Hybrid Bat Algorithm with Artificial Bee Colony

Trong-The Nguyen[1], Jeng-Shyang Pan[2], Thi-Kien Dao[2],
Mu-Yi Kuo[2], and Mong-Fong Horng[2]

[1] Department of Information Technology,
Haiphong Private University, Vietnam
vnthe@hpu.edu.vn
[2] Department of Electronics Engineering,
National Kaohsiung University of Applied Sciences, Taiwan

Abstract. In this paper, a hybrid between Bat algorithm (BA) and Artificial Bee Colony (ABC) with a communication strategy is proposed for solving numerical optimization problems. The several worst individual of Bats in BA will be replaced with the better artificial agents in ABC algorithm after running every *Ri* iterations, and on the contrary, the poorer agents of ABC will be replacing with the better individual of BA. The proposed communication strategy provides the information flow for the bats to communicate in Bat algorithm with the agents in ABC algorithm. Four benchmark functions are used to test the behavior of convergence, the accuracy, and the speed of the proposed method. The results show that the proposed increases the convergence and accuracy more than original BA is up to 78% and original ABC is at 11% on finding the near best solution improvement.

Keywords: Hybrid Bat Algorithm with Artificial Bee Colony, Bat Algorithm, Artificial Bee Colony Algorithm Optimizations, Swarm Intelligence.

1 Introduction

Computational intelligence algorithms have also been successfully used to solve optimization problems in the engineering, the financial, and the management fields for recently years. For example, genetic algorithms (GA) have been successfully various applications including engineering, the financial, the security [1-3], particle swarm optimization (PSO) techniques have successfully been used to forecast the exchange rates, the optimizing, [4-6], to construct the portfolios of stock, human perception [3, 7, 8], ant colony optimization (ACO) techniques have successfully been used to solve the routing problem of networks, the secure watermarking [9, 10], artificial bee colony (ABC) techniques have successfully been used to solve the lot-streaming flow shop scheduling problem [11], cat swarm optimization (CSO) [12] techniques have successfully been used to discover proper positions for information hiding [13]. Communication between two algorithms is taking advantage of the strength points of each type of algorithms. This idea is based on communication strategies in parallel processing for swarm intelligent algorithms. They only exchange information between populations when the communication strategy is triggered. The existing

J.-S. Pan et al. (eds.), *Intelligent Data Analysis and Its Applications*, *Volume 2*,
Advances in Intelligent Systems and Computing 298,
DOI: 10.1007/978-3-319-07773-4_5, © Springer International Publishing Switzerland 2014

methods of these fields are such as such as ant colony system with communication strategies [14], parallel particle swarm optimization algorithm with communication strategies [15], parallel cat swarm optimization [16], Island-model genetic algorithm [17], and parallel genetic algorithm [18]. The parallelized subpopulation of artificial agents increases the accuracy and extends the global search capacity than the original structure. The parallelization strategies simply share the computation load over several processors. The sum of the computation time for all processors can be reduced compared with the single processor works on the same optimum problem. Those algorithms are working only in them self.

In this paper, the concepts of parallel processing and communication strategy are applied to hybrid Bat algorithm with Artificial Bee Colony algorithm is proposed. In the new proposed method, the several poorer individuals in BA algorithm will be replaced with better artificial agents in ABC algorithm after running R_i iterations.

The rest of this paper is organized as follows: a briefly review of ABC and BA is given in session 2; our analysis and designs for the hybrid BA-ABC is presented in session 3; a series of experimental results and the comparison between original BA, original ABC and Hybrid BA-ABC are discussed in session 4; finally, the conclusion is summarized in session 5

2 Related Work

In 2010, Xin-SheYang proposed a new optimization algorithm, namely, Bat Algorithm or original Bat Algorithm (BA), based on swarm intelligence and the inspiration from observing the bats [19]. Original BA simulates parts of the echolocation characteristics of the micro-bat in the simplicity way. It is potentially more powerful than particle swarm optimization and genetic algorithms as well as Harmony search. The primary reason is that BA uses a good combination of major advantages of these algorithms in some way. Moreover, PSO and Harmony search are the special cases of the Bat Algorithm under appropriate simplifications. Three major characteristics of the micro-bat are employed to construct the basic structure of BA. The used approximate and the idealized rules in Xin-SheYang's method are listed as follows:

All bats utilize the echolocation to detect their prey, but not all species of the bat do the same thing. However, the micro-bat, one of species of the bat is a famous example of extensively using the echolocation. Hence, the first characteristic is the echolocation behavior. The second characteristic is the frequency that the micro-bat sends a fixed frequency f_{min} with a variable wavelength λ and the loudness A_0 to search for prey.

1. Bats fly randomly with velocity v_i at position x_i. They can adjust the wavelength (or frequency) of their emitted pulses and adjust the rate of pulse emission $r \in [0, 1]$, depending on the proximity of their target.
2. There are many ways to adjust the loudness. For simplicity, the loudness is assumed to be varied from a positive large A_0 to a minimum constant value, which is denoted by Amin.

In Yang's method, the movement of the virtual bat is simulated by equation (1) – equation (3):

$$f_i = f_{min} + (f_{max} - f_{min}) * \beta \quad 1)$$

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_{best}) * f_i \quad (2)$$

$$x_i^t = x_i^{t-1} + \mathrm{v}_\mathrm{i}^\mathrm{t} \quad (3)$$

where f is the frequency used by the bat seeking for its prey, f_{min} and f_{max}, represent the minimum and maximum value, respectively. The x_i denotes the location of the i^{th} bat in the solution space, v_i represents the velocity of the bat, t indicates the current iteration, β is a random vector, which is drawn from a uniform distribution, and $\beta \in [0, 1]$, and x_{best} indicates the global near best solution found so far over the whole population. In addition, the rate of the pulse emission from the bat is also taken to be one of the roles in the process. The micro-bat emits the echo and adjusts the wave-length depending on the proximity of their target. The pulse emission rate is denoted by the symbol r_i, and $r_i \in [0, 1]$, where the suffix i indicates the i^{th} bat. In every iteration, a random number is generated and is compared with r_i. If the random number is greater than r_i, a local search strategy, namely, random walk, is detonated. A new solution for the bat is generated by equation (4):

$$x_{new} = x_{old} + \varepsilon A^t \quad (4)$$

where ε is a random number and $\varepsilon \in [-1, 1]$, and at represents the average loudness of all bats at the current time step. After updating the positions of the bats, the loudness Ai and the pulse emission rate r_i are also updated only when the global near best solution is updated and the random generated number is smaller than Ai. The update of A_i and r_i are operated by equation (5) and equation (6):

$$A_i^{t+1} = \alpha A_i^t \quad (5)$$

$$r_i^{t+1} = r_i^0[1 - e^{-\gamma t}] \quad (6)$$

where α and γ are constants. In Yang's experiments, $\alpha = \gamma = 0.9$ is used for the simplicity. The process of BA is depicted as follows:

Step1. Initialize the bat population, the pulse rates, the loudness, and define the pulse frequency
Step2. Update the velocities to update the location of the bats, and decide whether detonate the random walk process.
Step3. Rank the bats according to their fitness value, find the current near best solution found so far, and then update the loudness and the emission rate.
Step4. Check the termination condition to decide whether go back to step 2 or end the process and output the result.

In 2005, the Artificial Bee Colony (ABC) algorithm was proposed Karaboga [20], and in 2008, on the performance of ABC was analyzed [21] whose are based on inspecting the behaviors of real bees on finding nectar and sharing the information of food sources to the bees in the nest. There is three kinds of bee was defined in ABC as being the artificial agent known as the employed bee, the onlooker, and the scout. Every kind of the bees plays different and important roles in the optimization process. For example: the employed bee stays on a food source, which represents a spot in the solution space, and provides the coordinate for the onlookers in the hive for reference. The onlooker bee receives the locations of the food sources and selects one of the food sources to gather the nectar. The scout bee moves in the solution space to discover new food sources. The process of ABC optimization is listed as follows:

Step1. Initialization: Spray n_e percentage of the populations into the solution space randomly, and then calculate their fitness values, called the nectar amounts, where n_e represents the ratio of employed bees to the total population. Once these populations are positioned into the solution space, they are called the employed bees. The fitness value of the employed bees is evaluated to take account in their amount of nectar.

$$P_i = \frac{F(\theta_i)}{\sum_{k=1}^{S} F(\theta_k)} \tag{7}$$

Step2. Move the Onlookers: Calculate the probability of selecting a food source by equation (1), where θ_i denotes the position of the i^{th} employed bee, $F(\theta_i)$ denotes the fitness function, S represents the number of employed bees, and P_i is the probability of selecting the i^{th} employed bee. The roulette wheel selection method is used to select a food source to move for every onlooker bees and then determine the nectar amounts of them. The onlookers are moved by equation (2), where x_i denotes the position of the i^{th} onlooker bee, t denotes the iteration number, θ is the randomly chosen employed bee, j represents the dimension of the solution and $\Phi(.)$ produces a series of random variable in the range from -1 to 1.

$$x_{ij}(t+1) = \theta_{ij}(t) + \emptyset(\theta_{ij}(t) - \theta_{kj}(t)) \tag{8}$$

Step3. Update the Best Food Source Found So Far: Memorize the best fitness value and the position, which are found by the bees.

Step4. Move the Scouts: If the fitness values of the employed bees do not be improved by a continuous predetermined number of iterations, which is called "*Limit*", those food sources are abandoned, and these employed bees become the scouts. The scouts are moved by equation (3), where r is a random number and $r \in$ range from 0 to 1.

$$\theta_{ij} = \theta_{jmin} + r \times (\theta_{jmax} - \theta_{jmin}) \tag{9}$$

Step5. Termination Checking: Check if the amount of the iterations satisfies the termination condition. If the termination condition is satisfied, terminate the program and output the results; otherwise go back to Step 2.

3 Hybrid BA with ABC

Hybrid optimization algorithm is structured by communication strategies between two algorithms. This idea is based on replacing the weaker individuals according to fitness evaluation of one algorithm with stronger individuals from other algorithm in parallel processing for swarm intelligent algorithms. Several groups in a parallel structure are created from dividing the population into subpopulations to construct the parallel processing algorithm. Each of the subpopulations evolves independently in regular iterations. They only exchange information between populations when the communication strategy is triggered. It results in taking advantage of the individual strengths of each type of algorithm, replacing the weaker individuals with the better one from other, the reducing of the population size for each population and the benefit of cooperation is achieved.

The hybrid BA-ABC is designed based on original BA optimization and Artificial Bee Colony optimization algorithm. Each algorithm evolves by optimization independently, i.e. the BA has its own bats and near best solution to replace artificial agents of ABC worst and not near best solution. In contrast, the artificial agents better of ABC are to replace the poorer bats of BA after running *Ri* iterations. The total iteration contains R times of communication, where $R = \{R_1, 2R_1, 3R_1, \ldots\}$. The bats in BA don't know the existence of artificial bees of ABC in the solution space.

Let N be the number of population of hybrid BA-ABC, and N_1, N_2 be the number of population of BA and ABC respectively, where $N_1 = N_2 = N/2$. If $t \cap R \neq \varphi$, k agents with the top *k fitness* in N_1 will be copied to N_2 to replace the same number of agents with the worst fitness, where t denotes the current iteration count, R_1 and k are the predefined constants. The diagram of the hybrid BA-ABC with communication strategy is shown in figure 1.

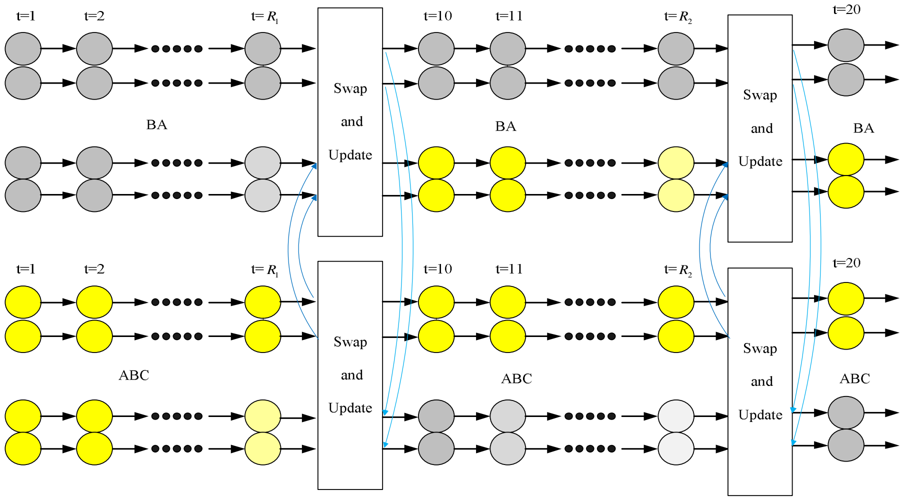

Fig. 1. The diagram of hybrid BA-ABC with a communication strategy

1. **Initialization:** Generate populations for both BA and ABC. Each population is initialized by ABC or by BA independently. Defined the iteration set R for executing the communication strategy. The N_1,N_2 bats and artificial agents S_{ij}^T and X_{ij}^T for populations of BA and ABC respectively, $i = 0, 1, \ldots, N_1 - 1$, $j = 0, 1$, and t is current iteration number. Set $t = 1$.

2. **Evaluation:** Evaluate the value of $f_1(S_{ij}^T)$, $f_2(X_{ij}^T)$ for both BA and ABC in each population. The evolvement of the populations is executed independently by both BA and ABC optimization.

3. **Update:** Update the velocity and the positions of Bat using Eqs. (1), (2) and (3). Update the best Food Source Found So Far: Memorize the best fitness value and the position, which are found by the bees using Eqs. (7),(8) (9).

4. **Communication Strategy:** Migrate the best bats among all the individual of BA's population, copy k individuals with the top *k fitness* in N_1 replace the poorer agents in N_2 of ABC's population and update for each populations in every R_1 iterations. Conversely, migrate the best artificial agents among all the individual of ABC's population, copy k agents with the top *k fitness* in N_2 replace the poorer bats in N_1 and update for each population every R_1 iterations.

5. **Termination:** Repeat step 2 to step 5 until the predefined value of the function is achieved or the maximum number of iterations has been reached. Record the best value of the function $f(S^t)$ and the best bat position among all the bats S^t. Record the best value of the function $f(X^t)$ and the best food Source among all the agents X^t.

4 Experimental Results

This section presents simulation results and compares the hybrid BA-ABC with the original BA, and original ABC, both in terms of solution quality, convergence capability, and the execution time in the number of function evaluations taken. Four benchmark functions are used to test the accuracy and the convergence of hybrid BA-ABC. All the benchmark functions for the experiments are averaged over different random seeds with10 runs.

Let $S = \{s_1, s_2, ..., s_n\}$, $X = \{x_1, x_2, ..., x_n\}$ be the m-dimensional real-value vectors for BA and ABC respectively. The benchmark functions are Griewank, Rastrigin, Rosenbrock and Spherical and listed in equation (10) to equation (13). The goal of the optimization is to minimize the outcome for all benchmarks. The population size of Hybrid BA-ABC, original BA and original ABC are set to 20 for all the algorithms in the experiments. The detail of parameter settings of BA can be found in [19] and setting of ABC can be found in [21].

$$f_1(x) = 1 + \sum_{i=1}^{N} \frac{x_i^2}{4000} + \prod_{i=1}^{N} cos \frac{x_i}{\sqrt{i}} \tag{10}$$

$$f_2(x) = \sum_{i=1}^{N} [10 + x_i^2 - 10cos2\pi x_i \tag{11}$$

$$f_3(x) = \sum_{i=1}^{n-1} (100(x_{i-1} - x_i^2)^2 + (1 - x_i)^2 \tag{12}$$

$$f_4(x) = \sum_{i=1}^{N} x_i^2 \tag{13}$$

The initial range and the total iteration number for all test functions are listed in Table 1.

Table 1. The initial range and the total iteration of test standard functions

Function	**Initial range** $[x_{min}, x_{max}]$	**Total iteration**
$f_1(x)$	[-100, 100]	400
$f_2(x)$	[5.12, 5.12]	400
$f_3(x)$	[-30,30]	400
$f_4(x)$	[-100, 100]	400

The optimization for all of these test functions is to minimize the outcome. The parameters setting for hybrid BA-ABC with original BA side are the initial loudness $A_i^0 = 0.25$, pulse rate $r_i^0 = 0.5$ the total population size N_1 = 10 and the dimension of the solution space $M = 10$, frequency minimum f_{min} = *the lowest of initial range function* and frequency minimum f_{max} = *the highest of initial range function,* and with original ABC side are the initial 'limit'=10 of food source the total population size N_2 = 10 and the dimension of the solution space $M = 10$:. Each benchmark function contains the full iterations of 400 is repeated by different random seeds with 10 runs. The final result is obtained by taking the average of the outcomes from all runs. The results are compared with the original BA and original ABC respectively.

Table 2 compares the quality of performance and time running for numerical problem optimization between Hybrid BC-ABC and original Bat algorithm. It is clearly seen that, almost these cases of testing benchmark functions for Hybrid BA-ABC are better than original BA in terms of convergence. It is special case with test function $f_4(x)$, the Spherical has the mean of value function minimum of total 10 seed runs is 3.18E+07 for Hybrid BA-ABC's performance evaluation, but, for original BA is 2.63E+08, reaches at 78% improvement of convergence. However, all benchmark functions for average time consuming of Hybrid BA-ABC are double time taken in comparison in original BA, for the reasons, the Hybrid algorithm must perform mutation and update operations.

Table 2. The comparison between original BA and hybrid BA-ABC in terms of quality performance evaluation and speed

Function	Performance evaluation		Time running evaluation (seconds)	
	Original BA	***Hybrid BA-ABC***	***Original BA***	***Hybrid BA-ABC***
$f_1(x)$	31.3202	31.0783	0.1657	0.473
$f_2(x)$	4.60E+02	1.39E+03	0.225	0.4318
$f_3(x)$	7.46E-01	3.83E-01	0.1773	0.3897
$f_4(x)$	2.63E+08	3.18E+07	0.1639	0.3491
Average value	**2.63E+08**	**3.18E+07**	**0.7319**	**1.6436**

Figures from 2 to 3 show the experimental results of four benchmark functions in running 10 seeds output with the same iteration of 400 in comparison win original BA.

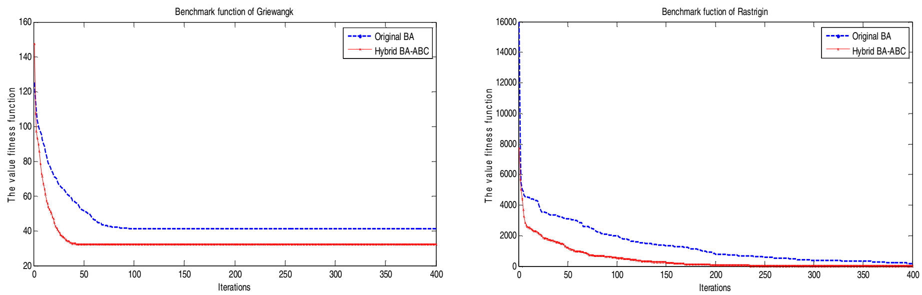

Fig. 2. The mean of function minimum curves in comparing Hybrid BA-ABC and original BA algorithms for function of Griewangk and Rastrigin

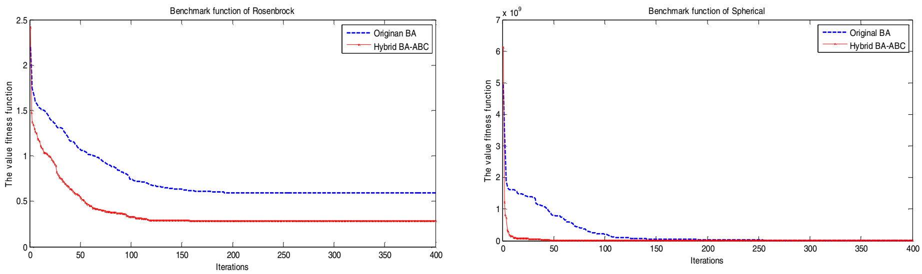

Fig. 3. The mean of function minimum curves in comparing Hybrid BA-ABC and original BA algorithms for function of Rosenbrock and Spherical

Table 3 compares the quality of performance and time running for numerical problem optimization between Hybrid ABC-BA and original ABC. It is clearly seen that, almost these cases of testing benchmark functions for Hybrid ABC-BA are more convergence than original ABC. Average value of all benchmark functions for Hybrid ABC- BA is 8.02E+06 in performance evaluation, but this figure is 9.93E+06 for original ABC, reaches at 11% improvement of convergence. However, average times consuming of all benchmark functions for Hybrid ABC-BA are longer taken than that for original ABC. For this result, the reason is the Hybrid algorithm must perform mutation and update operations.

Table 3. The comparison between hybrid ABC-BA and origianl ABC in terms of quality performance evaluation and speed

Function	Performance evaluation		Time running evaluation (seconds)	
	Original ABC	***Hybrid ABC-BA***	***Original ABC***	***Hybrid ABC-BA***
$f_1(x)$	0.2134	0.078	0.3281	0.4503
$f_2(x)$	2.55E+02	2.04E+02	0.2249	0.4218
$f_3(x)$	9.93E+06	8.02E+06	0.2333	0.3997
$f_4(x)$	8.49E+02	5.94E+02	0.1759	0.3591
Average value	**9.93E+06**	**8.02E+06**	**0.9622**	**1.6309**

Figures from 4 to 5 show the experimental results of four benchmark functions in running 10 seeds output with the same iteration of 400 in comparison win original ABC.

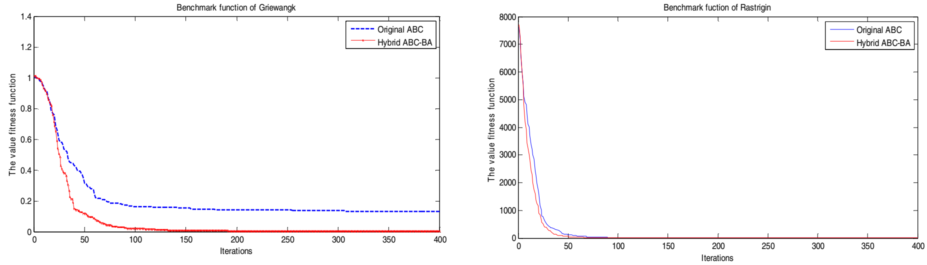

Fig. 4. The mean of function minimum curves in comparing Hybrid ABC-BA and original ABC algorithms for function of Griewangk and Rastrigin

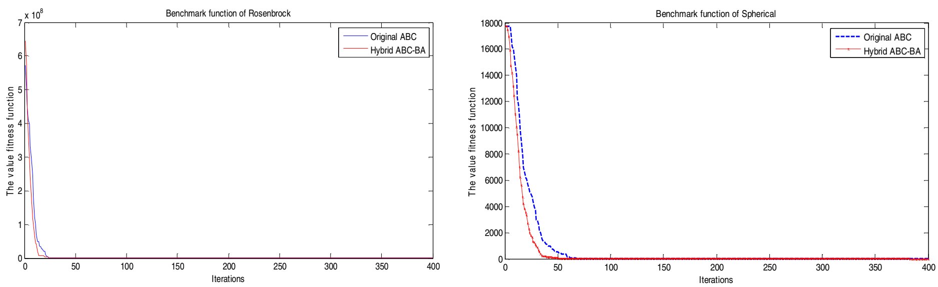

Fig. 5. The mean of function minimum curves in comparing Hybrid ABC-BA and original ABC algorithms for function of Rosenbrock and Spherival

5 Conclusion

This paper, a novel proposed optimization algorithm is presented, namely Hybrid BA-ABC (hybrid Bat algorithm with Artificial Bee Colony). The implementation of hybrid for optimization algorithms could have important significance for taking advantage of the power of each algorithm and achieving cooperation of optimization algorithms. In new proposed algorithm, the several worse individual of Bats in BA are replaced with the better artificial agents in ABC algorithm after running every Ri iterations, and on the contrary, the poorer agents of ABC are replacing with the better individual of BA. The proposed communication strategy provides the information flow for the bats to communicate in Bat algorithm with the agents in ABC algorithm. The performance of Hybrid BA-ABC algorithm is better than both original BA and ABC in terms of convergence and accuracy. The results of proposed algorithm on a set of various test problems show that Hybrid BA-ABC increases the convergence and accuracy more than original BA is up to 78% and original ABC is at 11% on finding the near best solution improvement.

Acknowledgement. The authors would like to express their sincere thanks to the National Science Council, Taiwan (ROC), for financial support under the grants NSC 102-2221-E-151-037-.

References

1. Srinivas, M., Patnaik, L.M.: Genetic Algorithms: A Survey. Computer 27, 17–26 (1994)
2. Wang, S., Yang, B., Niu, X.: A Secure Steganography Method based on Genetic Algorithm. Journal of Information Hiding and Multimedia Signal Processing 1, 8 (2010)
3. Ruiz-Torrubiano, R., Suarez, A.: Hybrid Approaches and Dimensionality Reduction for Portfolio Selection with Cardinality Constraints. IEEE Computational Intelligence Magazine 5(2), 92–107 (2010)
4. Chen, S.-M., Chien, C.-Y.: Solving the traveling salesman problem based on the genetic simulated annealing ant colony system with particle swarm optimization techniques. Expert Systems with Applications 38(12), 14439–14450 (2011)
5. Hsu, C.-H., Shyr, W.-J., Kuo, K.-H.: Optimizing Multiple Interference Cancellations of Linear Phase Array Based on Particle Swarm Optimization. Journal of Information Hiding and Multimedia Signal Processing (4), 292–300 (2010)
6. Chen, S.-M., Kao, P.-Y.: TAIEX forecasting based on fuzzy time series, particle swarm optimization techniques and support vector machines. Information Sciences 247, 62–71 (2013)
7. Jui-Fang, C., Shu-Wei, H.: The Construction of Stock's Portfolios by Using Particle Swarm Optimization, p. 390 (2007)
8. Parag Puranik, P.B., Abraham, A., Palsodkar, P., Deshmukh, A.: Human Perception-based Color Image Segmentation Using Comprehensive Learning Particle Swarm Optimization. Journal of Information Hiding and Multimedia Signal Processing 2(3), 227–235 (2011)
9. Pinto, P.C., Nagele, A., Dejori, M., Runkler, T.A., Sousa, J.M.C.: Using a Local Discovery Ant Algorithm for Bayesian Network Structure Learning. IEEE Transactions on Evolutionary Computation 13(4), 767–779 (2009)

10. Khaled Loukhaoukha, J.-Y.C., Taieb, M.H.: Optimal Image Watermarking Algorithm Based on LWT-SVD via Multi-objective Ant Colony Optimization. Journal of Information Hiding and Multimedia Signal Processing 2(4), 303–319 (2011)
11. Pan, Q.-K., Tasgetiren, M.F., Suganthan, P.N., Chua, T.J.: A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem. Inf. Sci. 181(12), 2455–2468 (2011)
12. Chu, S.-C., Tsai, P.-W.: Computational Intelligence Based on the Behavior of Cats. International Journal of Innovative Computing, Information and Control 3(1), 8 (2006)
13. Wang, Z.-H., Chang, C.-C., Li, M.-C.: Optimizing least-significant-bit substitution using cat swarm optimization strategy. Inf. Sci. 192, 98–108 (2012)
14. Chu, S.-C., Roddick, J.F., Pan, J.-S.: Ant colony system with communication strategies. Information Sciences 167(1-4), 63–76 (2004)
15. Chang, J.F., Chu, S.C., Roddick, J.F., Pan, J.S.: A parallel particle swarm optimization algorithm with communication strategies. Journal of Information Science and Engineering 21(4), 9 (2005)
16. Pei-Wei, T., Jeng-Shyang, P., Shyi-Ming, C., Bin-Yih, L., Szu-Ping, H.: Parallel Cat Swarm Optimization, pp. 3328–3333 (2008)
17. Whitley, D., Rana, S., Heckendorn, R.B.: The Island Model Genetic Algorithm: On Separability, Population Size and Convergence. Journal of Computing and Information Technology 1305/1997, 6 (1998)
18. Abramson, D., Abela, J.: A Parallel Genetic Algorithm for Solving the School Timetabling Problem. Division of Information Technology, pp. 1–11 (1991)
19. Yang, X.-S.: A New Metaheuristic Bat-Inspired Algorithm. In: González, J.R., Pelta, D.A., Cruz, C., Terrazas, G., Krasnogor, N. (eds.) NICSO 2010. SCI, vol. 284, pp. 65–74. Springer, Heidelberg (2010)
20. Karaboga, D.: An Idea based on Honey Bee Swarm for Numerical Optimization. Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department (2005)
21. Karaboga, D., Basturk, B.: On the Performance of Artificial Bee Colony (ABC) Algorithm. Applied Soft Computing 1, 687–697 (2008)