International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.8 (2015), pp.243-252
http://dx.doi.org/10.14257/ijmue.2015.10.8.25

A Novel Hybrid Optimization Algorithm Based on GA and ACO for Solving Complex Problem

Bin Gao, Jing-Hua Zhu and Wen-chang Lang

Wenzhou Key Laboratory of Material Processing and Die & Mould Technology,Wenzhou Vocational & Technical College, Wenzhou 325035, PR China

Abstract

In allusion to the deficiencies of the ant colony optimization algorithm for solving the complex problem, the genetic algorithm is introduced into the ant colony optimization algorithm in order to propose a novel hybrid optimization (NHGACO) algorithm in this paper. In the NHGACO algorithm, the genetic algorithm is used to update the global optimal solution and the ant colony optimization algorithm is used to dynamically balance the global search ability and local search ability in order to improve the convergence speed. Finally, some complex benchmark functions are selected to prove the validity of the proposed NHGACO algorithm. The experiment results show that the proposed NHGACO algorithm can obtain the global optimal solution and avoid the phenomena of the stagnation, and take on the fast convergence and the better robustness.

Keywords: *Genetic algorithm, Ant colony optimization algorithm, Hybrid optimization algorithm, pheromone, benchmark functions*

1. Introduction

There are a great amount of problems, which can be transformed into the combinatorial optimization problems in the industry, national defense, information and other fields. And in the past decades, a lot of intelligent optimization algorithms are used to solve the combinatorial optimization problems. The intelligent optimization algorithm is a heuristic optimization algorithm [1-3], including genetic algorithm (GA), particle swarm optimization (PSO) algorithm, ant colony optimization (ACO) algorithm, Tabu search algorithm, simulated annealing (SA) algorithm, bee colony optimization (BCO) algorithm, and so on. These intelligent optimization algorithms can take on better optimizing and solving ability for he complex problems, but they exits some disadvantages, such as the slow convergence and easy falling into local optimal solution and so on. In allusion to the specific complex problems, a lot of hybrid intelligent algorithms based on combining different intelligent optimization algorithms in order to realize the complementary of the advantages for solving complex optimization problems [4-12]. Xing *et al*. [4] proposed a new algorithm called intelligent genetic algorithm (IGA) for the global optimization of multi-minima functions. The IGA integrates and selects many cross operator, mutation operator and reattempt operation according to the current optimization result. It converges to the global optimization solution without the influence of random searching process. Aslantas *et al*. [5] improved the performance of a fragile watermarking method based on discrete cosine transform (DCT) by using intelligent optimization algorithms (IOA), namely genetic algorithm, differential evolution algorithm, colonial selection algorithm and particle swarm optimization algorithm. Ali *et al*. [6] used particle swarm optimization and genetic algorithm as population-based stochastic search algorithms to optimize the weights and biases of neural network in order to propose an intelligent approach for optimal prediction of gas deviation factor. Vazhayil and

ISSN: 1975-0080 IJMUE

Balasubramanian [7] developed a fast hybrid algorithm, namely, Intelligent Pareto-search Genetic Algorithm (IPGA), which systematically evolves successively efficient frontiers and finally converges to the global Pareto-optimal front. Christian [8] hybridized the solution construction mechanism of ACO with beam search, which is a well-known tree search method (Beam-ACO). The usefulness of Beam-ACO is demonstrated by its application to open shop scheduling (OSS). Chaharsooghi and Kermani [9] proposed a modified version of ant colony optimization (ACO) algorithm for obtaining a set of Pareto solution efficiently. In this algorithm, the efficiency of algorithm is increased by increasing the learning of ants. Twomey *et al*. [10] proposed a study in which we analyze the impact that different communication policies have on the solution quality reached by a parallel homogeneous multi-colony ACO algorithm for the traveling salesman problem. Niknam and Amiri [11] proposed a new hybrid evolutionary algorithm (FAPSO-ACO–K) to solve nonlinear partitional clustering problem. The proposed hybrid evolutionary algorithm combined fuzzy adaptive particle swarm optimization, ACO and k-means algorithms. Elloumi *et al* [12] proposed a novel approach by introducing a PSO, which is modified by the ACO algorithm to improve the performance. The PSO-ACO is validated using the TSP benchmarks and the empirical results considering the completion time and the best length. The hybrid intelligent algorithms can better solve the complex optimization problems, but they exist some disadvantages.

The GA is a highly parallel, random and adaptive search algorithm by referring the mechanism of the natural selection and natural genetic in the living nature. It can search for the optimal target by the population and emphasizes the global search ability. But the GA is inadequate for the local meticulous search ability. The ACO algorithm strengthens the local search ability by the positive feedback of the pheromone. But it is not strong for the global search ability. The GA and ACO algorithm are intelligent optimization algorithms by simulating biological evolution process. In essence, they take on the parallelism, robustness and self-organization, so they do not rely on mathematical model for solving complex optimization problem, and they are to rely on the ability of the population to search for the optimal solution. If the GA and ACO algorithm can be combined to overcome their own shortcomings and avoid the premature phenomenon and the stagnation phenomenon in the collaborative optimization on the maximum extent. So the dynamic characteristics, mechanism, optimization strategies and convergence of the GA and ACO algorithm are analyzed, then a novel hybrid optimization (NHGACO) algorithm is proposed in this paper. Finally, some complex benchmark functions are used to test the performance of the NHGACO algorithm.

2. Genetic Algorithm and Ant Colony Optimization Algorithm

2.1. Genetic Algorithm

The GA [13] is a class of population-based stochastic search technique that solves problems by imitating processes observed during natural evolution. It is based on the principle of the survival and reproduction of the fitness. The GA is a parallel iterative algorithm with certain learning ability, which repeats the evaluation, selection, crossover and mutation operator after initialization until the stopping criteria are reached. In the GA, a population of candidate solution is evolved. Each solution in candidate solutions is encoded as a binary string, called a chromosome. A fitness function is used to evaluate the fitness value of each chromosome. A real-coded GA is a genetic algorithm representation that uses a vector of floating-point numbers instead of 0 and 1 for implementing chromosome encoding. With some modifications of the genetic operators, the real-coded GA has better performance than the binary-coded GA. The crossover operator of a real-coded GA is performed by the borrowing concept of convex combination. The random mutation operator is used to change the gene with one random number. Assuming that we employ GA to search for the largest fitness value with a given fitness function, shown in Figure 1.

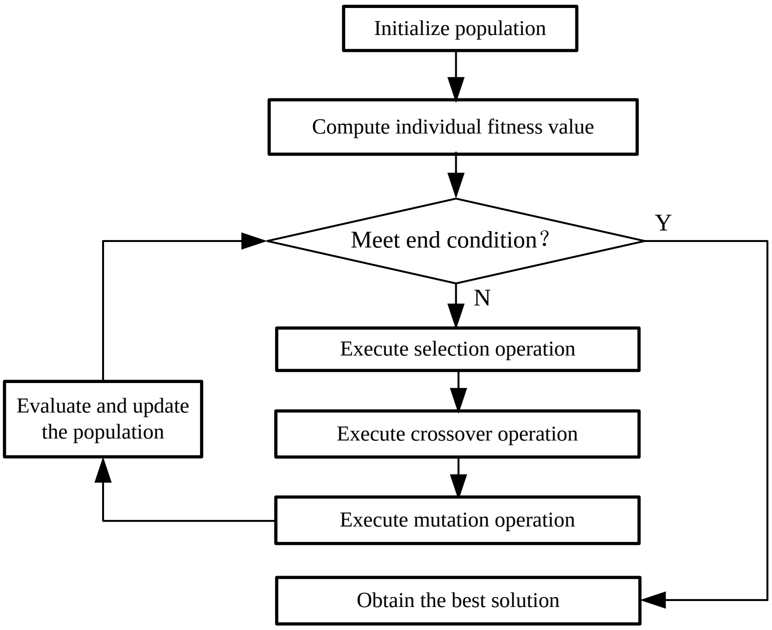

Initialize population
Compute individual fitness value
Meet end condition?
Y
N
Execute selection operation
Execute crossover operation
Execute mutation operation
Evaluate and update the population
Obtain the best solution

Figure 1. Searching Procedure of the GA

2.2. Ant Colony Optimization Algorithm

Ant colony optimization algorithm (ACO) was introduced by Marco Dorigo [14]. It is a branch of newly developed form of artificial intelligence called swarm intelligence, which studies ‘‘the emergent collective intelligence of groups of simple agents”. The ACO is a metaheuristic inspired by the behavior of real ants in their search for the shortest path to food. When ants move, ants will leave a chemical pheromone trail on the ground. The ACO simulates the techniques employed by real ants to rapidly establish the shortest route from a food source to their nest and vice versa (adapting to changes in the environment) without the use of visual information. The basic idea of the ACO is to model the problem that is being solved, just as the search for a minimum cost path in a graph that uses artificial ants to search for good paths. Artificial ants walk through this graph to look for good paths. Since the ants have a rather simple behavior, they can only find poor-quality paths on their own. Better paths are found as the emergent result of the global cooperation among ants.

The ACO consists of a number of iterations of solution construction. In each iteration, a number of ants construct complete solutions by using heuristic information and the collected experiences of the previous groups of ants. These collected experiences are represented by the pheromone trail which is deposited on the constituent elements of a solution. Small quantities are deposited during the construction phase while larger amounts are deposited at the end of each iteration in proportion to solution quality. Pheromone can be deposited on the components and/or the connections used in a solution depending on the problem. In the ACO algorithm, the ACO simulates the optimization of ant foraging behavior. The ACO procedure is illustrated in Figure.2.

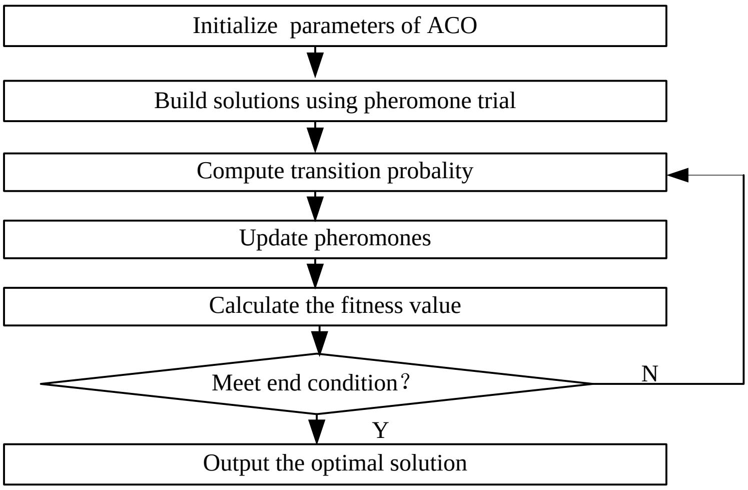

Initialize parameters of ACO
Build solutions using pheromone trial
Compute transition probality
Update pheromones
Calculate the fitness value
Meet end condition?
N
Y
Output the optimal solution

Figure.2 Searching Procedure of the ACO Algorithm

(1) The transition rule

To visit the next city s with the probability $p_k(r,s)$,

$$p_k(r,s) = \begin{cases} \dfrac{\tau(r,s)^{\alpha} \cdot \eta(r,s)^{\beta}}{\sum_{u \in J_r^k} \tau(r,u)^{\alpha} \cdot \eta(r,u)^{\beta}} & if \ s \in J_r^k \\ 0 & otherwise \end{cases} \quad if\ q > q_0 (Bias\ Exploitati\ on) \qquad (1)$$

(2) The pheromone update rule

In order to improve the solution, the pheromone trails must be updated. Trail updating includes local updating and global updating. The local trail updating formula is given by:

$$\tau(r,u) = (1-\rho)\tau(r,s) + \sum_{k=1}^{m} \Delta\tau_k(r,s) \qquad (2)$$

In the formula (6), ρ (0< ρ <1) is the pheromone trial evaporating rate. $\Delta\tau_k(r,s)$ is the amount of pheromone trail added to the edge(r,s) by ant *k* between time *t* and t+Δt in the tour. It is given by:

$$\Delta\tau_k(r,s) = \begin{cases} \dfrac{Q}{L_k} & (\mathrm{r},s) \in \pi_k \\ 0 & otherwise \end{cases}$$

(3)

where Q is a constant parameter, L_k is the distance of the sequence π_k toured by ant inΔ *t*.

3. The Hybrid Optimization Algorithm

3.1. The Idea of Hybrid Optimization Algorithm

The GA applies the idea of natural selection, the survival of the fittest and genetic mutation in the process of biological evolution to optimization problems. It a global random search algorithm. The ACO algorithm takes on the advantages of distributing, self-organizing, positive feedback, and the disadvantages of slow convergence and easy falling into local optimal solution. The GA has the advantages of self-adaptation, parallelism and strong global convergence. But it exists the weak local search ability. Two or more intelligent optimization algorithms are integrated or fused according to certain rules or optimization thought in order to form hybrid intelligent optimization idea. The intelligent algorithm can effectively avoid weaknesses and fully use the advantages of intelligent algorithms in order to greatly improve all performances of the algorithm. So a novel hybrid optimization (NHGACO) algorithm based on GA and ACO algorithm is proposed in this paper. In the proposed NHGACO algorithm, the GA and ACO algorithm are across invoked according to the change rate of the objective function. The solution of ACO algorithm is regarded as the seed of the genetic operation. When the population evolution is close to the stagnation, the ACO algorithm is invoked. The NHGACO algorithm can dynamically control the invoked time of the GA and ACO algorithm. The corresponding elitist strategy is used to cross and pheromone update method is used control the pheromone. And the NHGACO algorithm can find two best chromosomes in the parent generation for crossover operation, in order to increase the possibility for obtaining better solutions and improve the convergence of the proposed NHGACO algorithm.

3.2. The Steps of the NHGACO Algorithm

The description of the NHGACO algorithm is followed:

Step 1. Generate the initial population and initialize parameters of the NHGACO algorithm.

The NHGACO algorithm need initialize these parameters: population sizes (M), the crossover probability P_c and mutation probability P_m,the control parameters α and β, the pheromone trial evaporation rate ρ, stochastic selection threshold (q_0), the maximum iteration times $T_{\max}$, the iteration algebraic counter $t=0$.

Step 2. Initialize the position for each ant.

Step 3. Update the pheromone, and turn to Step.5. Otherwise turn to Step4.

In order to improve the solution, the pheromone trails must be updated. Trail updating includes local updating and global updating.

Step 4. Select the next city with the probability $p_k(r,s)$.

Step 5. The obtained initial solution and global optimal solution are regarded as the initial population of genetic algorithm. Then the initial population is executed the selection operation, crossover operation and mutation operation. Finally, the population is updated.

Step 6. When the GA achieves the minimum number of iterations, the GA or the ACO algorithm is dynamically selected according to the above method. If the ACO algorithm is selected in here, then the pheromone is updated according to the formula (1). Otherwise return to Step 2.

Step 7. When the GA achieves the maximum number of iterations, turn to Step 8. Otherwise the genetic operation is continued to execute.

Step 8. If the number of iterations of the NHGACO algorithm achieves the maximum number, the optimal solution is obtained and the NHGACO algorithm is terminated. Otherwise $t=t+1$, and return Step 5.

4. Experimental Results and Analysis

In order to test the performance of the proposed NHGACO algorithm for solving complex functions, the famous Benchmarks testing functions are used to test the performance. Some typical testing functions are selected from Benchmarks, such as Rosenbrock function, Ackley function, Step function, Schwefel2.21 function, Noisy function and Schaffer function. The global optimum values of the selected testing functions are zero. The proposed NHGACO algorithm is compared with GA, ACO algorithm and CACO algorithm. The parameters of these algorithms are selected after testing. The experiment environments are: Matlab 7.8, the Pentium CPU 2.40GHz, 2.0GB RAM. The experimental parameters are set : population size M = 200, the function dimension is 30, the maximum evolution generation $T_{\max}$ =1000, the crossover probability P_c =0.9, and mutation probability P_m =0,01, the control parameters α =0.5 and β =2.0, the pheromone trial evaporation rate ρ =0.05, stochastic selection threshold q_0 =0.90 , the iteration algebraic counter $t = 0$. Each algorithm is run independently 20 times. The specific expression and variables range of all functions are shown in Table 1.

Table 1. Benchmarks Testing Functions

Function name	Function expression
Rosenbrock	$F_1(x) = \sum_{i=1}^{n} 100(x_i - x_{i-1}^2)^2 + (x_{i-1} - 1)^2, \lvert x_i \rvert \le 30, n = 30$
Ackley	$F_3(x) = -20\exp[\sum_{i=1}^{n} \frac{x_i^2}{n}] - \exp(\sum_{i=1}^{n} \cos(2\pi x_i)/n) + 20 + e, \lvert x_i \rvert \le 32, n = 30$
Step	$F_4(x) = \sum_{i=1}^{n} (\lfloor x_i + 0.5 \rfloor)^2, \lvert x_i \rvert \le 100, n = 30$
Schwefel2.21	$F_5(x) = \max_i \{\lvert x_i \rvert, 1 \le i \le n\}, \lvert x_i \rvert \le 30, n = 30$
Noisy	$F_6(x) = \sum_{i=1}^{n} i x_i^4 + ran[0,1], \lvert x_i \rvert \le 1.28, n = 30$
Schaffer	$F_7(x) = \frac{(\sin\sqrt{x_1^2 + x_2^2})^2 - 0.5}{1 + \frac{(x_1^2 + x_2^2)^2}{1000}} - 0.5, \lvert x_i \rvert \le 100, n = 30$

The performance of the proposed NHGACO algorithm is compared with other published versions of standard ACO algorithm and CACO algorithm. For each test case, all algorithms were run 25 times. The solution quality with the best value and worst value, the achievable accuracy with the mean value and the stability and robustness with the standard deviation value are reported in table as shown in Table 2.

Table 2. The Results for All Benchmarks Functions

Function	Algorithm	Max value	Min value	Mean value	Standard deviation
Rosenbrock	**ACO**	2.863 61E+00	4.823 37E-02	5.623 47E-01	1.236 39E+00
	CACO	4.347 31E-06	1.057 26E-08	5.589 26E-06	4.853 55E-06
	NHGACO	1.946 26E-12	2.471 64E-14	3.438 45E-13	6.104 55E-14

Ackley	**ACO**	4.452 03E+00	1.973 26E-02	2.643 16E-01	4.452 17E-02
	CACO	2.952 13E-08	4.156 10E-10	5.964 15E-09	5.269 85E-08
	NHGACO	2.145 34E-12	2.565 32E-14	8.034 17E-12	1.342 03E-08
Step	**ACO**	4.142 17E-14	1.423 58E-18	3.152 94E-15	2.056 35E-12
	CACO	3.492 68E-18	8.342 05E-21	6.157 45E-19	5.361 31E-18
	NHGACO	2.325 16E-28	4.045 27E-34	6.786 14E-30	1.564 23E-20
Schwefel2.21	**ACO**	4.072 1E-02	8.372 93E-03	9.768 56E-02	4.452 69E-03
	CACO	4.142 12E-05	5.241 46E-08	5.145 19E-06	5.342 86E-05
	NHGACO	1.358 23E-04	6.351 72E-06	4.143 38E-05	2.968 34E-05
Noisy	**ACO**	3.453 15E-02	7.145 36E-04	6.321 06E-03	3.215 00E-04
	CACO	2.234 14E-04	3.006 24E-06	1.032 36E-05	8.056 14E-04
	NHGACO	4.034 17E-06	6.146 18E-08	5.703 45E-07	3.153 46E-06
Schaffer	**ACO**	-4.064 36E-01	-8.341 16E-02	-9.648 32E-01	2.734 42E-02
	CACO	-3.352 24E-02	-6.341 39E-04	-5.263 17E-03	1.239 45E-02
	NHGACO	-5.545 17E-03	-1.956 49E-05	-3.7973 21E-04	5.360 47E-04

As can be seen in Table 2, the proposed NHGACO algorithm can obtain better optimization performance than the ACO algorithm for giving all benchmarks functions and the CACO algorithm for Rosenbrock function, Ackley function, Step function, Noisy function and Schaffer function. And the obtained solutions are very close to the global optima for Step function. On the benchmark test function Schwefel2.21 at 30 dimensions, although the ACO algorithm is better than the proposed NHGACO algorithm, the NHGACO algorithm outperforms the CACO algorithm in standard deviation value. But the proposed HMSDE algorithm is superior to the standard ACO algorithm in all benchmark test functions at 30 dimensions.

5. Conclusion

The ACO algorithm is a stochastic, population-based, evolutionary search algorithm. It is an efficient and powerful optimization algorithm, which widely applied in scientific research and engineering field. The GA applies the idea of natural selection, the survival of the fittest and genetic mutation in the process of biological evolution to optimization problems. The performance of GA depends on the selected mutation, crossover and mutation strategies. In allusion to the premature and stagnation phenomenon of the GA and ACO, the operating characteristics of the GA and ACO are analyzed. A novel hybrid optimization (NHGACO) algorithm based on the GA and ACO algorithm for solving complex optimization problems. In the proposed NHGACO algorithm, the NHGACO algorithm uses the characteristics of the random search, global convergence of the GA and the parallelism, positive feedback mechanism and high solving efficiency of the ACO algorithm to update the global optimal solution. The performance of the proposed NHGACO algorithm is evaluated on set of benchmark problems and is favorably compared with ACO, CACO algorithm in the literature. The results demonstrate that the proposed NHGACO algorithm is overall more effective in obtaining better searching precision, convergence speed, stability, global convergence ability.

References

[1] C. Blum and A. Roli, “Metaheuristics in combinatorial optimization: Overview and conceptual comparison,” Acm Computing Surveys, vol. 35 no. 3, **(2003)**, pp. 268-308.

[2] E. Corchado and A. Abraham, “Hybrid intelligent algorithms and applications,” Information Sciences, vol. 180 no. 14, **(2010)**, pp. 2633-2634.

[3] P. Tarasewich and P. R. McMullen, “Swarm intelligence: Power in numbers,” Communications of the ACM, vol. 45 no. 8, **(2002)**, pp. 62-67.

[4] L. N. Xing, Y. W. Chen and H. P. Cai, “An intelligent genetic algorithm designed for global optimization of multi-minima functions,” Applied Mathematics and Computation, vol. 178 no. 2, **(2006)**, pp. 355-371.

[5] V. Aslantas, S. Ozer and S. Ozturk, “Improving the performance of DCT-based fragile watermarking using intelligent optimization algorithms,” Optics Communications, vol. 282 no. 14, **(2009)**, pp. 2806-2817.

[6] C. Ali, M. Ali and S. Abdolhami, “An intelligent approach for optimal prediction of gas deviation factor using particle swarm optimization and genetic algorithm,” Journal of Natural Gas Science and Engineering, vol. 14 no. 1, **(2013)**, pp. 132-143.

[7] J. P. Vazhayil and R. Balasubramanian, “Optimization of India’s electricity generation portfolio using intelligent Pareto-search genetic algorithm,” International Journal of Electrical Power & Energy Systems, vol. 55 no. 1, **(2014)**, pp. 13-20.

[8] B. Christian, “Beam-ACO—hybridizing ant colony optimization with beam search: an application to open shop scheduling,” Computers & Operations Research, vol. 32 no. 6, **(2005)**, pp. 1565-1591.

[9] S. K. Chaharsooghi and A. H. M. Kermani, “An effective ant colony optimization algorithm (ACO) for multi-objective resource allocation problem (MORAP),” Applied Mathematics and Computation, vol. 200 no. 1, **(2008)**, pp. 167-177.

[10] C. Twomey, T. Stützle, M. Dorigo, M. Manfrin and M. Birattari1, “An analysis of communication policies for homogeneous multi-colony ACO algorithms,” Information Sciences, vol. 180 no. 12, **(2010)**, pp. 2390-2404.

[11] T. Niknam and B. Amiri, “An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis,” Applied Soft Computing, vol. 10 no. 1, **(2010)**, pp. 183-197.

[12] W. Elloumi, H. E. Abeda, A. Abrahama and A. M. Alimia, “A comparative study of the improvement of performance using a PSO modified by ACO applied to TSP,” Applied Soft Computing, vol. 25 no. 12, **(2014)**, pp. 234-241.

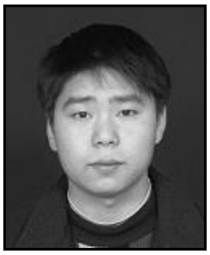
[13] J. H. Holland, “Adaptation in natural and artificial systems,” Ann Arbor: University of Michigan Press, **(1997)**.

[14] M. Dorigo and L. M. Gambardella, Ant colonies for the travelling salesman problem,” Biosystems, vol. 43 no. 2, **(1997)**, pp. 73-81.

Authors

Bin Gao, Associate Researcher, received the Doctor degree in mechanical engineering from Zhejiang University in 2006, Hangzhou, China. The main research directions: optimization design, CAE, physical vapor deposition.

Jing-Hua Zhu, Lecturer, received the Engineering Master degree in computer software from University of Electronic Science and Technology of China in 2011, Chengdu, China. The main research directions: Artificial intelligence，CAD&CG.



Wen-chang Lang, Assistant Researcher, received the Doctor degree in materials processing engineering from Chinese Academy of Sciences in 2010, Shenyang, China. The main research directions: physical vapor deposition.