International Journal of Control and Automation
Vol.8, No.3 (2015), pp.53-66
http://dx.doi.org/10.14257/ijca.2015.8.3.08

Improved Artificial Bee Colony Algorithm and Application in Path Planning of Crowd Animation

Bai Jing and Liu Hong

Information Science and Engineering College Shandong Normal University
Shandong Provincial Key Laboratory for Novel Distributed Computer Software Technology
Bai_sdnu@163.com

Abstract

Crowd animation is a new and continuous challenge in computer animation. In tradition, crowd animation can be realized by key frame technology, and animators should set every character's expression, action, motion, and behavior. Therefore animators' workload by hand will increase tremendously with characters growing, which lead to difficult to realize crowd animation for low efficiency and poor global controllability, especially in path planning by appointing a target position for each individuals. In order to overcome these, an improved artificial bee colony (IM-ABC) algorithm is proposed to apply on the path planning of crowd animation. The IM-ABC is fit to simulate the crowd motion in animation based on the following two merits over the others in crowd animation. One is the rule of role transformation, which can make the rapid convergence of the result and avoid getting trapped in the local optima. The other is the realization of multi-object optimization in the process of iteration, which reaches the uniformly distributed result of swarm motion and especially fits to realize the path plan. In this paper, we simply reviews classical ABC algorithm proposed by Karaboga at the beginning. Then, in order to speed the convergence and make individuals generate paths more realistic and natural, some measures are taken to modify the classical ABC (called IM-ABC) algorithm, which include initializing colony based on chaos sequence, self-adaptively selecting the follower bees, and adaptively controlling parameters, etc. After the experiments of benchmark functions, the results confirm that the IM-ABC have better performance than the classical ABC algorithm and others. Finally, the IM-ABC algorithm is used for path planning to generate the route from the initial to the destination without collision. Through simulation experiments based on four motion models it is showed that this method can succeed generating the optimum paths with efficiency, intelligence, and natural features.

Keywords: *IM-ABC, self-adaptive, crowd animation, motion model, path planning*

1. Introduction

Crowd animation is a new and continuous challenge in computer animation. It can be simply defined as the animation based on the crowd behavior, such as birds flocking, fish schooling, and ants foraging, which can be applied in the electronic game, animation films, advertisement product and teaching demonstration to make the large scale scene. In tradition, crowd animation can be realized by key frame technology, and animators should set every character's expression, action, motion, and behavior. But animators' workload will increase greatly with characters growing, which lead to difficult to realize crowd animation for low efficiency and poor global controllability, especially in path planning by appointing a target position for each individuals. Besides, people appreciating the crowd animation always are more enjoyable to watch the crowd motion as a whole than

ISSN: 2005-4297 IJCA

the irregular and individual motion. Therefore, we propose a novel method of the path planning in crowd animation based on the swarm intelligence.

Swarm intelligence (SI) is one of the fairly new and important research fields in the Artificial Intelligence (AI). It is briefly defined as the collective behavior of the decentralized and self-organized swarms. The term swarm is used for an aggregation of animals like fishes, birds and insects such as ants, termites and bees performing collective behavior. The intelligence of the swarm lies in the behavior that the social organisms perceives their environment, interacts among them ,and then takes action to maximize their chances of success. SI includes a set of colony-inspired computational methodologies and approaches to solve complex problems of the real world applications, such as the ant colony optimization (ACO [1], described by Dorigo *et al.*, 1991), the particle swarm optimization (PSO[2], introduced by Kennedy and Eberhat, 1995) and the artificial bee colony(ABC [3], addressed by Karaboga, 2005). All of methods are based on the natural intelligent swarms (ACO based on ants, PSO on birds, and ABC on bees) and have been studied by many researchers to address the actual complex problems, because the problems have many important counterparts in several engineering areas of the real world. Among so many SI approaches, ABC has attracted highly the interest of the researchers in s. The successful applications of the ABC include the numerical optimization to single-objethat it has the advantages of less parameters and easier to compute and realize than the otherctive or multi-objective problems [4], robotic path planning [5], artificial neural network training [6], data mining [7], wireless network optimization [8], filter design [9], resource scheduling [10], manufacture operation [11] and etc. The application fields include compute, electronics, software, management, control engineer and etc. In this paper, we propose an improved ABC (IM-ABC) algorithm to apply on the path planning of crowd animation based on the following two merits over the others in crowd animation. One is the rule of role transformation, such as employed bees can transform to scout bees while searching for the new nectar source, and onlooker bees can transform to employed bees while employed bees finding more nectar. In the IM-ABC algorithm, the position of a food source stands for a possible solution to the problem and the nectar amount of a food source stands for the quality (fitness) of the associated solution. The role transformation of onlooker bees can make the rapid convergence of the result, and the role transformation of the scout bees can guarantee the avoidance of the local optima. The other merit is the realization of multi-object optimization in the process of iteration, which reaches the uniformly distributed result of swarm motion and especially fits to realize the crowd path planning.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the related research work of crowd animations. In Section 3, the classical ABC algorithm proposed by Karaboga is simply reviewed as the base of our proposed algorithm. In Section 4, an IM-ABC algorithm is proposed based on the chaotic initialization and self-adaptive technology. After the simulation for the benchmark functions the IM-ABC algorithm is confirmed to have the better performance to self-adaptively solve the optimization problem, whatever the problem is a uni-model problem, a multi-model problem, or a difficult constrained problem. In Section 5, the planning path of crowd animation through the IM_ABC algorithm is showed in the fields of crowd aggregation, diffusion, escaping and following, *etc*. Conclusions and future research programs are described in Section 6.

2. Related Work

The study on crowd animations covers a large range of applied areas, including physical-based systems, particle systems, and artificial life systems based on multi-agent [12~16]. Bouvior and Cohen presented a microscopic approach that consists of describing and simulating crowd behavior using a physical-based model [12]. Corradi *et al.* proposed an interacting particle model with macroscopic dynamic behavior [13]. A crowd of pedestrians represented by particles interacting via pair wise forces was considered. Seyfried *et al.*, treated person as self-driven objects moving in a continuous space [14]. Treuille *et al.* proposed a real-time crowd model based on continuum dynamics, simulating a continuous crowd as particles, which is suitable for rendering and establishing a stable and efficient simulation system [15]. Helbing, D., Farkas, I., and Vicsek, T. presented the analysis of a systematic research on crowd evacuation behaviors in panic situations, and established a crowd simulation model for panic conditions [16].

Yang *et al.*, designed a crowd evacuation simulation model based on cellular automata (CA) [17]. The simulation's performance was evaluated with respect to autonomy, cooperation and scalability in a confined space. Chen and Lin proposed a simulation model of crowd behavior based on particle swarm optimization (PSO) algorithm, in which the optimal path from the initial position to the destination was computed by the PSO algorithm [18]. Banarjee *et al.* applied ant colony optimization (ACO) algorithm to evacuation simulation in social emergency [19]. They also built a computational framework of crowd simulation based on the ACO algorithm.

3. ABC Algorithm

Without generality the optimization problem is to find the minimum of *f(X)* as follows

$$\begin{array}{lll} & \min f(X), & X = (x_1, x_2, \ldots, x_n) \in R^n \\ subject\ \ to. & h_k(X) = 0 & k = 1, 2, \ldots, q \\ & g_i(X) \le 0 & i = q+1, q+2, \ldots, m \end{array} \quad (1)$$

Where X∈ F∈ S. The objective function f is defined on the search space S ∈ Rn and the set F ∈ S defines the feasible region by a set of m additional constraints (m ≥ 0). Usually, the search space S is defined as a n-dimensional rectangle in R^n (domains of variables defined by their lower and upper bounds). If the m equates 0, the optimization problem of the objective function f is simplified to unconstrained one.

ABC algorithm was proposed by Karaboga (2005) to simulate the bee colony foraging behavior since the foraging selection leads to the emergence of collective intelligence of bee swarms. In the model at least three essential components are included: food sources, employed foragers and unemployed foragers [3].The amount of food sources are represented by the "profitability" value. The location of food sources are extracted as the solution of the real problem, and the process of bees foraging nectar can be regarded as the search of optima solutions. The employed foragers are associated with a particular food source which they are currently exploiting or are "employed" at. They carry with them information about the particular source to the hive and information can be the distance and direction from the nest. Unemployed foragers are continually at look out for a good source to exploit. There are two types of the unemployed foragers: scouts and onlookers. The former search the environment surrounding the nest for new good sources and the latter wait in the nest and establish a good source through the information shared by employed foragers.

In ABC algorithm two leading modes of the behavior are defined: the recruitment to a rich nectar source and abandonment of an exhausted source. The general algorithmic structure can be concluded as follows.

Step1 Initialization Stage

Produce a randomly distributed initial population ($X_1, X_2, \ldots, X_M$) of M solutions.

Step2 Employed Stage

Employed bees make a search within the neighborhood of the food source after modified on the position (solution) based on the following formula:

$$V^i_j \leftarrow X^i_j + R^i_j(X^i_j - X^k_j)$$
$$X^i \leftarrow (X^i_1, X^i_2, \ldots, V^i_j \ldots, X^i_D) \qquad (2)$$

Employed bees compute the nectar amount (fitness value) of the new position. And if the nectar amount of the new one is higher than that of the previous one, the bee memorizes

Step3 Onlooker Stage

Onlooker bees choose their nectar source according to the information provided by the employed bees. And the selection probability of each site is depended on the following formula:

$$P_i = fit_i / \sum_{i=1}^{M} fit_i \qquad (3)$$

Step4 Scout Stage

If the nectar which the employed bees or follower bees forage for in their position cannot be improved through a predetermined number of trials, called “Limit”, the bees will become scouts and their original position will be abandoned in order to search randomly for new solution.

Step5 Storage and cycle control stage

Store the best solution achieved so far and jump to the step 2 until a termination criterion is met.

In the step1, the ABC generates a randomly distributed initial population *($X_1, X_2, \ldots, X_M$)* of M solutions, where M denotes the size of population. Each solution X_i *($i = 1, 2, \ldots, M$)* is initialized as the following operation

$$X^j_i = X^j_{i\min} + (X^j_{i\max} - X^j_{i\min}) * rand(0,1)$$
$$i = 1, \ldots, M\ ; \quad j = 1, \ldots, N \qquad (4)$$

where *N* is the dimension number of the solutions depending on the parameters of optima problem.

In the step2, every employed bee make a search according to the formula (2) in order to generate a candidate solution from the old one in memory. In the formula (2), $k \in \{1, 2, \ldots, M\}$ and $j \in \{1, 2, \ldots, N\}$ are randomly selected indexes, and $R_{ij} \in [-1, 1]$ is a random number, which control the neighbor scope of X_{ij}. After generating the new solution, its profitability is calculated and a greedy selection is applied between the two solutions.

In the step3, fit_i is the fitness (profitability) value of the X_i which can be computed by the objective function of optimization problem after inputting the value of X_i. Then the onlooker bees become follower bees and probabilistically follow the employed bees in order to select their food sources. During the time, selection technique can be applied, such as the roulette wheel selection method. After selection, the followers make a search as the same with employed bees in step2, and then keep the better solution based on the greedy selection.

In the step4, if the employed bees or follower bees cannot improve the nectar in their position through a number of “Limit”, they will become scouts and their position will be initialized from the beginning according to the formula (4). If not, the operation will go to step5.

In the step5, the ABC stores the best solution, then examine if a termination criteria is satisfied, such as a cycle maximum number or a CPU maximum time. If satisfied, the ABC will terminate, otherwise jump to the step2.

4. Improved Artificial Bee Colony Algorithm (IM-ABC Algorithm)

4.1. Initialization of the ABC based on Chaos Sequence

The initialization of honey bees' population in the classical ABC algorithm is made randomly and cannot be changed during new iterations, which may affect the uniformity of the initial solution and the algorithm performance on convergence speed. To deal with this shortcoming, an initialization method of the ABC based on chaos sequence is proposed. There are some other chaotic methods proposed in the past. Bilal Alatas [20] proposed the approaches introducing chaotic maps with ergodicity, irregularity and the stochastic property in ABC to improve the global convergence by escaping the local solutions. In his paper, a random number each time needed by the classical ABC algorithm is generated by the use of chaotic number. The test result of benchmark functions proved that the method can increase the solution quality. Yudong Zhang *et al.* [21] advocated an adaptive chaotic PSO (ACPSO) method to improve the performance of PSO. The ACPSO trains a two-hidden-layer neural network to construct a crop classification classifier and experiments on Flevoland site show that the proposed ACPSO-NN acquires more efficiently and effectively than BP, ABP, MBP, PSO and RPROP methods. In our article, the initialization of population adopting the method based on chaos sequence includes the following three steps.

Step 1: Generate chaotic variable C j i in (0, 1) iterating the chaotic map as following

$$Logistic\quad Map: \qquad C^j_i = 4C^j_{i-1}(1 - C^j_{i-1}) \tag{5}$$

$$Henon\quad Map: \qquad C^j_i = C^j_{i-1} + 1.2 - (0.5/2\pi)\sin(2\pi C^j_{i-1}) \bmod(1) \tag{6}$$

$$Sinus\quad Map: \qquad C^j_i = C^j_{i-1} + 1.2 - (0.5/2\pi)\sin(2\pi C^j_{i-1}) \bmod(1) \tag{7}$$

where i is the number of the colony member, and j is the dimension.

Step2: Map C^j_i back to the initialization process of the classic ABC:

$$X^j_i = X^j_{i\min} + (X^j_{i\max} - X^j_{i\min}) \times C^j_i \tag{8}$$

Step3: Scouts initialize their position as the same with step2.

4.2. Self-adaptive Method to Select the Follower Bees

Follower bees choose the food source based on the roulette wheel selection method in the classical ABC, but this may lead the diversity to fall during the iteration and premature convergence. Thus various selection pressures should be considered for the varying stage in search of the optimal solution, for example in the early phase the less selection pressure makes the poor individual have the survival chance in order to keep the high diversity, and the more selection pressure the later phase to decrease the searching range and improve the updating speed of the solution [22]. To adjust dynamically the selection pressure during search of the optimal solution, an adaptive choosing method of the follower bees by use of reinforcement learning method and self-control technology is proposed as follow.

Supposed the reinforcement learning state space S = {s_1, s_2,…, s_N}, and the action selection space is A={a_1,a_2,…,a_M}, where N and M stands for the scale of state space and action space. The relative award function of one State-action pair <s_n, a_i> is denoted as $q(s_n, a_i)$ by the formula (10), which means the relative award value choosing the action a_i after adopting the optimal strategy in the of state s_n.

$$P(a_i) = \exp(q(s_n, a_i)/T) / \sum_{j=1}^{M} \exp(q(s_n, a_j)/T) \tag{9}$$

$$q(s_n, a_i) = \frac{fit(s_n, a_i) - fit^{worst}}{fit^{best} - fit^{worst}} \tag{10}$$

$$T = T_0(0.99^{c-1}) \tag{11}$$

In the IM-ABC algorithm, the $q(s_n, a_i)$ shows the relative award after the follower bee selecting an employed bee, and can avoid the impact of super individual making the algorithm converge prematurely. In the formula (9), P(a_i) stands for the selection probability, and the parameter T denoted in the formula (11) comes from simulated annealing methods, whose value will decrease with the iteration number c increasing, which reflects the selection stress will vary with the epoch increasing.

4.3. Adaptive Parameter Control

In the classical ABC algorithm, if the nectar cannot be improved through a predetermined number of trials, called “Limit”, the bees will become scouts and their original position will be abandoned in order to search randomly for new solution. The parameter “Limit” is constant during the whole of algorithm operation. The value of “Limit” is proportioned inversely with the scout production frequency, such as the number of scout production approaching to zero when the value goes to infinity. Generally the adjustment of “Limit” needs some experience. In the Ref. [23], a great deal of numerical tests for benchmark functions have been done to determine the value of “Limit”, and the result shows the moderate values for limit, such as 0.5×n_e×D and n_e×D can improve the search ability of the ABC algorithm, where n_e is the number of employed bees and D is the dimension of the problem. Based on this, in the IM-ABC algorithm the dynamic value of “Limit” is considered for the following

$$\text{Limit}(t) = 0.5 \times [1 + \frac{t}{MCN}] \times n_e \times D \tag{12}$$

Where t is the current number of iteration, and MCN is the maximum iteration number. When 1⩽t⩽MCN, the value of “Limit” varies from 0.5×n_e×D to n_e×D.

4.4. Flow Chart of the IM-ABC Algorithm

Figure 1shows flow chart of the IM-ABC algorithm. At first the initialization of IM-ABC is based on the chaos sequence proposed in 3.1 to produce initial population (X_1, X_2, …, X_M) of M solutions. The first “n_e” solutions are regarded as the position of employed bees after sorting their fitness. Then the employed bees search their neighborhood for better nectar by the formula (8). After choosing the follower bees based on adaptive method, the follower bees search their neighborhood for better nectar as the employed bees done before. Simultaneously the best fitness and the corresponding solution are reserved in the memorizer. The last stage of the IM-ABC algorithm is to check whether the number n_t trapped in the local optimum exceeds the “Limit” threshold value. If exceeding, the bee will transform to a scout bee, jump out from the old position and search for new nectar source; otherwise, the bee colony will continue to forage until a termination criterion is met.

4.5. Simulation of IM-ABC Algorithm

Benchmark functions as Table 1 showed are applied to test the performance of the IM-ABC. The complexity and other properties of these benchmark functions can be easily acquired from their definitions. And the difficulty levels in most benchmark functions are adjustable by setting their arguments. The main properties of four benchmark functions are showed as the Table 1, two of which are Multi-modal functions (having local multi-optima, but only one global optimum) and the rest of which is an uni-modal function (having sole optimum) and a difficult constrained optimization function.

Table 1. Six Benchmark Functions

NO	Function name	Definition	Value range	opt	property
1	Rastrigin	$f_1(x)=\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)+10]$	[-50 50]	0	Multi-modal
2	Griewank	$f_2(x)=\frac{1}{4000}\sum_{i=1}^{n}x_i^2-\prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}})+1$	[-50 50]	0	Multi-modal
3	Rosenbrock	$f_3(x)=\sum_{i=1}^{100}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$	[-30 30]	0	Uni-modal
4	Bump	$f_4(X)=-\left\lvert(\sum_{i=1}^{n}\cos^4(x_i)-2\prod_{i=1}^{n}\cos^2(x_i))/\sqrt{\sum_{i=1}^{n}ix_i^2}\right\rvert$ $s.t.\ \begin{cases}\sum_{i=1}^{n}x_i\le 7.5n\\ \prod_{i=1}^{n}x_i\ge 0.75\\ 0\le x_i\le 10,\quad 1\le i\le 10\end{cases}$	[0 10]	-0.803169	COP

"opt" stands for optimum point. "COP" denotes the constrained optimization problem.]

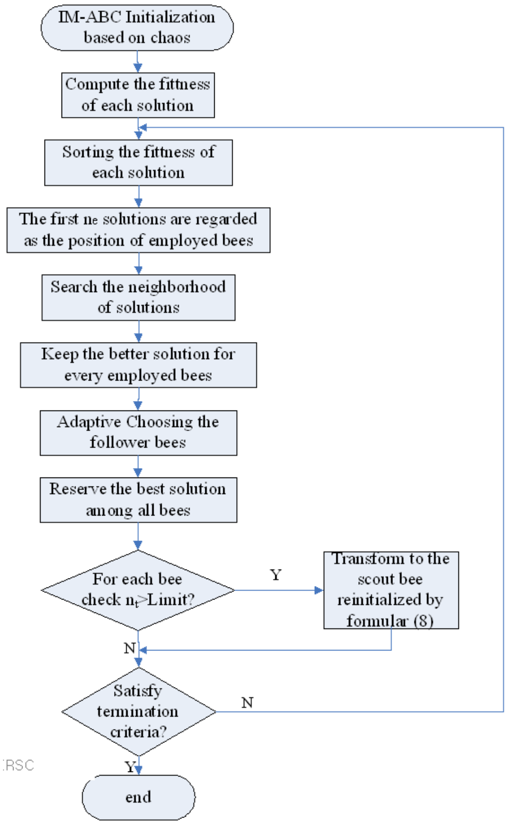

IM-ABC Initialization based on chaos
Compute the fittness of each solution
Sorting the fittness of each solution
The first ne solutions are regarded as the position of employed bees
Search the neighborhood of solutions
Keep the better solution for every employed bees
Adaptive Choosing the follower bees
Reserve the best solution among all bees
For each bee check n_t>Limit?
Y
Transform to the scout bee reinitialized by formular (8)
N
Satisfy termination criteria?
N
Y
end

Figure 1. Flow Chart of the IM-ABC Algorithm

Three algorithms (ABC, PSO, IM-ABC) have been carried out on the experiments in order to compare the behavior of the algorithm introduced to others. In the initialization of IM-ABC, the chaotic sequence is mapped by adopting the formula (5), since the Logistic map has the advantage of good performance and simple realization concluded from the reference [20]. The setting of experimental parameters ABC, PSO, IM-PSO is listed in the following Table 2.

Table 2. Setting of Experimental Parameters

ABC		PSO		IM-ABC	
Popsize	100	Popsize	50	Popsize	100
MCN	2000	MCN	2000	MCN	2000
D	50	D	50	D	50
n_e	50	ω	0.1→0.7	n_e	50
n_o	50	φ_{min}	0	n_o	50
Limit	$n_e \times D$	φ_{max}	2.0	Limit	Adaptive

Popsize, population size; Maxcycle, the maximum number of cycle (iteration); D, the dimension of problem; $n_{e,}$ Number of employed bees; $n_{o,}$ Number of onlooker bees; ω, inertia weight; φ_{min}, φ_{max}, lower and upper bounds of the random velocity rule weight.

Each experiment is repeated 50 times under the same conditions and the average function values of the best solutions found have been recorded at the interval of 50 epochs. As seen from the Y-axis in Figure 2 to Figure 4, the average function values are computed by the logarithms with 10 as a base, and the ones in Figure 5 is normal.

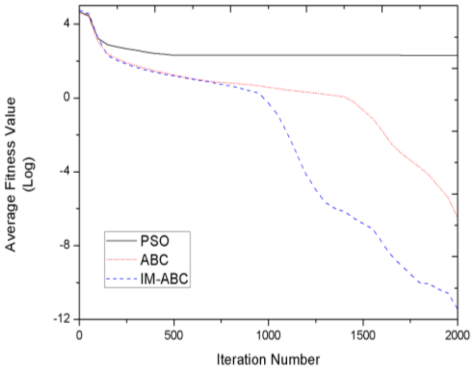


Figure 2. Rastrigin Function

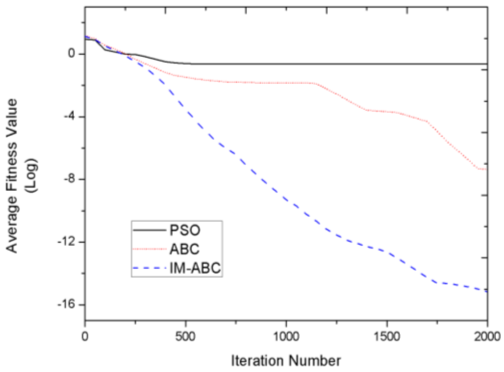


Figure 3. Griewank Function

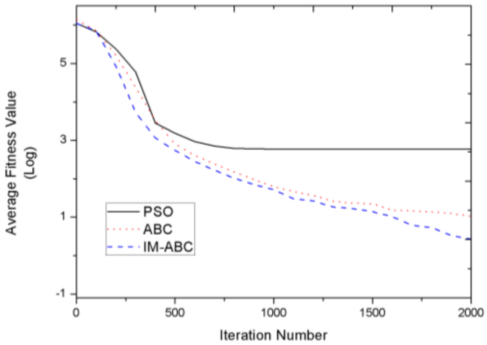


Figure 4. Rosenbrock Function

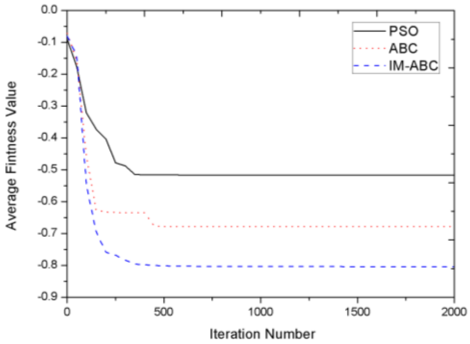


Figure 5. Bump Function

As seen from the Figure 2 and Figure 3, for the multi-modal functions, the PSO algorithm cannot reach the optimal value since trapped in the local optimum, and the ABC algorithm has the slower convergence speed and the worse accuracy compared with the IM-ABC algorithm, although the ABC algorithm can get to the optimum. Especially from the figure 3 the average function values in the IM-ABC algorithm converges to the optimum almost with the linear speed and have the significant difference in the accuracy of results. So it is showed that the IM-ABC algorithm have the performances of good convergence and rapid global search.

As seen from the Figure 4, for the complex uni-modal function, in the early period the three algorithms can converge rapidly. But in the mid- to late period, the PSO algorithm is trapped to the local optimum not to get the global optimization, while the rest two algorithm can continue to converge to the optimum. Because the Rosenbrock function is an extremely complex optimization problem, whose optima value locates in the smooth, long and narrow parabolic belt, the successful chance in the search of optimum is one in a million. The IM-ABC algorithm reaches to the optimum closer than the ABC algorithm.

As for the difficult constrained optimization problem in Table 1, the Bump function has the “three-super” peculiarity including the super nonlinearity, the super multimodality and the super multidimension. Since currently its optimal value is unknown in the world, the best solution is known as -0.803619, when the dimension D equates 20.After introduced by the Simple Penalty Function Method [24] to solve the constrained conditions in the Bump function, as seen for Figure 5, the PSO algorithm only reaches the result of “-0.52187941”, the ABC algorithm reaches the result of “-0.68145015”, and the IM-ABC algorithm can reach the value of “-0.80361928”. The conclusion provides further evidence that the IM-ABC has the better performance and more fit to solve the optimization problem though the problem is very difficult to solve.

5. Application on Path Planning of Crowd Animation

5.1. Model of Crowd Motion

In this section, we apply the IM-ABC algorithm on path planning of crowd simulation based the motion models as follows in order to automatically generate the optimum path from the original position to the own destination.

The model of crowd motion can be divided into four categories according to the features of natural creature: diffusion, aggregation, escaping and following motion. Diffusion can be defined as the motion with the effect of natural, uniform, and random distribution through setting multi-objective and selecting objective freely. For example, the motion that crowd entering the auditorium is belonging to this. Aggregation can be defined as the motion that individuals aggregate at scale through their cooperation with each other, such as the animal commonly forage feed. Escaping is also common in nature and can be defined as the motion for preventing serious damage and harm of life and property, such as from the animal meeting their predators in wilderness to natural or man-made disasters. Following is a classical colony motion, such as fish school following each other to forage, play and *etc*.

We view each individual of crowd as a bee within an N-dimensional space and with meaningful individual properties such as varied velocities, position, *etc*. After the follower bees choosing nectar source in according to the adaptive selection, the bees in the same location comprise a little bee colony, among which collision avoidance and path planning are processed through the information sharing. In order to simulate motion trail in the crowd animation vividly, the distance of individual to a destination, the individual number crowding around the destination and the size of destination are considered in the IM-ABC algorithm.

5.2. Fitness Function

The fitness function is proposed as the following

$$\mathrm{fit}_i = w_1 \cdot dist_i / dist_{\max} + w_2 \cdot \mathrm{count}_i / num_e + w_3 \cdot \mathrm{g}_i / \mathrm{r}_d \qquad (13)$$

where $\mathrm{dist}_i = \sqrt{(x_i - x_o)^2 + (y_i - y_o)^2 + (z_i - z_o)^2}$ denotes the distance of the individual to the destination in the scene of crowd animation, and (x_i, y_i, z_i) , (x_o, y_o, z_o) denote the location coordination of the individual “i" and the destination, respectively; $\mathrm{dist}_{\max}$ denotes the max distance of all individuals to the destination; count_i stands for the individual number crowding around the destination; num_e stands for the total of bees; g_i denotes the size of individual; r_d denotes the radius of destination; w_1~w_3 stands for the distance factor, crowding factor and site size factor, respectively.

The value of the factors w_1~w_3 is very important in the algorithm design, and various with different crowd motion models proposed above. The factor w_1 enables the individual choose the nearer distance from itself to destination; w_2 enables the individual move toward the position of the lesser crowding; w_3 enables the individual number basically proportional to the size of the destination.

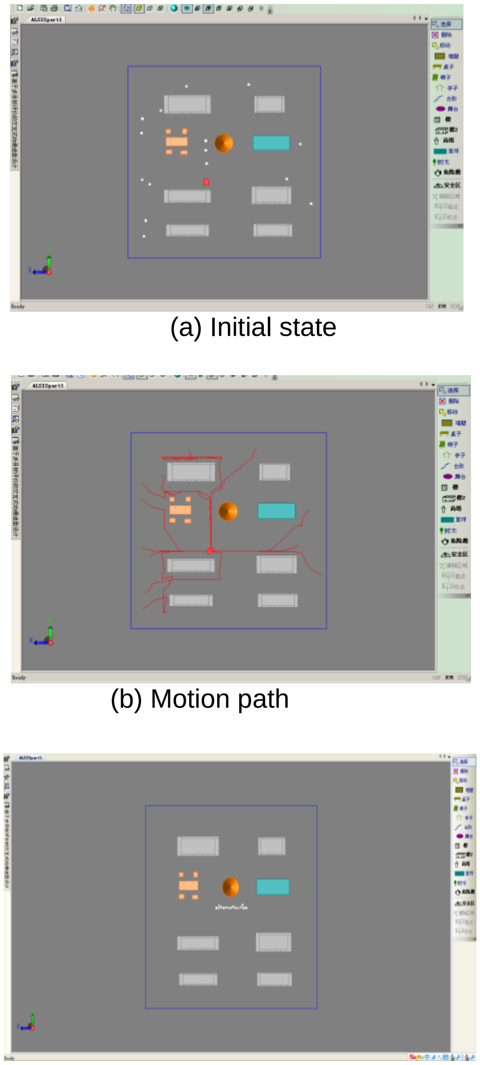
The pseudo code of IM-ABC algorithm on crowd motion based on the above is as follows.

```
Initialization() {
        Population.size          //size of bee colony
        Individual.position     // initial position of bees by chaotic sequence
        Population.iteration    // maximum iteration
        Population.property     // set parameters of property
        Objective.position; }   //position of the motion objective
While(!=Population.iteration || !=Objective.position){
        Individual.fitness()       // compute individual function of fitness showed in (13)
        Individual.role()          // self-adaptively select the role of bee colony
        Employed.search()
        Onlooker.selection ()
        Individual.feasiblity()  //computer the feasibility according to obstacles
        Individual.position_update()  //keep the better individual position
        Save.optimalpath()      // save the global optimal path
        Scout.search()}         // scout bees search for new position if trapped
```

5.3. Simulation Experiments

To demonstrate the good performance of the proposed algorithm, we make microcosmic simulation experiments of four crowd motions on virtual environment by VC++ language. The process of these simulation experiments are showed in Figures 6-Figure 9.

Figure 6 shows the simulation of diffusion motion, whose (a) shows all bees (white particles) in the initial state and the diffusion destinations (blue spots), (c) shows all bees finding their optimal position in end after serial iterations, and (b) shows the path planning in the diffusion motion. Figures 7-9 show respectively the analogous process in aggregation motion, escaping motion, and following motion, where the golden circular cone stands for the pavilion, blue-green rectangle for the lawn, gray rectangles for the buildings, and 4 red little blocks (in Figure 7) for emergency exits.

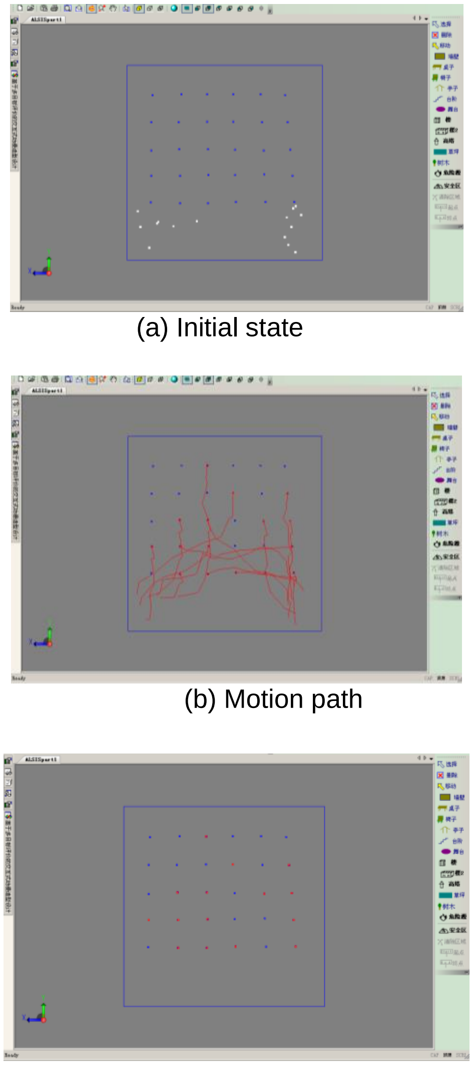


(a) Initial state

(b) Motion path

(c) Terminate state

Figure 6. Simulation of Diffusion Motion

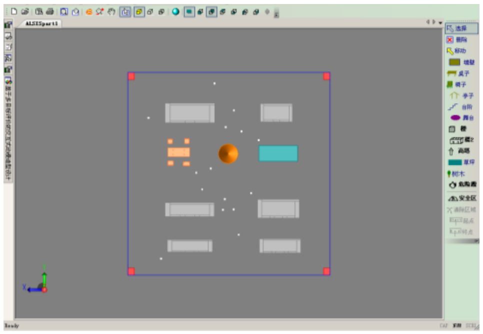


(a) Initial state

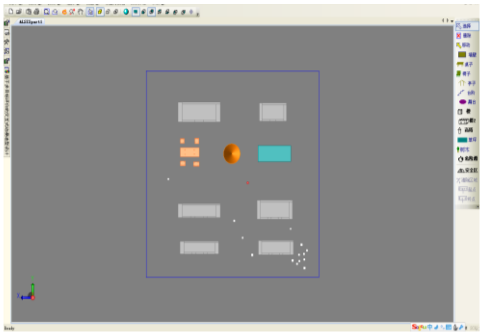
(b) Motion path

(c) Terminate state

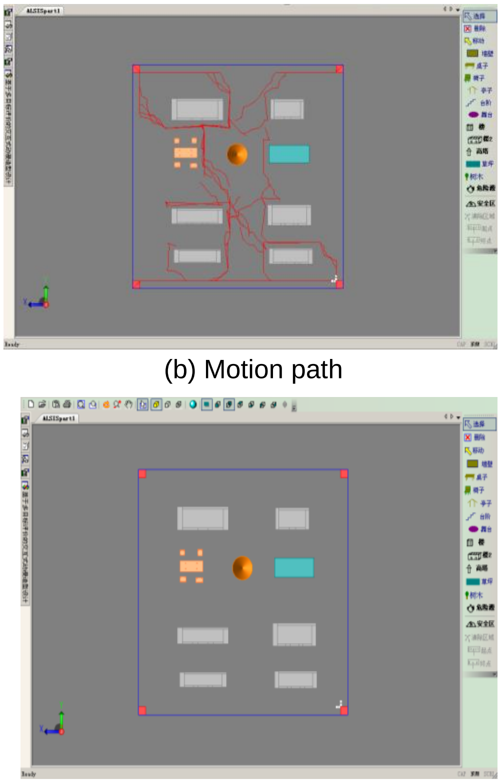
Figure 7. Simulation of Aggregation Motion



(a) Initial state



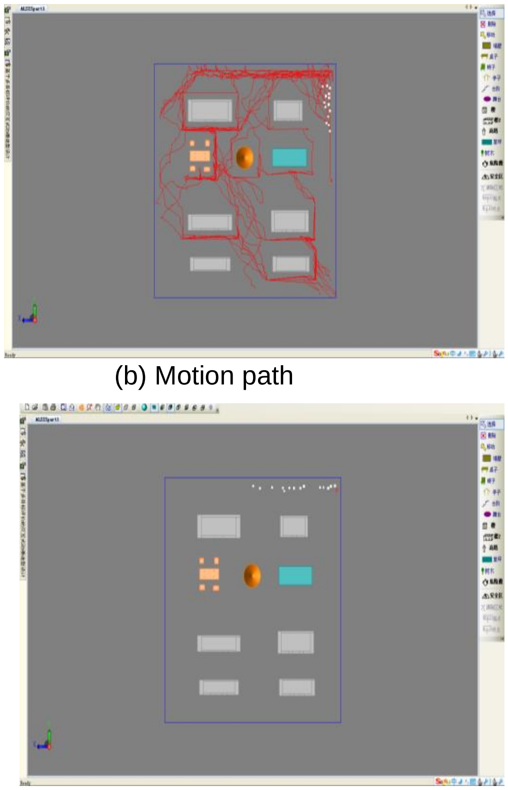
(a) Initial state



(b) Motion path

(c) Terminate state

Figure 7. Simulation of Escaping Motion



(b) Motion path

(c) Terminate state

Figure 8. Simulation of Following Motion

6. Conclusion

In this paper, an improved artificial bee colony algorithm is proposed on chaotic sequence and self-adaptive control parameters. We confirm the good performance of the algorithm through numerous benchmark functions experiment and then make microcosmic simulation experiments to generate the motion path in crowd animation. On one hand, the global convergence is improved by the approach introducing chaotic maps with ergodicity, irregularity and the stochastic property to initialize the employed bees. On the other hand, a self-adaptive choosing method of the follower bees by use of reinforcement learning method and self-control technology adjusts dynamically the selection pressure during search of the optimal solution, and the adaptive value for “Limit” can improve the search ability of scout bees. The path planning of four crowd motions, based on IM-ABC algorithm, improves the system’s intelligence, efficiency and natural features.

Our future research will focus on: (1) make more extensive simulation experiments to test the performance of our proposed algorithm, such as in the condition of multi-objective; (2) study path planning in the more complex environment, such as in the dynamic and random scene of crowd animation; (3) consider the problem how to add the individual features and emotion factors in the crowd motion.

Acknowledgments

This research is supported by: the Natural Science Foundation of China under Grant Nos. 61272094, 61202225, 61303007, 61303157; the Natural Science Foundation of Shandong Province under Grant No. ZR2010QL01

References

[1] M. Dorigo, A. Colorni and V. Maniezzo, "Positive feedback as a search strategy", Technical Report 91-016, Dipartimento di Elettronica, Politecnico di Milano, Milan, Italy (**1991**).

[2] J. Kennedy and R. Eberhart, "Particle swarm optimization", In: IEEE international conference on neural networks, vol. 4, (**1995**), pp. 1942–1948.

[3] D. Karaboga, "An idea based on honey bee swarm for numerical optimization", Technical report. Computer Engineering Department, Engineering Faculty, Erciyes University (**2005**).

[4] R. Hedayatzadeh, B. Hasanizadeh, R. Akbari and K. Ziarati, "A multi-objective artificial bee colony for optimizing multi-objective problems", In: 2010 3rd international conference on advanced computer theory and engineering (ICACTE), vol. 5, (**2010**), pp. V5–277–V5–281.

[5] Z. Hu and M. Zhao, "Research on robot path planning based on ABC algorithm", J. Electric Welding Machine, vol. 39, no. 4, (**2009**), pp. 93 –96.

[6] D. Karaboga and B. Akay, "Artificial bee colony (abc) algorithm on training artificial neural networks", In: 2007 IEEE 15th signal processing and communications applications, vol. 1-3, IEEE, (**2007**), pp. 818–821.

[7] MAM Shukran, YY Chung, WC Yeh, N Wahid and AMA Zaidi, "Artificial bee colony based data mining algorithms for classification tasks", J. Mod Appl. Sci., vol. 5, no. 4, (**2011**), pp. 217–231.

[8] S. Okdem, D. Karaboga and C. Ozturk, "An application of wireless sensor network routing based on artificial bee colony algorithm", In: 2011 IEEE congress on evolutionary computation (CEC), (**2011**), pp. 326–330.

[9] N. Karaboga, E. Uzunhisarcıklı, F. Latifoglu, T. Koza and S. Koçkanat, "Filtering anatomic and electronic noises on mitral valve signal by IIR filters designed with abc (in Turkish)", In: 2011 national electrical-electronics and computer symposium, Elazıg, Turkey, (**2011**), pp. 288–292.

[10] A. Tahooneh and K. Ziarati, "Using artificial bee colony to solve stochastic resource constrained project scheduling problem", In: Tan Y, Shi Y, Chai Y, Wang G (eds) Advances in swarm intelligence. Lecture notes in computer science, vol. 6728. Springer, Berlin, (**2011**), pp. 293–302.

[11] P. Pansuwan, N. Rukwong and P. Pongcharoen, "Identifying optimum artificial bee colony (abc) algorithm's parameters for scheduling the manufacture and assembly of complex products", In: 2010 second international conference on computer and network technology (ICCNT), (**2010**), pp. 339–343.

[12] E. Bouvier and E. Cohen, "Simulation of Human Flow with Particle Systems", Proc. Simulators Int. XII, (**1995**), pp. 349-354.

[13] O. Corradi, P. G. Hjorth and J. Starke, "Equation-free detection and continuation of a Hopf bifurcation point in a particle model of pedestrian flow", J. SIAM Appl. Dynam. Syst., vol. 11, no. 3, (**2012**), pp. 1007-1032.

[14] A. Seyfried, B. Steffen and T. Lippert, "Basics of Modelling the Pedestrian Flow", J. Phys. A, vol. 368, no. 1, (**2006**), pp. 232-238.

[15] A. Treuille, S. Cooper and Z. Popovic, "Continuum Crowds", ACM Trans. Graph., vol. 25, no. 3, (**2006**), pp. 1160- 1168.

[16] D. Helbing, I. Farkas and T. Vicsek, "Simulating dynamical features of escape panic", Lett. Nat. vol. 407, no. 6803, (**2000**), pp. 487–490.

[17] L. Yang, J. Li and D. Zhao, "Evacuation microcosmic disperse model based on individual's behavior", Sci. China Ser. E: Eng. Mater. Sci., vol. 34, no. 11, (**2000**), pp. 487–490.

[18] Y. Chen and Y. Lin, "Controlling the movement of crowds in computer graphics by using the mechanism of particle swarm optimization", J. Appl. Soft Comput., vol. 9, no. 3, (**2009**), pp. 1170–1176.

[19] S. Banarjee, C. Grosan and A. Abraham, "Emotional ant based modeling of crowd dynamics", In: Seventh International Symposium on IEEE Symbolic and Numeric Algorithms for Scientific Computing (**2005**).

[20] B. Alatas, "Chaotic bee colony algorithms for global numerical optimization", J. Expert. Syst. Appl. vol. 37, no. 8, (**2010**), pp. 5682–5687.

[21] Y. Zhang and L. Wu, "Crop Classification by Forward Neural Network with Adaptive Chaotic Particle Swarm Optimization", J. Sensors, vol. 11, (**2011**), pp. 4721-4743, doi:10.3390/s110504721.

[22] H.-j. Ding and Q.-x. Feng, "Artificial Bee Colony Algorithm Based on Boltzmann Selection Policy", J. Computer Engineering and Applications, vol. 45, no. 31, (**2009**), pp. 53-55.

[23] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm", J. Appl. Soft Comput., vol. 8, no. 1, (**2008**), pp. 687–697.

[24] B. Li, X. Yunshi and Q. Wu, "Hybrid algorithm based on particle swarm optimization for solving constrained optimization problems", J. Control and Decision, vol. 19, no. 7, (**2004**), pp. 804-807.

Authors

Bai Jing, She is a PhD student of School of Information Science and Engineering, Shan dong Normal University, Jinan, China. She is also the instructor of that School. Her current research interests include Swarm intelligence, Computer Simulation, Resource Scheduling and Cloud Computing.

Liu Hong, She is currently the president of School of Information Science and Engineering, Shan dong Normal University, Jinan, China. She is also the director of Shandong Provincial Key Laboratory for Distributed Computer Software Novel Technology. She has contributed over 100 articles to professional journals, such as Computer Science and Information Systems, International Journal of Computer Applications in Technology, International Journal of Advances in Engineering Software, etc. Her current research interests include Distributed Artificial Intelligence, Software Engineering and Computer Aided Design