes the conclusion.

II. RELATED WORK

A. CLPSO

In order to utilize the useful information effectively during the process of historical search, J.J. Liang [1] proposed particle swarm optimization algorithm based on comprehensive learning strategy (CLPSO) on the basis of the original PSO in 2006. CLPSO uses a novel learning method and utilizes history optimal information of all other particles to update the velocities of particles, which maintains the diversity of particles in the algorithm and partly avoids the premature convergence.

CLPSO uses a learning strategy which can not only implement search in a D-dimension space but also choose different learning objects in different dimensions independently. The updating formulas of the speed and position of CLPSO are:

$$V_i^d \leftarrow \omega * V_i^d + c * rand_i^d * (pbest_{f_i(d)}^d - X_i^d) \quad (2)$$

$$X_i^d \leftarrow X_i^d + V_i^d \quad (3)$$

Where $f_i = [f_i(1), f_i(2), \dots, f_i(D)]$ defines $pbest$ of the i -th particle. $rand_i^d$ is a random number in the uniform distribution $[0, 1]$. $pbest_{f_i(d)}^d$ is the d -th dimension of $pbest$ of the i -th particle, which represents the learning object of the i -th particle in d -th dimension. This learning object is decided by learning probability Pc through the following strategies. Each particle produces a random number in $[0, 1]$. If the random number is greater than the predetermined learning probability Pc for particle i , set the historical best position $pbest$ as the learning object. Otherwise, select two individuals randomly from the population and pick out the best historical best position as the learning object according to the tournament selection strategy.

CLPSO has a good performance on dealing with multimodal problems, but the effect is not good enough on uni-modal problems.

B. Tabu search

Tabu search, which is effective for leaping out of the local optima, is a kind of modern heuristic algorithm proposed by Fred Glover of University of Colorado in 1986 [6]. Tabu search can avoid circuitous search by introducing a flexible storage structure and corresponding Tabu criteria, thereby guaranteeing the effective exploration to achieve global optimization.

The basic idea of classic Tabu algorithm [7] is preset a current solution (initial solution) and a neighborhood structure, and then generate several candidate solutions in the neighborhood of the current solution. If the target value corresponding to the best candidate solution is better than the state of "best so far", then ignore the Tabu characteristic of the best candidate solution. Using the best candidate solution to replace the current solution and the state of "best so far", and the corresponding solution is added into the Tabu list. At the

same time the term of the solution in the Tabu list is modified. If there is no such a candidate solution, then the best state which has non-Tabu characteristic among candidate solutions is used as the new current solution without comparing it with the current solution. Moreover, the corresponding solution is added into the Tabu list and the term of the solution in the Tabu list is modified. Then the above iterative search process is repeated until meeting the stopping criterion.

The aspiration criterion applied in this article is based on the global form of the fitness value criterion. The specific operation is: if the fitness value of a candidate solution with Tabu characteristic is better than the state of "best so far", then ignoring the Tabu characteristic and the candidate solution is relieved from Tabu list as the current state and the new state of "best so far". This criterion helps the algorithm to search a better solution [8].

Theoretically, the termination criterion is to achieve the traversal state space in the condition of Tabu length sufficiently enough. In the practical application, the criterion is often difficult to achieve. In this paper, the termination criterion is as follows: the maximum iteration, which is the number of total circulation after each run, is preset.

III. CLPSO WITH TABU OPERATOR

A. Ripple Neighborhood

Inspired by the ripple phenomena [9], this paper puts forward a new neighborhood structure, which simulates the ripple phenomena based on the Gaussian distribution and it can adaptively control the distribution range, to maintain the diversity of neighborhood solutions and guarantee the quality.

Gaussian distribution is also known as Normal distribution. The probability density function of normal distribution is called the bell curve because of its bell-shaped appearance. The square of certain interval on transverse of normal curve reflects the probability of variables lying in this interval, in which mathematical expectation μ decides the center position of normal curve, standard deviation σ decides the degree of steep or flat.

$$3\text{-sigma principle: } P(\mu - \sigma < X \leq \mu + \sigma) = 68.3\%$$

$$P(\mu - 2\sigma < X \leq \mu + 2\sigma) = 95.4\%$$

$$P(\mu - 3\sigma < X \leq \mu + 3\sigma) = 99.7\%$$

3-sigma principle means that, in normal curve, the square between one standard deviation away from the expectation μ is 68.3%, the square between two standard deviation away from the expectation μ is 95.4%, the square between three standard deviation away from the expectation μ is 99.7%. It means that 99.7% of values drawn from Gaussian distribution are within one standard deviation σ away from the expectation μ .

The proposed Tabu in this paper adopts 3-sigma principle of Gaussian distribution. The initial solutions are set to location parameter μ . It ensures that the candidate solutions will be selected from the area of three standard deviations away from the initial solutions and the shorter distance from the initial solutions, the greater probability that random variable will be selected. The size of σ means the range of

candidate solutions. When σ is large, the corresponding range of candidate solutions is large, so it can improve the candidate solutions' diversity by increasing the σ value. When σ is small, the corresponding range of candidate solutions is small, and this can enhance candidate solutions' quality. To control the size of σ , this paper introduces the ripple phenomena into neighborhood structure to form the adaptive dynamic change strategy. The strategy changes σ adaptively along with the evolutionary generation and thus to influence the range of candidate solutions.

We take a water-drop or stone as the initial solutions to cause a water ripple. The out diffusion of the ripple corresponds to the increase of the range of neighborhood solutions. The ripple is the transmission of energy, for the resistance of real world, energy is consumed little by little, the wave length enlarges accordingly, and the wave crest is gradually reduced until the energy is used up. σ value is the distance between wave crest and wave source. With the spread of the ripple, the interval of each σ will become larger and larger. When the result improved, we use it as a new wave source to trigger another round spread of ripple. This paper ignores the superposition of new and original ripple.

B. CLPSO with Tabu based on ripple neighborhood

In order to further enhance local search capabilities of CLPSO and improve solution quality, we introduce the Tabu search to CLPSO. In this paper, we constitute the *Strategy1* by integrating the ripple Gaussian neighborhood structure into Tabu algorithm. To increase the new neighborhood structure's diversity, the *Strategy2* disorganizes the change order of σ in the *Strategy1* to increase the probability of obtaining more accurate solutions.

Strategy1: The range of neighborhood solutions is designed by simulating the spread of the water ripple. Take the initial solution as the water drop to cause the spread of the water ripple. The range of neighborhood solutions turns bigger when the out forward diffusion of the ripple, which means σ turns larger. In each evolution, if the global best solution is updated, we use the new global best solution as the initial solution, which means a new drop causing a new round of spread. Otherwise, keep the last spread going on until reaching the edge of the spread.

Take the six advantage σ values {0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1}, change σ from small to big in every evolution. If the state of 'best so far' is not improved for 50 continuous generations, the *Strategy1* will turn the σ to a big one. When the σ is the biggest one and the state of 'best so far' is not improved for 50 continuous generations, the *Strategy1* will reset the σ with the smallest one. Then, keep operating until reaching the stop condition.

Strategy2: Take the six σ values in *Strategy1* {0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1}, select one σ randomly and apply it into the algorithm. If the state of 'best so far' is not improved for 50 continuous generations, choose a new one excluding the last one to determine the range of candidate solutions. Repeat the step above until reaching the stop condition.

The total flow of the proposed algorithm is presented in TABLE I. According to the above two strategies, CLPSO with Tabu operator based on *Strategy1* and *Strategy2* are proposed, which are called CLPSO+Tabu1 and CLPSO+Tabu2 respectively. The classic CLPSO algorithm is good at solving multimodal problems due to the use of a novel learning strategy. It is not sensible to use CLPSO to optimize unimodal problems and some other problems. Experiments show that CLPSO+Tabu algorithms are more effective than classical CLPSO algorithm because of the introduction of Tabu algorithm. Tabu helps CLPSO to overcome the lack of local search ability and increase the diversity of population. Results of the two algorithms are outstanding when optimizing rotate functions. And in the dispersed and ill-conditioned problem, results of CLPSO+Tabu algorithms are improved to some extent.

TABLE I. THE FLOW OF CLPSO+TABU ALGORITHM

Algorithm: The flow of CLPSO+Tabu algorithm	
Input:	test function and parameter setting
Parameters:	the size of the population: popsize, the max generation: max_gen, cross probability: pc, mutate probability: pm
Output:	the best solution of current iteration
Step 1:	Initialization
Step 1.1	Initialize the position of every individual in population randomly
Step 1.2	Compute the fitness of initial individuals, then store the fitness of the best solution in gbestval and save the correspond best solution in gbest
Step 1.3	Initialize the speed of every individual in population randomly
Step 2:	Select the learning object by tournament selection
	Address to every dimension, select two individuals randomly from the population, according to the size of the fitness of the two individuals' pbests, the superior one is winner. If the value is same, select one randomly. Then use the superior one as learning object. If one individual's all learning object is its own pbest, select one dimension randomly to study the pbest of others.
Step 3:	Update
	Update the individuals' position and speed according to the position and speed update function (2) and (3), choose the best solution from new individuals and update the pbest and pbestval.
Step 4:	Loop Step 2 and Step 3 , then stop CLPSO
	When the number of FEs reaches to the threshold 300000, stop the evolution of CLPSO and take the last best solution as initial solution.
Step 5:	Use the Tabu algorithm to update individuals
Step 5.1	Select candidate solutions from the neighborhood of initial solution generated in Step 4
Step 5.2	Choose the best one from candidate solutions as the state of 'best so far'
Step 5.3	Update the initial solution, Tabu list and the state of 'best so far', according to whether candidate solutions satisfy the aspiration criterion
Step 5.4	Judge the Tabu attribute of candidate solutions, and update the initial solution and Tabu list.
Step 6:	Stop the Tabu search
	If the state of 'best so far' isn't improved for 50 continuous generations, stop the search and output the last state of 'best so far'.
Step 7:	Add 1 to iterations and return to the Step 1 , when iterations reach to the max iterations, stop the evolution and output the best state of 'best so far'.

IV. EXPERIMENTS AND ANALYSIS

A. Benchmark Functions

To verify the performance completely, experiments are conducted on 26 benchmark functions with different characteristics [10].

There are two problems in traditional benchmark functions which are used to estimate optimization algorithm. One is that the global best solution is usually distributed over the center area of the search space. The other one is that the local best solutions are usually distributed in alignment and the variables are separable. To solve these two problems, shifting and rotating the functions to let the functions are closed to the real issues [10].

In [10], the functions' global best solution is shifted to different positions in each dimension. Based on the characteristics, the benchmark functions can be divided to three groups: (1) benchmark functions with separable variables; (2) benchmark functions with mis-scaled variables; (3) benchmark functions with noisy landscapes. For the details, please refer to [10].

B. Parameter Setting

1. Learning probability

The set of learning probability has effect on the learning ability of each particle. Experience [11] shows that the learning probability can be represented in (4).

$$Lp_i = Lp_{\min} + (Lp_{\max} - Lp_{\min}) \frac{\exp(\frac{10(i-1)}{N-1}) - 1}{\exp(10) - 1} \quad (4)$$

Where Lp_{\max} and Lp_{\min} correspond to the maximum and minimum of learn probability respectively. It is verified that $Lp_{\max} = 0.5$ and $Lp_{\min} = 0.05$ [4]. N represents the number of particles in population .

2. Inertia weight

It shows that bigger inertial weight is more helpful to jump off local optimal points, and smaller inertia weight is better for the algorithm convergence and improving the search quality [4]. Inertia weight can be represented by (5). Usually, the value of inertia weight is between 0.1 and 0.9.

$$w = w_{\max} - \frac{k}{k_{\max}} (w_{\max} - w_{\min}) \quad (5)$$

Where w_{\max} and w_{\min} are the maximum and minimum value of inertia weight respectively. Usually, $w_{\max} = 0.9$, $w_{\min} = 0.4$

3. Update refreshing gap

In order to ensure that the particles can learn from good objects as much as possible and do not waste time on poor objects, refreshing gap m is set for the selection process of learning object. After the fitness value of the particle remains unchanged for m generations, the learning object is redistributed. Otherwise, the learning object is unchanged.

4. Tabu list and Tabu length

Tabu length is that the maximum number of times which Tabu object are not allowed being selected without considering the aspiration criterion. Only when the term of object is 0, the object is lifted. Tabu length is a very important

parameter. Its size affects the search process and behavior of the whole algorithm. On the one hand, the amount of calculation and storage should be as small as possible, which requires Tabu length as small as possible. On the other hand, if Tabu length is too short, it will cause the circular search. The Tabu length is set to 8 in this paper.

5. Other factors in CLPSO

Acceleration coefficient is usually 1.5. The velocity range of particle is determined by the predetermined scope of the particle position. The maximum speed is 20% of the particle position range.

C. Experiment Results

In the experiment, all the algorithms set the dimension size of benchmark functions to 30, set the max number of function evaluations to 900000, set the number of particles in CLPSO to 60, set the number of neighborhood solutions in Tabu search to 100 and set the Tabu list to 60. For all benchmark functions, every algorithm carries out 10 independent runs.

TABLE II shows the results of CLPSO+Tabu1, CLPSO+Tabu2 and the classical CLPSO tested on 26 benchmark functions with 10 independent runs. The last three lines are the Wilcoxon rank sum test between CLPSO+Tabu1 and CLPSO, CLPSO+Tabu2 and CLPSO, when the confidence level is 0.05. In TABLE II, the number zero means that the algorithm result has the same distribution as CLPSO, the number one means that the algorithm has a different distribution from CLPSO. When the distribution is different, compare the optimal mean value. We found that there are 16 functions whose Wilcoxon rank sum test is one in the CLPSO+Tabu1 algorithm, among these 16 functions, 9 functions have better solutions and 4 functions have the same solutions, but there are 3 functions turn to worse. In CLPSO+Tabu2, there are 13 functions whose Wilcoxon rank sum test is 1, and 9 functions of them have better solutions and one functions have the same solutions, but there are 3 functions turn to worse.

Convergence procedure of part of 26 functions is shown in Fig.1. The horizontal axis means the number of fitness evolutions, and the vertical axis is the logarithmic form of present global optimal value. It is shown that CLPSO+Tabu1 and CLPSO+Tabu2 have a large improvement on the precision of global optimal value on F2, F4, F5, F12, F15, F17, F18, F19, and F25 (at least one order of magnitude).

D. Analysis of experimental results:

TABLE II shows the statistical results of optimal values. For F1 to F5 (which are uni-modal problems), most algorithms can solve the problems F1 and F3 perfectly. CLPSO+Tabu has made more excellent improvement than CLPSO on other problems.

Functions F6 to F11 are discrete multimodal problems, the local optimal values become more with the increase of dimension. Functions F8 and F9 are the hardest two problems. On these functions, the performance of CLPSO+Tabu algorithms is the same as the classic CLPSO.

The uni-modal function by Rosenbrock and the multimodal function by Rastrigin are the most two challenge problems. Functions F12 to F14 are mis-scaled problems generated from these two problems. CLPSO+Tabu has slightly improvement in these three functions. Function F14 needs powerful local search capability. It means that CLPSO+Tabu is short of forceful local search capability. Function F15 is a problem generated by F2, the Gaussian noise system makes it difficult to find global optimum values. On this issue, the CLPSO+Tabu algorithms have improved at least one order of magnitude when it compared with the classical CLPSO

algorithm. Functions F16 to F26 are designed to test the capability of solving ill-conditioned and non-separable problems. F17, F18 and F19 are convex quadratic terms, they can be converted into Sphere problems. F23 to F25 are multimodal problem which are extended from original Ackley, Griewank, Rastrigin problems. As can be seen from the TABLE II, CLPSO+Tabu keeps and maintains the diversity and quality due to the introduction of Gaussian distribution that integrate with ripple phenomenon. It makes improvement in most functions.

TABLE II. OPTIMIZATION RESULTS OF CLPSO+TABU FOR FUNCTIONS 1-26

Functions	CLPSO+Tabu1		CLPSO+Tabu2		CLPSO	
	<i>gbestVal mean</i>	<i>gbestVal std</i>	<i>gbestVal mean</i>	<i>gbestVal std</i>	<i>gbestVal mean</i>	<i>gbestVal std</i>
F1	1.14E-14	2.40E-14	1.14E-14	2.40E-14	1.71E-14	2.75E-14
F2	6.07E-02	2.57E-02	6.93E-03	9.89E-03	1.26E+01	2.60E+01
F3	5.68E-14	0.00E+00	5.68E-14	0.00E+00	7.96E-14	2.94E-14
F4	2.10E-02	1.40E-02	4.24E-03	5.37E-03	1.96E-02	3.33E-02
F5	9.44E+00	7.70E+00	1.20E+01	8.82E+00	1.96E+01	1.97E+01
F6	1.14E-13	0.00E+00	1.14E-13	0.00E+00	1.19E-13	1.80E-14
F7	1.19E+03	3.30E-13	1.19E+03	7.58E-14	1.19E+03	3.03E-13
F8	5.12E-14	1.80E-14	4.55E-14	2.40E-14	5.12E-14	1.80E-14
F9	5.68E-14	0.00E+00	5.12E-14	1.80E-14	5.68E-14	0.00E+00
F10	2.27E-14	2.94E-14	1.71E-14	2.75E-14	5.68E-15	1.80E-14
F11	2.84E-14	3.00E-14	3.41E-14	2.94E-14	1.71E-14	2.75E-14
F12	1.66E+02	2.06E+02	6.85E+01	5.58E+01	1.30E+02	7.45E+01
F13	5.68E-14	0.00E+00	4.55E-14	2.40E-14	5.12E-14	1.80E-14
F14	5.68E-14	0.00E+00	4.55E-14	2.40E-14	5.68E-14	0.00E+00
F15	3.85E+01	1.20E+02	8.41E+02	1.41E+03	1.55E+03	5.40E+02
F16	8.53E-14	3.00E-14	8.53E-14	3.00E-14	8.53E-14	3.00E-14
F17	5.97E-01	1.41E-01	1.31E-01	2.39E-01	9.50E+01	9.77E+01
F18	3.05E-01	1.53E-01	1.07E-01	2.03E-01	4.81E+02	3.35E+02
F19	1.09E+05	1.76E+05	2.45E+05	2.74E+05	5.67E+06	6.50E+07
F20	4.18E-03	2.09E-03	1.15E-03	1.11E-03	4.15E-03	1.91E-03
F21	5.48E+01	7.33E+01	9.50E+01	9.81E+01	8.68E+01	1.35E+02
F22	2.15E-07	2.32E-07	1.70E-06	3.57E-06	4.58E-10	7.73E-07
F23	3.83E+03	3.39E-13	3.83E+03	3.39E-13	3.83E+03	4.29E-13
F24	4.93E+01	1.31E+01	5.00E+01	1.24E+01	7.72E+01	7.49E+00
F25	2.07E+02	6.34E+02	1.44E+03	1.66E+03	3.51E+03	9.55E+02
F26	1.22E-01	1.96E-01	2.66E-02	2.16E-02	6.71E-03	2.48E-03
Wilcoxon			1			0
CLPSO+Tabu1			16			10
CLPSO+Tabu2			13			13

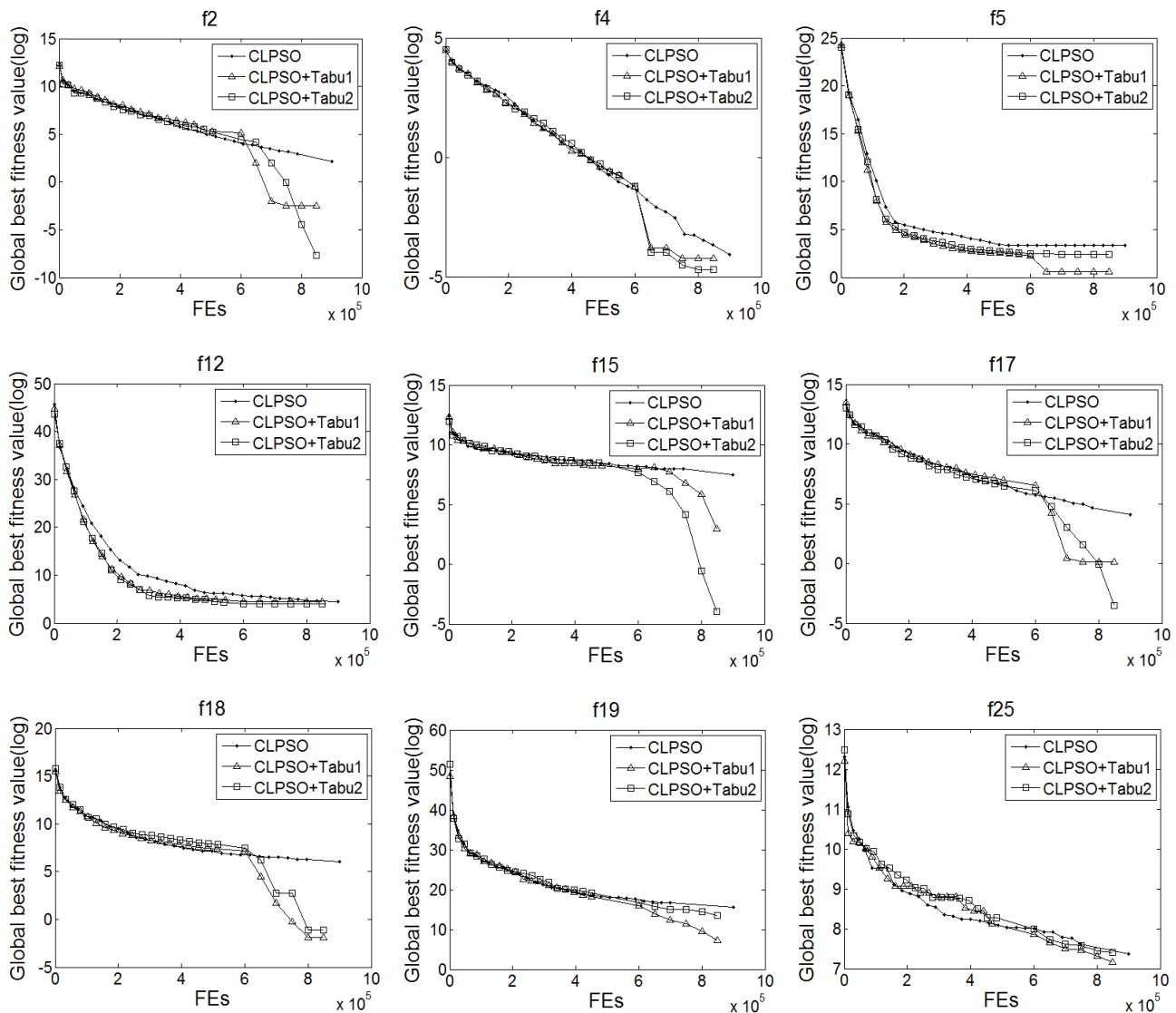


Figure 1. Convergence characteristics of CLPSO+Tabu1, CLPSO+Tabu2 and CLPSO on F2, F4, F5, F12, F15, F17, F18, F19, and F25

Compare with CLPSO, CLPSO+Tabu1 have better results on 12 functions, in which seven of them have a large improvement of at least one order of magnitude (namely F2, F5, F15, F17, F18, F19 and F25). CLPSO+Tabu2 has better values on 18 functions, and seven of them have a large improvement of at least one order of magnitude (namely F2, F4, F12, F15, F17, F18 and F19). It can be seen that CLPSO+Tabu achieves competitive results than the classical CLPSO on uni-modal problems, rotate problems, non-separable problems and ill-condition problems.

V. CONCLUSION

Aimed at improving the robustness and partial search ability of CLPSO, this paper introduces Tabu search into CLPSO. However, the capacity and neighborhood structure of Tabu search have a close relation to Tabu list, which greatly affects Tabu search. Therefore, the paper adapts neighborhood structure of Tabu search algorithm by applying ripple inspired Gaussian distribution operator to form a new neighborhood structure. Experimental results on 26

benchmark functions with different characteristics show that the proposed algorithms, compared with conventional CLPSO algorithm, have better performance on most of the test functions in terms of solution quality and convergence speed.

The two CLPSO+Tabu algorithms with different neighborhood structures have different results in the experiments. Based on *Strategy2*, CLPSO+Tabu2 algorithm performs more robustness with its adaptive strategy.

ACKNOWLEDGMENT

This paper was supported by National Natural Science Foundation of China (61401225), China Postdoctoral Science Foundation funded project (2015M571790), NUPTSF (Grant No. NY213047, NY213050, NY214102, NY214098).

REFERENCES

- [1] J. J. Liang, A. K. Qin, P. N. Suganthan, S. Baskar, "Evaluation of comprehensive learning particle swarm optimizer," in *Neural Information Processing*. vol. 3316, N. R. Pal, N. Kasabov, R. K. Mudi, S.

- Pal, S. K. Parui, Eds., ed Berlin: Springer-Verlag Berlin, 2004, pp. 230-235.
- [2] J. J. Liang, P. N. Suganthan, "Adaptive comprehensive learning particle swarm optimizer with history learning," in *Simulated Evolution and Learning, Proceedings*. vol. 4247, T. D. Wang, X. Li, S. H. Chen, X. Wang, H. Abbass, H. Iba, et al., Eds., ed Berlin: Springer-Verlag Berlin, 2006, pp. 213-220.
 - [3] A. Ismail, A. P. Engelbrecht, "The Self-adaptive Comprehensive Learning Particle Swarm Optimizer," *Swarm Intelligence (Ants 2012)*, vol. 7461, pp. 156-167, 2012.
 - [4] X. Yu, X. Q. Zhang, "Enhanced comprehensive learning particle swarm optimization," *Applied Mathematics and Computation*, vol. 242, pp. 265-276, Sep 2014.
 - [5] M. Hasanzadeh, M. R. Meybodi, M. M. Ebadzadeh, "Adaptive Parameter Selection in Comprehensive Learning Particle Swarm Optimizer," *Artificial Intelligence and Signal Processing, Aisp 2013*, vol. 427, pp. 267-276, 2014.
 - [6] D. Dewerra, A. Hertz, "TABU SEARCH TECHNIQUES - A TUTORIAL AND AN APPLICATION TO NEURAL NETWORKS," *Or Spektrum*, vol. 11, pp. 131-141, 1989.
 - [7] S. Belhaiza, P. Hansen, G. Laporte, "A hybrid variable neighborhood Tabu search heuristic for the vehicle routing problem with multiple time windows," *Computers & Operations Research*, vol. 52, pp. 269-281, Dec 2014.
 - [8] J. P. van der Gaast, C. A. Rietveld, A. F. Gabor, Y. Q. Zhang, "A Tabu Search Algorithm for application placement in computer clustering," *Computers & Operations Research*, vol. 50, pp. 38-46, Oct 2014.
 - [9] T. Kano, T. Nishida, "SURFACE-WATER WAVES," *Comptes Rendus Hebdomadaires Des Seances De L Academie Des Sciences Serie A*, vol. 287, pp. 83-86, 1978.
 - [10] Y. Wang, B. Li, T. Weise, J. Y. Wang, B. Yuan, Q. J. Tian, "Self-adaptive learning based particle swarm optimization," *Information Sciences*, vol. 181, pp. 4515-4538, Oct 2011.
 - [11] J. J. Liang, A. K. Qin, P. N. Suganthan, S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *Ieee Transactions on Evolutionary Computation*, vol. 10, pp. 281-295, Jun 2006.