Expert Systems with Applications 38 (2011) 5787–5793

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Ant colony optimization-based algorithm for airline crew scheduling problem

Guang-Feng Deng *, Woo-Tsong Lin

Department of Management Information Systems, National Chengchi University, 64, Sec. 2, Chihnan Rd., Wenshan Dist, Taipei 116, Taiwan, ROC

ARTICLE INFO

Keywords:
Ant colony optimization (ACO)
Airline crew scheduling
Swarm intelligence
Combinatorial optimization problem

ABSTRACT

Airline crew scheduling is an NP-hard constrained combinatorial optimization problem, and an effective crew scheduling system is essential for reducing operating costs in the airline industry. Ant colony optimization algorithm (ACO) has successfully applied to solve many difficult and classical optimization problems especially on traveling salesman problems (TSP). Therefore, this paper formulated airline crew scheduling problem as Traveling Salesman Problem and then introduce ant colony optimization algorithm to solve it. Performance was evaluated by performing computational tests regarding real cases as the test problems. The results showed that ACO-based algorithm can be potential technique for airline crew scheduling.

1. Introduction

Crew cost, second to fuel cost is essential to airline carrier operations. The scheduling of crew members, which is the assignment of crew members to a flight for some period of time, is generally divided into crew scheduling problem and crew rostering problem. The objective of crew scheduling problem is to construct a set of feasible pairings that minimizes the total crew assigning cost and also satisfies the given flight schedule, the fleet routes, labor union and government regulations, and the company's own policy. For rostering problems, the pairings are assigned to crew members that satisfy their skills, vacations, and other requirements. This paper focuses on the problem of crew scheduling problem because crew scheduling results influence crew operating costs seriously and directly (Yan & Chang, 2002).

Airline crew scheduling problem (ACSP) is a difficult combinatorial optimization problem that is traditionally formulated as the set cover problem (SCP) or the set partition problem (SPP) and then used heuristic or mathematic programming to solve it. The vast studies applied this approach to ACSP include (Gamache, Hertz, & Ouellet, 2007; Levine, 1996; Park & Ryu, 2006; Yan & Tu, 2002; Yan & Chang, 2002). However, this approach has significant drawbacks. First, such an approach is computationally unsatisfactory, especially when the number of flights is large. Second, approaches using column generation techniques attempt to keep the best columns for SCP or SPP but may sometimes need suboptimal columns to produce better solution. For above question, Ozdemir and Mohan (2001) applied flight-based scheduling approach to ACSP and then used Genetic Algorithms to solve. The computational experiments showed that the flights scheduling approach can get better results than the SCP-based approach.

Ant colony optimization (ACO) algorithm that was proposed by Dorigo in 1992 is a meta-heuristic for combinatorial optimization problems. ACO that mimics a real ant colony with positive feedback characteristics has been noticed by researchers in the field of optimization (Bonabeau, Dorigo, & Theraulaz, 1999; Dorigo, Di Caro, & Gambardella, 1999; Engelbrecht, 2005). Many researchers in related fields have successfully applied ACO to solve many difficult and classical optimization problems such as the traveling salesman problems (TSP) (Dorigo & Gambardella, 1997; Wu, Zhao, Ren, & Quan, 2009), quadratic assignment problems (QAP) (Maniezzo & Colorni, 1999) constraint satisfaction problems (CSP), (Solnon, 2002) and multimodal problems (ToksarI, 2009).

When airline crew scheduling problems formulate as flight-based scheduling model, we find that this model is actually in the same form as the Traveling Salesman Problem with constrained. Therefore, this paper proposed ACO to solve airline crew scheduling problem by applying the flight-based scheduling representation and attempt to search shortest path from the flight graph such as TSP. The performance evaluation results indicate that ACO is an effective and efficient heuristic algorithm to minimize the total crew costs by effectively scheduling flights to reduce overnight stay in hotels, deadhead times and sitting time for solving airline crew scheduling problems.

The rest of this paper is organized as follows. Section 2 describes the constraints and cost function of our optimizing airline crew scheduling problems. Section 3 present details of the ACO-based algorithm solving the airline crew scheduling problem. In Section 4 the analysis of the performance of ACO-base algorithm is provided. Conclusions are finally drawn in Section 5, along with recommendations for future research.

* Corresponding author.
E-mail addresses: 95356502@nccu.edu.tw (G.-F. Deng), lin@mis.nccu.edu.tw (W.-T. Lin).

0957-4174/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2010.10.053

2. Airline crew scheduling problem

The goal of airline crew scheduling problem is to minimize the total crew costs by effectively scheduling flights to reduce overnight stay in hotels, deadhead times and sitting time to achieve feasible minimum cost of pairing sets. The cost function of ACSP is determined by summing up the payments of three parts, including required pay (minimum payment) for each duty in each pairing (rotation), rest expenses between pairings, and under-utilized time between duties in each pairing. For an airline crew scheduling problem that consists of n flights, m duties and o pairings of duties, the cost function of the ACSP can be defined as follows:

$$Min\sum_{p=1}^{m} C_{duty\_basic} + C^{fly}_{extre\_per\text{-}diem}\cdot\left[\left(\sum_{i=1}^{n} z_{ip}\cdot T^{fly}_{i}\right) - T^{\min}_{paid\_fly}\right] + \sum_{p=1}^{m}\sum_{q=1}^{m} y_{pq}\cdot\Big(C_{rest\_hotel} + C^{rest}_{per\text{-}diem}\cdot T^{rest}_{pq} + C_{under\text{-}utility}\cdot\left(T^{rest}_{pq} - T^{req\_rest}_{pq}\right)\Big) \tag{2.1}$$

The cost function components for the pairing sequence assigned to a crew includes:

(1) Required pay for each duty in each pairing:
C_{duty_basic}: the basic minimum payment for a duty, $C^{fly}_{extre\_per\text{-}diem}$: the bonus for overtime flying for each extra minute, $T^{\min}_{paid\_fly}$: minimum paid flying time in minutes in a duty. $\sum_{i=1}^{n} z_{ip}\cdot T^{fly}_{i}$: flying time in the duty p. T^{fly}_{i} is the flying time of flight i. Z_{ipl} is used to check whether flight i is contained in the duty p of the pairing o (set to 1) or otherwise (set to 0).

(2) Rest expenses between pairings:
The rest expenses include hotel and per-diem expenses when crew stays away from his/her domicile. T^{rest}_{pq} is the rest time between two consecutive duties p and q, and $C^{rest}_{per\text{-}diem}$ represents the additional pay for each minute of rest time, y_{pq} is designed to check whether duty q was flown after duty p, ($y_{pq} = 1$) or otherwise ($y_{pq} = 0$). C_{rest_hotel}: the hotel expense.

(3) Under-utilized time between duties in each pairing:
If the length of rest between two duties is longer than the required rest time ($T^{req\_rest}_{pq}$), then this under-utility crew time is added. $C_{under\text{-}utility}$: the loss to the company when one minute of a crew time is wasted.

The ACSP decision rules contain constraints on flying hours, and working and sitting hours for each duty. The decision rules for the ACSP are listed as follows:

(1) Consecutive city constraint: the next flight must leave from the destination city of the previous flight.

(2) Flight numbers constraints: these are settings of the maximum flights allowed in a duty ($N^{\max}_{duty}$) and the maximum flights allowed in a pairing ($N^{\max}_{pairing}$).

(3) Maximum sitting time for a duty: this is a setting of the maximum sitting time between flights ($T^{\max}_{sit}$). If a minimum time delay between consecutive flight legs in a duty is larger than $T^{\max}_{sit}$, the crew members must be sent to rest in a hotel.

(4) Working hour constraint for a duty: the total working time in a duty must be less than the maximum working time in a duty ($T^{\max}_{duty\_work}$):

$$T^{duty\_work}_{p} \leqslant T^{\max}_{duty\_work} \tag{2.2}$$

where $T^{\max}_{duty\_work}$: maximum work time in a duty. $T^{duty\_work}_{p}$: working time in a duty. It is calculated as following:

$$T^{duty\_work}_{p} = T_{check\_in} + T_{check\_out} + \left(\sum_{i=1}^{n}\sum_{j=1}^{n} z_{ipl}\cdot T^{fly}_{i} + x_{ij}\cdot T^{sit}_{ij}\right) \quad \forall\, l = 1,2,3,\ldots,o \quad \forall\, p = 1,2,3,\ldots,m \tag{2.3}$$

(5) Flying hour constraint for a duty: a duty can contain one or more shortly connected flights, but the total flying time in a duty must be less than the maximum flying time in a duty ($T^{\max}_{duty\_fly}$). The expression is as follows:

$$\sum_{i=1}^{n} z_{ip}\cdot T^{fly}_{i} \leqslant T^{\max}_{duty\_fly} \quad \forall\, p = 1,2,3,\ldots,m \tag{2.4}$$

(6) Sitting hour constraints:

- This is a setting of the minimum sitting time (or time delay) between consecutive flights ($T^{\min}_{sit}$):

$$T^{sit}_{ij} \geqslant T^{\min}_{sit} \quad \forall\, x_{ij} = 1 \quad i = 1,2,3,\ldots,n \quad \forall\, j = 1,2,3,\ldots,n \tag{2.5}$$

- A cabin crew that is assigned to a duty which is longer than 12 h, must be put to rest for at least 24 consecutive hours. Decision rule for this restriction can be expressed as follows:

$$T^{req\_rest}_{pq} \geqslant 24 \quad if\ T^{duty\_work}_{p} \geqslant 12\ \&\ y_{pq} = 1 \tag{2.6}$$

where $T^{req\_rest}_{pq}$: the minimum required rest time between consecutive duty p and duty qA cabin crew that is assigned to work for more than 8 h but less than 12 h should be put to rest for at least 12 consecutive hours. The expression is as follows:

$$T^{req\_rest}_{pq} \geqslant 12 \quad if\ T^{duty\_work}_{p} \geqslant 8\ \&\ y_{pq} = 1 \tag{2.7}$$

- A cabin crew that is assigned to a duty period for less than 8 hours should be given a scheduled rest period of at least 8 consecutive hours. The expression is as follows:

$$T^{req\_rest}_{pq} \geqslant 8 \quad if\ T^{duty\_work}_{p} \leqslant 8\ \&\ y_{pq} = 1 \tag{2.8}$$

3. Ant colony optimization (ACO) for airline crew scheduling

The idea of ACO was inspired by the foraging behavior of ant colonies that find the shortest route between ant's nest and a source of food by exchanging information via pheromone deposited on the trips. This pheromone information is used to guide the route search and let ants cooperate with each other as a whole community to achieve robust behavior capable of finding high quality solutions in a large search space. When applying ACO algorithm to solve combinatorial optimization problems, the main task is to model the problem as the search of the shortest path over a weighted and constraint graph. Then, ants walk though the graph, keep track of promising search routes by laying trails of pheromone, and look for good paths depending on both pheromone trail and some problem-specific local heuristic data.

This section explains details of construction of ACO-based algorithm for solving airline crew scheduling problems systematically. This work formulates the airline crew scheduling problem as a Traveling Salesman Problem (TSP) model by applying flight-based scheduling and builds the shortest path from the flight graph by ACO. After transforming the ACSP into a TSP-type model that uses a flight graph representation, the scheduling problem can then be recognized as the search for the path with minimum cost in a graph that uses flights as nodes of paths, and the connecting edges to conform to the constraints between two consecutive flights. So