TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.3, March 2014, pp. 2196 ~ 2202
DOI: http://dx.doi.org/10.11591/telkomnika.v12i3.4487

A Dynamic Multi-nest Ant Colony Algorithm for Aircraft Landing Problem

Feng Xiao-rong*, Feng Xing-jie, Liu-dong
School of Computer Science and Technology, Civil Aviation University of China
JInbei Road of the Dongli District, NO.2898, Tianjin, China 086-02224092067
*Corresponding author, e-mail: fengxiaorong@163.com

Abstract

Aircraft landing problem is an NP-hard problem. The article presents a static method for measure of the distance between flights, defines the distance as the pheromone of flights and analyzed experimentally firstly. Then proposes a dynamic multi-nest ant colony optimization algorithm for solving this problem, by dynamically calculates the pheromone between flights. The experimental results show that the algorithm has better global search ability and relatively fast convergence rate and compared with traditional first come first serve, genetic algorithm and particle swarm algorithm, this method can quickly give the better flight approach and landing order to help controllers make efficient aircraft scheduling policy and reduce flight delays.

Keywords: *dynamic, information entropy, ant colony optimization, global search*

1. Introduction

Nowadays, the contradictions between the sharp increase of air traffic demand and the limited airspace resources are getting more and more intense. How to improve the automation level of ALP (Aircraft Landing Problem) is not only an effective way to saves fuel consumption and reduce air pollution, but also is important to reduce airline operating costs and reduce the work pressure of ATC (Air Traffic Control) controllers. Thus it increasingly attracted many scholars' attention.

When the aircraft entered the airport terminal area, the tower will dispatch the aircraft landing sequence and assign a landing time for each aircraft. However, in actual operation, the aircraft landed time must be met some constraints, such as that the airflow which is engendered by the leading flight will generate a great impact on the trailing aircraft, therefore, the time between two aircraft landing requires a specific time interval. ALP has been proven as the TSP, which is a typical multi-constrained NP-hard combinatorial optimization problem [1]. Let's the number of flights landing is N, then the solution space of the ALP ($N!$). In large hub airports, the number of flights increasing, will generate a greater pressure for the ATC controllers and prone to scheduling difficulties. In order to reduce the work pressure ATC controllers, and enhance the automation level of the aircraft landing scheduling, many experts and scholars made extensive research and propose solutions. In small or medium size airports, FCFS (first come first server) is the most simple and common scheduling policies to solve ALP and it able to get the optimal solution even the best solutions, but with the increase in the size of the aircraft, it is difficult to obtain optimum solution, sometimes even no solution. Nowadays, there are two main solutions proposed to solve the ALP. One is the dynamic programming method, such as mixed integer linear programming method [2] suggested by Beasley J E and hybrid simplex algorithm [3] suggested by Ernst A T. In the case of small amount of flights, such algorithms can be faster to get optimal solution, but as the number of flights increasing, the calculation of the algorithm will be increased dramatically and cause the algorithm has high time complexity. Then it can not be applied in the flight landing schedule problem which is a time-sensitive issue. Another is the use of such intelligent bionic algorithm and heuristics algorithm which is improvements from the intelligent bionic algorithm. A theory: aircraft scheduling control based on genetic algorithm suggested by V.H.L.Cheng [4] in 1999 had realized multiple runway ALP problems by using double chromosome coding. In the basement of that, literature [5-7] put

Received August 20, 2013; Revised September 24, 2013; Accepted October 16, 2013

forward a new method: to encode flights、runway or aircraft type respectively and accomplish much more flights partial genetic search is adopted to accomplish the aircraft landing scheduling of multiple flights and runway. Literature [8-10] introduces of RHC (Receding Horizon Control) and the concept of collaborative scheduling policy based on genetic algorithm to solve ALP. These algorithms seek to minimize time cost of the entire flight sequence for the target, but they have a good effect when a small amount of flights. Such algorithms have the following disadvantages: more sensitive to initial values, slow convergence and easy to fall into local optimum. Literature [11] proposes a hybrid immune clonal algorithm HICA (Hybrid Immune Clonal Algorithm), which can optimize the landing time and save the cost of flying under the determinate landing sequence.

ACO (Ant Colony Optimization) algorithm is a new bionic algorithm with strong robustness, distribution and global search capability. It has been applied in some NP-hard problem to get optimum results. This paper presents a static method for measure of the distance between flights, defines the distance as the pheromone of flights and analyzed experimentally firstly. Then proposes a dynamic multi-nest ant colony optimization algorithm (DMACO) for solving this problem by dynamically calculates the pheromone between flights. The experimental results show that the algorithm has better global search ability and relatively fast convergence rate and compared with traditional first come first serve, genetic algorithm and particle swarm algorithm, this method can quickly give the better flight approach and landing order to help controllers make efficient aircraft scheduling policy and reduce flight delays.

The remainder of this paper is organized as follows: Section2 introduces the mathematical model of ALP problem, in Section 3 describes the static method for measure of the distance between flights, and give the ant colony algorithm based on static pheromone for solving ALP. Our DMACO are described in detail in Section 4, this is followed by the experimental description and the corresponding results obtained in Section 5, and in Section 6 the related work conclusions and future work are presented.

2. Mathematical Model of ALP Problem

ALP question designed to find one of the plane's landing orders, the order meet landing minimum interval [14], and make the planes have the minimum total delay time. The mathematical model is described as fallows:

N={1,2,……,n} represents a set of flights to be landed; Vector ETA represents the estimated time of flights, while vector T represents the actual landing time of the flights. ALP to find the best landing scheduling order which has the minimum total delay time and meet the minimum landing interval.

$$Y = \sum_{i=1}^{n} (t_i - eta_i) \tag{1}$$

The Equation (1) is the mathematical model of ALP.

According to the international practice, aircraft can be divided into heavy machine (H), large (L) and small (S); Table1 shows the minimum interval time that different aircraft types must meet.

Table 1. The Requirements for Aircraft Separation

Leading	Trailing		
	Heavy	Large	Small
Heavy	1	1.5	2
Large	1	1.5	1.5
Small	1	1	1

3. Static Ant Colony Optimization Algorithm

3.1. Static Heuristic Description

In the traditional TSP problem the distance between cities can be calculate by the coordinates of the cities, and the distance is symmetric. Such as the distance from city A to city B is L, conversely, the distance from city B to city A also is L, so it can use the distance as the pheromone of cities. But in the ALP, using the time interval between flights as the pheromones must make appropriate changes or process, because this time interval is asymmetric. This is mainly for two reasons; first, the different types of aircraft have different landing time intervals, detailed in Table 1; On the other hand, for instance plane A's ETA is 12, while plane B's ETA is 15. Means that the aircraft A is greater than B at least 3 minutes to reach the airport, when A landed precedes B, the time distance A to B is 3,but when B landed precedes A, the time distance B to A not is 3.

Define 1: Let t_i and t_j represent the ETA of flight *i* and flight *j* respectively, the distance dij from flight *i* to flight *j* is defined as follows:

$$d_{ij} = \begin{cases} \max(t_j - t_i, \theta_{ij}), when(t_i <= t_j) \\ a(t_i - t_j) + b, when(t_i > t_j) \end{cases} \tag{2}$$

While θ_{ij} represent the minimum interval time of flight *i* and flight *j*,a>=1, and b>=0.

Define 2: the heuristic information η_{ij} between flight *j* and flight *i* are defined as follows:

$$\eta_{ij} = 1/d_{ij} \tag{3}$$

3.2. The Main Steps of Algorithm

Step 1: Calculated the heuristic information of flights according to the formula (2) and (3);
Step 2: All ants placed on a flight randomly;
Step 3: Add a not landing flight to the landing sequence with a certain probability;
Step 4: Back to Step3, until all flights have landed;
Step 5: Update pheromone matrix;
Step 6: Back to Step2, until all ants complete their travel;
Step 7: The algorithm reaches the end of the condition, if not then return to Step1;

3.3. Pheromone Matrix

Pheromone τ_{ij} indicates the degree of flight *j* keeping flight *i* landed. Initially, all the flights have the same pheromone, ie $\tau_{ij} = \tau_0, \forall(i, j) \in N$ while τ_0 is a positive integer. After all the ants have built a path, the pheromone update by the following formula:

$$\tau_{ij} = (1-\rho) \bullet \tau_{ij} + \sum_{k=1}^{m} \Delta\tau_{ij}^k \tag{4}$$

While ρ , whose ranges from 0-1,represents the pheromone evaporation rate. Its role is to make the algorithm to avoid the accumulation of pheromone infinite, but also forget poor previous path. $\Delta\tau_{ij}^k$ represents the amount of information of the edge (i,j) released by ant *k*, but also represents the flight *i* landed followed flight *j*. It defined as:

$$\Delta\tau_{ij}^k = \begin{cases} \frac{Q}{L_k}, \textit{flight i landed followed flight j} \\ 0, \textit{otherwise} \end{cases} \tag{5}$$

Q is a positive integer. L_k represents the total delay time of the landing sequence constructed by ant k.

3.4. Flight Selection

The probability of choosing flight *j* following flight *i* to land is calculated as follows:

$$p_{ij}^{k} = \begin{cases} \dfrac{\tau_{ij}^{\alpha} \bullet \eta_{ij}^{\beta}}{\sum_{t \in N_i^k} \tau_{it}^{\alpha} \bullet \eta_{it}^{\beta}}, j \in N_i^k \\ 0, otherwise \end{cases} \quad (6)$$

While N_i^k represents the collection of not landing flights. α and β represent the relative influence of the pheromone and heuristic information respectively. The experiment will analyze them in part 5.

4.Dynamic Multi-nest Ant Colony Optimization Algorithm

4.1 Definition of Heuristic Information

Define 2: Let flight *j* keeping flight *i* landed, the distance d_{ij} between flight *j* and flight *i* are defined as follows:

$$d_{ij} = t_j - t_i \quad (7)$$

d_{ij} represents idle time of the runway during flight *j* keeping flight *i* landed, and need to be dynamically calculated in algorithmic process. While t_i and t_j represent the landing time of flight *i* and flight *j* respectively.

For example, there are 12 flights to be landed, and their ETA and type is:

ETA	12	15	7	6	10	7	15	7	6	9	6	9
type	H	S	H	L	S	H	L	S	S	S	H	L

If an ant is assigned in the third flight, then the distance from it to others is calculated as follows:

d_{31}	d_{32}	d_{34}	d_{35}	d_{36}	d_{37}	d_{38}	d_{39}	$d_{31}0$	d_{311}	d_{312}
5	8	15	3	1	8	2	2	2	1	2

It needs to calculate the dynamic distance between the last landed flight and all no landing flights when performing flight selection.

Define 3: Let flight *j* keeping flight *i* landed, the heuristic information η_{ij} between flight *i* and flight *j* are defined as follows:

$$\eta_{ij} = 1 / d_{ij} \quad (8)$$

4.2. The Main Steps of Algorithm

The main flow algorithm is shown in Figure 1, after figure 1the main steps of the algorithm are given.

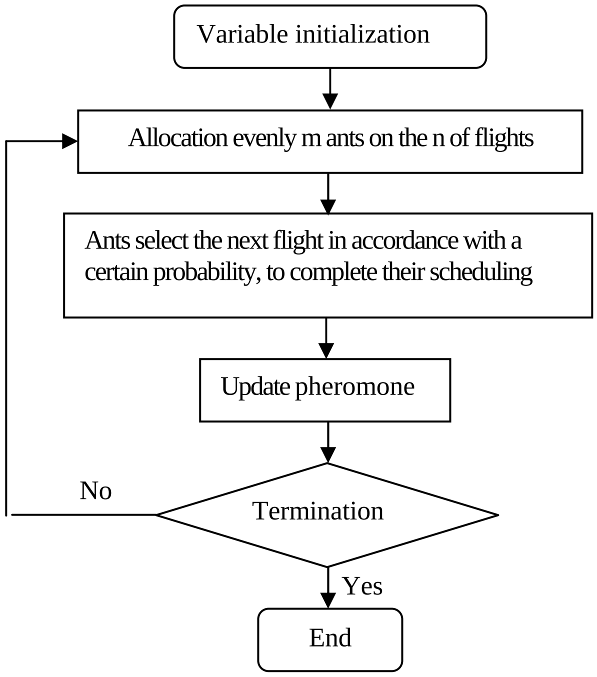


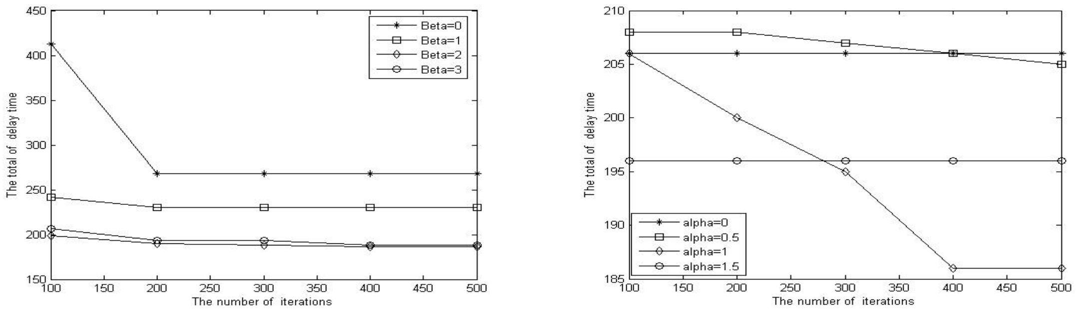
Figure 1. Algorithm Flowchart

Step 1: Ants evenly distributed among *n* flights;
Step 2: Calculating the heuristic information between the last flights of the path and all flights not falling;
Step 3: Add a not landing flight to the landing sequence with a certain probability using Equation (6) to calculated;
Step 4: Back to Step2, until all flights have landed;
Step 5: According to equation (4) and (5) to update the pheromone matrix update;
Step 6: Back to Step1 until all ants complete their travel;
Step 7: The algorithm reaches the end of the condition, if not then return to Step1;

5. Experiment and Analysis

Experimental uses one airport flight landing data, which has 37 flights moment, to test the algorithm. First, the experimental analyzes the parameters α and β, then compare the performance of ant colony algorithm, genetic algorithm, and particle swarm algorithm with other data sets.

5.1. Parameter Analysis



(a) (b)

Figure 2. Parameters Performance Analysis

Set β=2, Q=10, ρ=0.5, then set α values were 0,0.5,1,1.5 to analysis the performance of α. Data results of Figure 2(a) were the mean of 10 times simulation results.

Set α=1, Q=10, ρ=0.5, then set β values were 0,1,2,3 to analysis the performance of β. Data results of Figure 2(b) were the mean of 10 times simulation results.

Analysis from the result of the experiment in Figure 2(a), when α=0, Algorithm does not use pheromones, landing flight time closest *i* flights will be to select the next landing; the algorithm is equivalent to a random greedy algorithm. While when α>1, algorithm quickly stalled, then all the ants are moving follow the same route, and the results have a certain gap with the optimal value. The algorithm can obtain a better solution when α=1. Analysis from the result of the experiment in Figure 2(b), when β=0, or β=1the algorithm quickly stalled, and the results have a certain gap with the optimal value. The algorithm can obtain a better solution when β=2.

5.2. Comparison with other Algorithms

Experiment 1 uses five group of small-scale data sets for testing; the results are shown in Table 1.

Table 1. The Results of Small-scale Data Sets

No.	N	FCFS	GA	PSO	SMACO	DMACO
1	12	43	38	35	34	31
2	12	63	52	46	40	37
3	12	36	28	27	27	22
4	20	72	68	61	62	57
5	37	270	246	230	198	186

Experiment 2 uses three groups of larger data sets for testing; the results are shown in Table 2.

Table 2. The Results of Larger Data Sets

No.	N	FCFS	GA	PSO	SMACO	DMACO
1	100	348	280	264	268	231
2	200	674	613	583	588	582
3	250	835	725	678	723	668

N represents the number of flights. Columns 3-6 represent the result of using FCFS (first come first serve), GA (genetic algorithm), and PSO (particle swarm optimization),SMACO. The last column represent the result of using DMACO when α=1 and β=2. All ants are distributed on all flights evenly in DMACO; therefore, the algorithm has better global search capability than single-nest colony algorithm, moreover DMACO more precise description of the heuristic information, which is calculated by the flight dynamic time interval between flights, the algorithm can have a faster convergence speed than others.

From the experimental results, DMACO algorithm significantly better than FCFS, GA, PSO and SMACO in small-scale data sets. While in large-scale data optimization, DMACO algorithm is slightly better than the PSO algorithm, significantly better than FCFS and GA algorithms.

6. Conclusion

This paper presents a new DMACO algorithm with dynamic computing heuristic information for solving single runway landing flight scheduling problem, the experimental results show that the algorithm converges faster, and can get the optimal solution in a short time. The future will continue to be improved DMACO algorithm to use in solving multi-runway landing flight scheduling problems.

Acknowledgements

This work was supported by the National Natural Science Foundation of China (Grant No. U1233113, 61301245), and the Fundamental Research Funds for the Central Universities of China (Grant No. ZXH2012M005, ZXH2012M004, ZXH2011C010), and Civil Aviation Administration projects (MHRD201215)

References

[1] Bennell JA, Mesgarpour M, Potts CN. Airport runway scheduling. 4OR-Q J Oper Res. 2011; 9(2): 115-138.
[2] Beasley J E,Krishnamoorthy M,Sharaiha Y M,et al.Scheduling aircraft landings-the static case. Transportation Science.2000; 34(2): 180-197.
[3] Ernst AT, Krishnamoorthy M, Storer RH. Heuristic and exact algorithms for scheduling aircraft landings. *Networks.*1999; 34(3): 229-241.
[4] Cheng VHL, Crawford LS, Menon PK. *Air traffic control using genetic search techniques*. Proceedings of the IEEE International Conference on Control Applications, Hawaii, USA. 1999.
[5] Yu Hsin Liu. A genetic local search algorithm with a threshold accepting mechanism for solving the runway dependent aircraft landing problem. *Optim Lett*. 2011; 5: 229-245.
[6] James V, Hansen. Genetic search methods in air traffic control. *Computers & Operations Research.* 2004; 31(3): 445-459.
[7] MENG Xiangwei, ZHANG Ping, LI Chunjin. *Aircraft Category Based Genetic Algorithm for Aircraft Arrival Sequencing and Scheduling.* Proceedings of the 29th Chinese Control Conference, Beijing, China. 2010; 5188-5193.
[8] Hu XB, Chen WH. Receding horizon control for aircraft arrival sequencing and scheduling. *IEEE Transactions on Intelligent Transportation Systems*. 2005; 6(2): 189-197.
[9] Xiao-bing Hu, Wen-hua Chen. Genetic Algorithm Based on Receding Horizon Control for Arrival Sequencing and Scheduling. *Engineering Application of Artificial Intelligence*. 2005; 18(5): 633-642.
[10] Hu X B,Paolo E.Binary-representation-based genetic algorithm for aircraft arrival sequencing and scheduling. *IEEE Transactions on Intelligent Transportation Systems.* 2008; 9(2): 301-310.
[11] Liu Zhen, Li Rui2. Hybrid immune clonal algorithm for aircraft landing scheduling optimisation. *Computer Applications and Software*. 2013; 30(2): 116-121.