Chinese Journal of Electronics
Vol.25, No.3, May 2016

Cooperative Co-evolutionary Artificial Bee Colony Algorithm Based on Hierarchical Communication Model*

HU Chunyu[1,2], LIU Hong[1,2] and ZHANG Peng[1,2]

(1. *School of Information Science and Engineering, Shandong Normal University, Jinan 250014, China*)

(2. *Shandong Provincial Key Laboratory for Novel Distributed Computer Software Technology, Jinan 250014, China*)

Abstract — Canonical Artificial bee colony (ABC) algorithm with a single species is insufficient to extend the diversity of solutions and may be trapped into the local optimal solution. This paper proposes a new co-evolutionary ABC algorithm (HABC) based on Hierarchical communication model (HCM). HCM combines advantages of global and local communication pattern. With adjustment strategies on species and groups, HCM can reduce the computational complexity dynamically. Performance tests show that the HABC algorithm exhibit good performance on accuracy, robustness and convergence speed. Compared with ABC and Integrated co-evolution algorithm (IABC), HABC performs better in solving complex multimodal functions.

Key words — Artificial bee colony (ABC) algorithm, Multi-species co-evolution, Hierarchical communication model (HCM), Dynamical adjustment.

I. Introduction

Optimization problems play important roles in reality. There are some traditional methods to solve optimization problems, such as Newton method, the conjugate gradient method, simplex method and heuristics. However, these methods cannot meet requirements for solution accuracy and convergence speed in optimization of complex problems. With the presence of Ant colony optimization (ACO) algorithm in Ref.[1], Swarm intelligence (SI) emerged as a new way to solve optimization problems. The distributed and self-organized features enable it to perform well in function optimization problems. Particle swarm optimization (PSO) algorithm in Ref.[2] and ACO algorithm together constitute the classic Swarm intelligence algorithms. Most studies have commenced around them so far. ABC algorithm in Ref.[3] is a novel Swarm Intelligence algorithm. By simulating the self-organized labor division of employed bees and unemployed bees, ABC algorithm provides a fresh idea for solving complex optimization problems. However, canonical ABC algorithm has some drawbacks in solving complex optimization problems. It is insufficient to extend the diversity of solutions and easy to fall into the local optimal solutions. Therefore, we change the single-species mode and introduce multi-species co-evolution into the canonical ABC algorithm. Different species exchange elite solutions with each other by means of cooperation and competition, so as to motivate the evolution process and make up for limitation in diversity of solutions.

The main contributions of our paper are listed as follows. 1) Present a new communication model which is called Hierarchical communication model (HCM). It integrates the global communication pattern with the local communication pattern, and imposes adjustment strategies on both species and groups. 2) Combine the canonical ABC algorithm with HCM and propose a novel multi-species co-evolution algorithm named HABC. 3) Conduct performance tests on HABC algorithm and related algorithms. The results demonstrate that our method can extend the diversity of solutions and improve accuracy.

II. Related Work

So far, many scholars have launched researches on ABC algorithm, and gained numerous outstanding results. Karaboga and Basturk proposed ABC algorithm that can be applied to solve multi-dimensional numeric problems in Ref.[4]. In Ref.[5], ABC algorithm is applied

*Manuscript Received Sep. 26, 2014; Accepted Apr. 21, 2015. This work is supported by the National Natural Science Foundation of China (No.61472232, No.61272094, No.61202225).

 DOI:10.1049/cje.2016.05.025

to the path planning in crowd simulation. In Ref.[6], a novel binary version of the artificial bee colony algorithm based on genetic operators is proposed to solve the dynamic image clustering and 0–1 knapsack problems. In Ref.[7], the ABC algorithm is combined with PSO algorithm to plan path for crowd simulation. Paper Ref.[8] presented the best-so-far selection in ABC algorithm and applied it in numerical benchmark functions and image registration applications.

Co-evolutionary algorithm (CEA), which is based on modeling of coexistence phenomena of several species, emerged as a very promising area of evolutionary computation. Rosin firstly introduced the co-evolution into genetic algorithm to eliminate the defect of premature convergence existing in the genetic algorithm[9]. Some studies have demonstrated that the introduction of co-evolutionary idea performs significantly advantages over the conventional algorithm[10−13]. In Ref.[14], Bose *et al.* introduced CEA into ABC algorithm to solve the dynamic optimization problem. But this method is not universal as it is constrained for dynamic optimization problem. In Ref.[15], Zhang P. *et al.* presented three versions of Co-ABC, among which IABC combines the advantages of Co-evolution algorithm based on central control (CABC) fast convergence and Co-evolution algorithm based on annular transmission (TABC) high precision. However, the switch between local communication and global communication in IABC must depend on researchers'experience. It will decrease the performance of the algorithm. Thus, we proposed a novel multi-species co-evolutionary algorithm named HABC. It adjusts the species and groups dynamically according to implementation of the algorithm.

III. Co-evolutionary ABC Algorithm Based on HCM

1. ABC algorithm

ABC algorithm draws inspiration from foraging behavior model of honey bee which is proposed by Tereshko. The algorithm flow is described as follows.

Step 1: Initialize solution space, individual position, maxcycle. Set fitness function.

Step 2: Evaluate the fitness value of candidate solutions and sort the solution according to the fitness value.

Step 3: Select the top 50% candidate solutions as marked solution.

Step 4: Calculate the probability that employed foragers are chose.

Step 5: Onlookers follow employed foragers to search new solutions and update marked solution.

Step 6: Scouts exploit new food source. Memorize the best solution found so far.

Step 7: Output the best solution while the iterative condition is met.

2. Cooperative co-evolutionary ABC algorithm based on HCM model

We introduce multi-species co-evolution into ABC algorithm, and propose a novel co-evolution modelhierarchical communication model. This model integrates advantages of global communication with local communication, and adjusts species and groups dynamically. By adjustments on species and groups, the HABC algorithm can extend the diversity of solutions and accelerate the convergence rate.

The HCM communicates in two levels: species level and group level. One group is composed of one or several species. Species, within the same group, communicates with each other in a synchronous way like the island model to accelerate convergence speed. In the island model, each species transmits its employed matrix into a central community, which is used for optimal solutions communication among species. A global communication matrix is selected from the community and broadcasted into each species to impel its own evolution. Communications of optimal solutions in the community will contribute to the acceleration of convergence speed. On the other hand, groups communicate asynchronously with each other like the diffusion model. In the diffusion model, communication exists only between neighbors. This communication pattern can extend the diversity of solutions. These two patterns for exchange of information can push the evolution of species forward in different ways. In HABC algorithm, the number of species and groups can adjust dynamically. When they are the same, HCM is equivalent to the diffusion model. When group's number is only one, HCM is equivalent to the island model. So we can balance diversity of solutions and convergence rate according to implementation of the algorithm. The HCM is described in Fig.1.

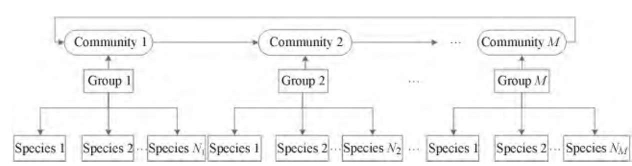


Fig. 1. The Hierarchical communication model (HCM)

In Fig.1, community is employed for communication of optimal solutions. It is set for keeping employed foragers. M represents the number of groups. N_i represents the number of species in group i, where $i \in \{1, 2, \cdots, M\}$. The arrow represents the flow of optimal solutions. In HCM, different species within the same group store their employed foragers in the same community, while adjacent groups exchange their employed foragers stored in their corresponding communities.

1) Formation and extinction of species

The formation and extinction of species are natural evolutionary phenomena. We introduce these phenomena into multi-species co-evolution as dynamical adjustments on species.

a) Species formation

In initialization stage of HABC, there are n species. The algorithm can compute with fast speed as the number of species is small. While the ability for searching optimal solution is limited. To improve the accuracy and diversity of solutions, we add new species dynamically. The process for finding optimal solutions of multimodal functions is promoted by multi-species cooperation.

In HCM, we add a new species if conditions below are all met.

① The number of species is smaller than $3n/2$.

② All species have not contributed to the global optimal solution for at least $g1_{form}$ iterations.

③ The latest generated species has pushed the evolution forward for at least $g2_{form}$ times.

Condition ① is set to limit the number of species, which can prevent the number of particles from increasing infinitely. If conditions ② and ③ are both satisfied, we conclude that the algorithm has not converged to the global optimal solution and the existing species cannot meet the need for function optimization. Thus, new species need to be added to assist the original species with optimization. In this paper, we initialize new species randomly and set up a new group for the new species. The new group is taken into the co-evolution model to accelerate the optimization process.

b) Species extinction

If the fitness of the best individual is not improved after a consecutive generations, the species may be trapped in local optimal solution or have converged to the global optimal solution. If the species is trapped in the local optimal solution, the optimization process stagnates and cannot find the global optimal solution. It needs to execute species extinction to reduce the waste of computing resource.

In HCM, we choose species that meet all conditions below go extinct.

① The number of species is larger than n.

② Species i has not changed its personal best position for g_{kill} generations.

③ The fitness value of ith species personal best position is not better than the fitness value of the global optimal solution.

We perform extinction operation only if all conditions above are met. Condition ① is set to ensure the lower limit of the species number. When only the second condition is satisfied, species may converge to the global optimal solution instead of the local optimal solution. If only the third one is satisfied, the optimization progress may be marching toward the global optimal solution. In this case, extinction operation may prevent the algorithm from finding out the global optimal solution.

2) Combination of groups

In HCM, groups are generated dynamically with the formation of new species. With the algorithm carries on, the number of groups increases with the formation of species. Meanwhile, efficiency of the algorithm is reduced as there are so many groups to communicate with their neighbourhoods. Then the combination of groups is required.

At the beginning of the algorithm, we set up a new group for a new species, as well as a community for a new group. In searching stage, different groups may march forward the same optimal solution. When the locations of optimal solutions found by different groups are too close, we take them searching in the same direction. In this case, we merge these groups into one group. Group i represents the ith group, while $pbest^i$ is the location corresponding to the optimal solution found by the ith group. μ is the threshold value. The condition for combination is described in Eq.(1).

$$\|pbest^{i1} - pbest^{i2}\| < u \tag{1}$$

In implementation of the algorithm, the change of groups is shown in Fig.2. According to Fig.2, we can get some conclusions. If each group corresponds to one species, HCM equals to the diffusion model. It performs well in extending the diversity of solutions. When all species belong to one group, HCM turns into the island model which shows good performance in convergence.

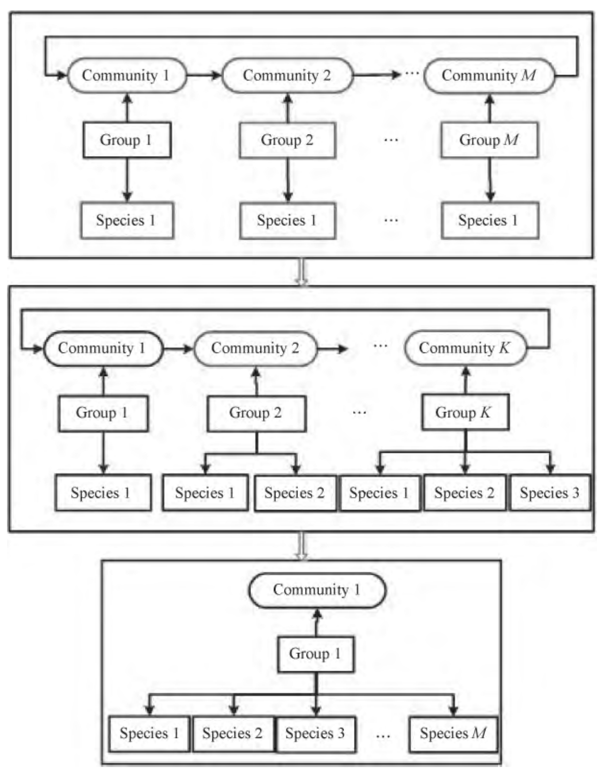


Fig. 2. Dynamic adjustments of groups

By eliminating the weak species and adding new species, we reduce the waste of computing resource and improve the diversity of solutions. Adjustments of groups combine the synchronous pattern with the asynchronous pattern efficiently. With dynamical integration of groups, communication pattern changes reasonably.

3) HABC algorithm

The process of the HABC algorithm based on HCM is described as follows.

Step 1: Initialize parameters such as group size, the maximum iteration number, the threshold of fitness, communication cycle, *etc.*

Step 2: Generate n new species and set up new groups corresponding to them. Perform canonical ABC algorithm.

Step 3: Each species in Group i store its current employed foragers into Community i upon reaching the end of each community cycle.

Step 4: If the number of groups is less than 2, jump to Step 6. Else, Group $i+1$ employs foragers from Community i to participate in its inner evolution and selection at the end of communication cycle. Group 1 imports from Community M as well.

Step 5: Take the foragers coming from the interior of Group $i+1$ and Community i as a whole, and sort them according to their fitness. Select top individuals as employed foragers for Group $i+1$ and broadcast them to species in Group $i+1$ (represents the number of individuals in species i).

Step 6: Determine whether the conditions for adjustments of species and groups are satisfied. If they are satisfied, we adjust the species or groups according to the measures stated above.

Step 7: Species update status according to the selection results of employed foragers, and then run canonical ABC algorithm again.

Step 8: Jump to Step 3 and iterate this process until reaching convergence.

In HABC algorithm, the exchange of elites occurs on both the group level and the species level. In group level, adjacent groups share the optimal solutions in an asynchronous way, which is beneficial to extend the diversity of solutions. In species level, different species within the same group share optimal solutions in a synchronous way, which aims at motivating the evolution of species.

IV. Experiments and Analysis

To make the experiments more reasonable, we control parameters of different algorithms to be the same in the same group of experiments. Both experiments and simulations are made on Lenovo M730E(i3-2120/4GB/500GB).

1. Performance tests

1) Performance tests on five benchmark functions

In experiments below, we set the species size of ABC algorithm as 100 and the number of species as 1. Meanwhile, we set the species size of HABC and IABC as 10, and the number of species as 5 and 10 respectively. The threshold of fitness is set as $limit = 50$. Five benchmark functions are listed below. The dimensions, initialization ranges, optimal solution, and optimal value of each function are listed in Table 1.

① Sphere $$f_1(x) = \sum_{i=1}^{n} x_i^2$$

② Rosenbrock $$f_2(x) = \sum_{i=1}^{n} 100(x_{i+1} - x_i^2) + (x_i - 1)^2$$

③ Griewank $$f_3(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}}) + 1$$

④ Rastrigrin $$f_4(x) = \sum_{i=1}^{n}(x_i^2 - 10\cos(2\pi x_i) + 10)$$

⑤ Ackley $$f_5(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos 2\pi x_i) + 20 + \mathrm{e}$$

Table 1. Parameters of the test functions

Function	Dim	Initial range	Opt. solution	Value	Iteration
Sphere	5	$[-100, 100]^D$	$(0, 0, \cdots, 0)$	0	400
Rosenbrock	50	$[-50, 50]^D$	$(1, 1, \cdots, 1)$	0	3000
Griewank	50	$[-600, 600]^D$	$(0, 0, \cdots, 0)$	0	2000
Rastrigin	50	$[-5.12, 5.12]^D$	$(0, 0, \cdots, 0)$	0	3000
Ackley	50	$[-32.768, 32.768]^D$	$(0, 0, \cdots, 0)$	0	3000

Firstly, we employ Sphere function and Rosenbrock function as benchmark functions to test the efficiency in solving unimodal optimization problems. The value of μ is set as 0.1. Test results are shown in Fig.3.

According to the experimental results, HABC and IABC show small improvements on accuracy, but none obvious advantage in convergence rate compared with canonical ABC algorithm. This can be attributed to the communication pattern of multispecies Co-ABC. The unimodal functions are relatively simple. Therefore, the time saved by the cooperation between species is offset by the cost of communication. The Co-ABC algorithm is unable to exhibit its convergence superiority. On the other hand, there is only one global optimal solution for unimodal function. Thus, Co-ABC cannot exhibit its superiority in extending diversity of solutions, which leads to a small improvement on solution accuracy. Compared with IABC, HABC exhibits small improvements on accuracy and convergence speed. This is due to the dynamical switch between local communication pattern and global communication pattern of HABC algorithm. The switch changes according to implementation of the algorithm, which leads to high efficiency.

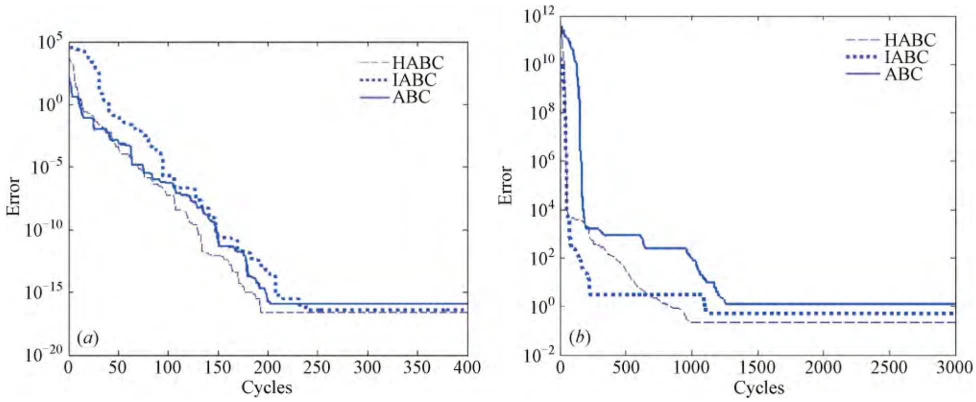


Fig. 3. The test results on unimodal functions. (a) Sphere; (b) Rosenbrock

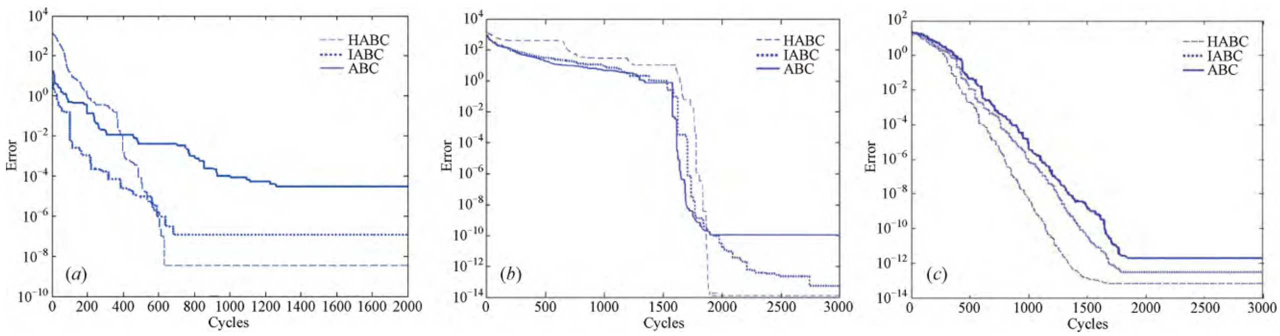


Fig. 4. The test results on multimodal functions. (a) Griewank; (b) Rastrigin; (c) Ackley

Secondly, we take three multimodal functions as benchmark functions to test the performance of HABC. The dimensions of these three functions are all set to 50. Thus, there are many optimal solutions for these three functions and most of these optimal solutions are local extremes. The value of μ is set as 0.01. We can conclude from the test results above that canonical ABC algorithm tend to be trapped in local optimal solutions and converge slowly. By communications between species, Co-ABC algorithm can extend the diversity of solutions. Both HABC and IABC perform well in accuracy and convergence rate in optimization of multimodal functions. On the other hand, HABC execute the formation and extinction of species and combination of groups dynamically according to the implementation of optimization. It can keep local communication and global communication in balance. Therefore, HABC performs better in accuracy and convergence rate than IABC.

Thirty runs of simulations were conducted on five benchmark functions to analyze the performances of the algorithms for function optimization. For HABC and IABC, we set the species size $n = 10$ maximum iteration number $c = 3000$, threshold of fitness $limit = 50$. The species number is set as 5 in HABC and 10 in IABC. For ABC algorithm, there is only one species and the size of species is set to 100. The other parameters are same to HABC and IABC. The test results are presented in Table 2, including the best, worst, mean and standard deviation of function values obtained in 30 runs.

Table 2. The experimental results based on five benchmark functions

Function		HABC	IABC	ABC
f_1	Best	2.52279e–18	1.75373e–18	2.69107e–18
	Worst	1.70703e–17	7.43089e–17	2.10957e–17
	Mean	9.30263e–18	2.63163e–18	1.00737e–17
	Std	4.03923e–18	2.41282e–18	4.60359e–18
	Time(s)	1.63	2.12	1.70
f_2	Best	0.0019888	0.0426744	0.0651929
	Worst	1.05032	1.47186	1.58339
	Mean	0.136007	0.438414	0.52598
	Std	0.235626	0.355936	0.428965
	Time(s)	9.95	10.41	10.83
f_3	Best	6.77236e–10	2.66454e–9	5.44009e–6
	Worst	2.99538e–8	2.31204e–6	7.85171e–5
	Mean	5.27356e–9	4.492e–7	1.22273e–5
	Std	5.8039e–9	6.50017e–7	5.93561e–5
	Time(s)	9.15	10.74	19.56
f_4	Best	8.41283e–15	1.36424e–15	9.54969e–12
	Worst	6.78353e–13	1.58021e–12	6.5413e–8
	Mean	4.89139e–14	8.05704e–13	3.28548e–9
	Std	1.32911e–13	2.88743e–13	1.19966e–8
	Time(s)	6.75	6.86	10.26
f_5	Best	2.21156e–15	2.14051e–13	1.61382e–12
	Worst	1.09512e–13	6.61693e–13	1.33555e–11
	Mean	4.52675e–14	3.97371e–13	6.42467e–12
	Std	1.79915e–14	1.05099e–13	2.47076e–12
	Time(s)	14.20	15.12	15.34

We can conclude from Table 2 that HABC performs better than IABC and canonical ABC in function optimization. The best, worst and mean values of HABC algorithm are closer to the global optimal solution. This demonstrates the advantage of HABC algorithm in accuracy. In addition, standard deviation of HABC algorithm is minimal, which highlight the robustness of HABC algorithm in function optimization. This can be attributed to the dynamical adjustments in HCM. In different runs, the HABC algorithm can adjust the species and groups according to the implementation, which can reduce the difference generated by the randomness of initialization. Thus, HABC performs better in convergence time.

We can draw some conclusions from above experiments.

① In optimization of unimodal function, the time saved by cooperation between species is offset by the cost of communication. Therefore HABC don't perform better than ABC.

② In optimization of multimodal function, HABC exhibits better performance in both accuracy and convergence rate compared with canonical ABC algorithm and IABC algorithm. The formation and extinction of species and dynamical combination of groups contribute to the reduction of time cost of HABC.

③ The hierarchical communication model with self-adaptive adjustment strategies makes the algorithm more robust. Due to the self-adjustments in implementation of algorithm, different runs gain very similar results.

2) Comparison with other state-of-the-art algorithm

In this section, we make comparisons and analysis on the HABC algorithm, the self-adaptive DE (Differential evolution) algorithm proposed in Ref.[16], and the PABC (Powell artificial bee colony) algorithm proposed in Ref.[17]. We use Schwefel function and Alpine function as test functions. Two test functions are described as follows.

$$f_6(x) = 418.98288727243369 \times n - \sum_{i=1}^{n} x_i \sin(\sqrt{|x_i|})$$

$$f_7(x) = \sum_{i=1}^{n} |x_i \cdot \sin(x_i) + 0.1 \cdot x_i|$$

We set the search range of f_6 as (-500, 500) and the search range of f_7 as (-10, 10). For the self-adaptive DE algorithm, we initialize population size as 100, $F = 0.5$, $CR = 0.9$. For PABC algorithm, we set species size $p = 100$, $limit = 200$. For HABC algorithm, we set species size $p = 10$, species number $n = 5$, $limit = 200$, $\mu = 0.01$. The maximum iteration times is set as 5000. Test results are listed in Table 3.

According to the test results in Table 3, the HABC performs better than the self-adaptive DE and the PABC in most cases. In optimization of multimodal functions, the cooperative evolutionary method can lead to higher accuracy and stronger robustness.

Table 3. Comparison with other state-of-the-art algorithms

		Best	Mean	Std	Worst
f_6 (d=30)	Self-adaptive DE	1.324e–11	1.753e–10	1.814e–10	4.999e–10
	PABC	0	0	0	0
	HABC	0	0	0	0
f_6 (d=60)	Self-adaptive DE	1.1634e–09	3.415e–08	2.4563e–08	4.9377e–07
	PABC	6.142e–13	3.15e–12	5.38e–11	2.456e–11
	HABC	4.325e–14	4.886e–13	6.1478e–14	5.178e–12
f_7 (d=30)	Self-adaptive DE	1.648e–11	5.3412e–10	8.456e–10	2.45e–10
	PABC	1.612e–15	2.458e–15	1.8123e–15	3.1536e–15
	HABC	2.145e–17	7.4612e–17	2.654e–16	3.423e–16

2. Performance analysis

1) Complexity analysis

The time complexity of HABC algorithm is mainly determined by the operation of selecting employed foragers. Thus, we take the cost of sorting operations as the major part of time complexity calculation. Quick sorting algorithm is adopted in this paper to select the employed foragers. To simplify the computation, we make assumptions as follows. In HABC, n species are divided into K groups on average, each group has M species and each species has N individuals. We know that the time complexity of sorting operations within one species is $O(N \log_2 N)$. The total time complexity for HABC algorithm is $O(nN \log_2 N + \frac{KMN}{2} \log_2(\frac{MN}{2}))$, where $n = K \times M$. Then we can simplify the total time complexity as $O(nN \log_2 N + \frac{nN}{2}(\log_2(MN) - 1))$. For IABC, it uses local communication pattern in the early period and global communication pattern in the later period. Thus, its time complexity is $O(nN \log_2 N)$ in the first half and $O(nN \log_2 N + \frac{nN}{2}(\log_2(MN) - 1))$ in the second half. The space complexity of IABC is $O(Nn)$ for the early period and $O(\frac{N}{2}(n+1))$ for the later period.

Compared with IABC, the initial species number of HABC is half of IABC. Besides, HABC can adjust the implementation of the algorithm according to the need of optimization, which will lead to the change of M. Therefore, it is difficult to make exactly comparison on HABC and IABC. Generally speaking, the time complexity and space complexity of HABC and IABC are basically in the same magnitude. However, we usually set the initialization of HABC as half of IABC. This configuration and dynamic adjustment strategies of HABC can reduce the computational complexity dynamically according to the optimization process. Thus, the ideas of hierarchical communication model and the HABC algorithm can improve the quality of solutions without increasing complexity or even reducing complexity. In Table 2, we list the time cost for convergence separately. By analysis, we conclude that HABC and IABC consume less time to convergence. HABC has a little advantage in time cost than IBAC.

2) Accuracy analysis

As mentioned above, the essential difference between HABC and ABC is the communication pattern of the employed matrix, which will contribute to the exploration and exploitation process. In HCM, the diffusion model is used to propagate the employed matrix between adjacent groups. It is beneficial to the exploration of the optimal solution. On the other hand, the island model is adopted to accelerate the communication within one group, which contributes to improve the exploitation ability of the algorithm. Compared with IABC, the adjustment strategies enable HABC to adjust the species and groups according to the need of optimization. The switch between island model and diffusion model is more flexible. Thus, HABC performs better than ABC and IABC in solution accuracy. Experiment results in Table 2 show the advantage of HABC in accuracy.

V. Conclusion

In this paper, we present a new communication model—HCM. It combines the local communication pattern with the global communication pattern efficiently by the mechanism of formation and extinction of species and combination of groups. Based on HCM, we propose a new co-evolutionary algorithm called HABC. Adjustments of solution space and species size are also introduced into HABC algorithm to reduce the computational cost. Performance tests show that HABC algorithm performs well in accuracy and convergence speed in solving optimization problems.

References

[1] M. Dorigo, V. Maniezzo and A. Colorni, "The Ant System: Optimization by a colony of cooperating agents", *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, Vol.26, No.1, pp.29–41, 1996.

[2] J. Kennedy and R.C. Eberhart, "Particle swarm optimization", *Proceedings of IEEE International Conference on Neural Networks*, pp.1942–1948, 1995.

[3] D. Karaboga, "An idea based on honey bee swarm for numerical optimization", *Technical report-TR06*, Erciyes University, 2005.

[4] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm", *Journal of Global Optimization*, Vol.39, No.3, pp.459–471, 2007.

[5] P. Zhang, H. Liu and Y. Ding, "Crowd simulation based on constrained and controlled group formation", *The Visual Computer*, Vol.31, No.1, pp.5–18, 2015.

[6] C. Ozturk, E. Hancer and D. Karaboga, "A novel binary artificial bee colony algorithm based on genetic operators", *Information Sciences*, pp.154–170, 2015.

[7] H. Liu, Y. Sun and Y. Li, "Modeling and path generation approaches for crowd simulation based on computational intelligence", *Chinese Journal of Electronics*, Vol.21, No.4, pp.636–641, 2012.

[8] A. Banharnsakun, T. Achalakul and B. Sirinaovakul, "The best-so-far selection in artificial bee colony algorithm", *Applied Soft Computing*, Vol.11, No.2, pp.2888–2901, 2011.

[9] C.D. Rosin, *et al.*, "New methods for competitive coevolution", *Evolutionary Computation*, Vol.5, No.1, pp.1–29, 1997.

[10] F. Xiao, J. Wang, L.J. Sun, *et al.*, "Coverage enhancement strategy based on novel perception and coevolution for multimedia sensor networks", *Chinese Journal of Electronics*, Vol.22, No.1, pp.135–140, 2013.

[11] W. Zhao, S. Alam and H.A. Abbass, "MOCCA-II: A multi-objective co-operative co-evolutionary algorithm", *Applied Soft Computing*, Vol.23, No.5, pp.407–416, 2014.

[12] M.N. Omidvar, X. Li, Y. Mei, *et al.*, "Cooperative co-evolution with differential grouping for large scale optimization", *IEEE Transactions on Evolutionary Computation*, Vol.18, No.3, pp.378–393, 2014.

[13] W. Ding, J. Wang and Z. Guan, "A novel minimum attribute reduction algorithm based on hierarchical elitist role model combining competitive and cooperative co-evolution", *Chinese Journal of Electronics*, Vol.22, No.4, pp.677–682, 2013.

[14] D. Bose, S. Biswas, *et al.*, "A strategy pool adaptive artificial bee colony algorithm for dynamic environment through multi-population approach", *Swarm, Evolutionary, and Memetic Computing*, Springer Berlin Heidelberg, pp.611–619, 2012.

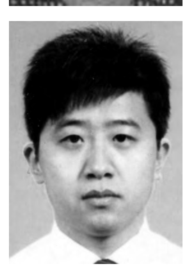
[15] P. Zhang, H. Liu and Y. Ding, "Dynamic bee colony algorithm based on multi-species co-evolution", *Applied Intelligence*, Vol.40, No.3, pp.427–440, 2014.

[16] J. Brest, S. Greiner, B. Boskovic, *et al.*, "Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems", *IEEE Transactions on Evolutionary Computation*, Vol.10, No.6, pp.646–657, 2006.

[17] W. Gao, S. Liu and L. Huang, "A novel artificial bee colony algorithm with Powell's method", *Applied Soft Computing*, Vol.13, No.9, pp.3763–3775, 2013.

HU Chunyu was born in 1990. She received M.S. degree from Shandong Normal University. Now She is a Ph.D. candidate in Institute of Computing Technology, Chinese Academy of Sciences. Her main interests include swarm intelligence and 3D modeling. (Email: huchunyusd@gmail.com)

LIU Hong (corresponding author) was born in 1955, is now a professor of computer science in the School of Information Science and Engineering, Shandong Normal University. She received Ph.D. degree from the Chinese Academy of Sciences in 1998. Her main research interests include computational intelligence and cooperative design. (Email: lhsdcn@126.com)



ZHANG Peng was born in 1989. He received M.S. degree from Shandong Normal University, and now is a Ph.D. candidate in Fudan University. His main research interests include swarm intelligence, social computing. (Email: zhpll@126.com)