SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

November 2017, Vol. 60 112202:1–112202:13
doi: 10.1007/s11432-016-9024-y

A large-scale multi-objective flights conflict avoidance approach supporting 4D trajectory operation

Xiangmin GUAN[1], Xuejun ZHANG[2*], Renli LV[1], Jun CHEN[3] & Weiszer MICHAL[3]

[1]*Department of General Aviation, Civil Aviation Management Institute of China, Beijing* 100102*, China;*
[2]*School of Electronic and Information Engineering, Beihang University, Beijing* 100191*, China;*
[3]*School of Engineering, University of Lincoln, Lincoln* LN6 7TS*, UK*

Received October 16, 2016; accepted January 20, 2017; published online September 1, 2017

Abstract Recently, the long-term conflict avoidance approaches based on large-scale flights scheduling have attracted much attention due to their ability to provide solutions from a global point of view. However, current approaches which focus only on a single objective with the aim of minimizing the total delay and the number of conflicts, cannot provide controllers with variety of optional solutions, representing different tradeoffs. Furthermore, the flight track error is often overlooked in the current research. Therefore, in order to make the model more realistic, in this paper, we formulate the long-term conflict avoidance problem as a multi-objective optimization problem, which minimizes the total delay and reduces the number of conflicts simultaneously. As a complex air route network needs to accommodate thousands of flights, the problem is a large-scale combinatorial optimization problem with tightly coupled variables, which make the problem difficult to deal with. Hence, in order to further improve the search capability of the solution algorithm, a cooperative co-evolution (CC) algorithm is also introduced to divide the complex problem into several low dimensional sub-problems which are easier to solve. Moreover, a dynamic grouping strategy based on the conflict detection is proposed to improve the optimization efficiency and to avoid premature convergence. The well-known multi-objective evolutionary algorithm based on decomposition (MOEA/D) is then employed to tackle each sub-problem. Computational results using real traffic data from the Chinese air route network demonstrate that the proposed approach obtained better non-dominated solutions in a more effective manner than the existing approaches, including the multi-objective genetic algorithm (MOGA), NSGAII, and MOEA/D. The results also show that our approach provided satisfactory solutions for controllers from a practical point of view.

Keywords air traffic management; conflict avoidance; combinatorial optimization; multi-objective; cooperative co-evolution

Citation Guan X M, Zhang X J, Lv R L, et al. A large-scale multi-objective flights conflict avoidance approach supporting 4D trajectory operation. Sci China Inf Sci, 2017, 60(11): 112202, doi: 10.1007/s11432-016-9024-y

1 Introduction

In recent years, the sharp increase in air traffic flow has reached the limits of airspace capacity which caused the air traffic congestion to become a more serious issue [1, 2]. As a result, the key airports and trunk routes of many countries and areas are facing a highly complicated traffic situation. In the local high-density operation, the safe separation among aircraft is often difficult to keep, which leads to

* Corresponding author (email: zhxj@buaa.edu.cn)

conflict situations and near-misses frequently. Furthermore, the air route network is very complicated with thousands of waypoints, air route segments and flights operations. On each air route segment, there are many flight levels in altitude height with about 300 m separation required for flights of different directions to avoid head-to-head conflict. With the rapid increase of air travel demand, the current airspace is becoming increasingly crowded and thus the conflict probability between aircraft especially at cross waypoints could dramatically rise. Along with the above mentioned problems, as the air traffic system is a tightly coupled and large-scale system with traffic flows intersecting each other, conflicts tend to spread within it, which not only impairs the safety, but also restricts sustained development of air transportation [3].

Conflict resolution approaches play a very important role in keeping a safe airspace. However, as the current sector-based air traffic system still cannot provide accurate traffic surveillance information covering a huge airspace, it is difficult to fully predict long-term conflicts and thus make decisions in advance to avoid them. As a result, current approaches are mainly focused on short-term conflict avoidance, which can efficiently solve conflicts in a relatively small short time window [4]. During the last decades, many approaches have been proposed, which can be mainly categorized into: rule-based methods [5], game theory methods [6, 7], field methods [8], geometric methods [9], numerical optimization methods [10–12], and multi-agent methods [13–15].

However, as the increase in air traffic flow continues, the above conflict resolution approaches cannot provide good solutions in terms of both effectiveness and timeliness due to the new features of the optimization problem, such as large scale, high complexity and tightly coupled variables. Moreover, without full consideration of the overall situation, providing short-term ad hoc solutions for flights could lead to a knock on effect due to the tight coupling between flights, which would jeopardize airspace safety [3].

In the recent years, the Federal Aviation Administration (FAA) and Eurocontrol proposed the concept of 4D-Trajectory (4DT) as the operation foundation of future air traffic management, which defines a flight trajectory using three spatial dimensions plus one time dimension. As the development of the advanced technology continues, flights can be accurately described in both space and time, which can significantly reduce the uncertainty of the flight trajectory. According to the initial operational experiment of the Eurocontrol, the uncertainty to all the waypoints of a flight path can be controlled within about 10 s. Most uncertainty will be eliminated through the adjustment of a flight, such as instant velocity change. As a result, the air traffic control can be realized with the current traffic situation and its evolutionary trend in a huge airspace. This also provides an operational and technical support for long-term management. Subsequently, the long-term conflict avoidance (LCA) method supporting 4DT operation has drawn much attention of researchers and practitioners from air traffic management domain, and it is envisioned as a key technology which can address the challenges caused by increased air traffic flow in the future [16, 17].

Considering thousands of flights in a complex air route network, the LCA problem is a large-scale combinatorial optimization problem with tightly coupled decision variables, as well as complicated constraints which make it difficult to solve by classical approaches. Therefore, an evolutionary algorithm (EA) is adopted [16]. A sliding forecast time window is introduced to reduce the dimension of the problem in order to obtain feasible solutions. However, it may overstock the large amount of flights in later time windows, causing a high difficulty for the EA-based approach to solve. Recently, a cooperative co-evolution (CC) strategy has been successfully used to handle the problem [18]. It uses a divide-and-conquer strategy to decompose the large-scale problem into several sub-problems which are easier to be solved. In the CC framework, the grouping strategy is a critical step especially for this large-scale complex problem. In order to improve the optimization efficiency, some other problem decomposition methods have been proposed, such as the splitting-in-half grouping [19], the correlation-based adaptive variable partitioning [20], the delta grouping [21], and the dependency identification technique [22]. Although these decomposition methods are effective in generic optimisation problems, they cannot take full advantage of the prior knowledge in order to minimise the interdependencies of the variables for the LCA problem.

Recently, with the aim to minimize the risk of premature convergence, a memetic algorithm (MA) is adopted [3]. It utilizes a specially designed local search operator and an adaptive local search frequency strategy to improve search capability of the algorithm. However, these previous works neglected the track error of flights, which makes them impractical. Furthermore, they considered the minimization of the aggregated flight delay and conflicts as a single objective [17]. While, in the real operation, controllers often try to seek a good trade-off between the flight delay and the number of conflicts.

In light of the above issues, in this paper, the conflict situation in the waypoint network is evaluated with consideration of track error of flights to make the model more practical and realistic. In order to incorporate more objectives, we formulate the long-term conflict avoidance problem as a multi-objective optimization problem, which can minimize the total delay and reduce the number of conflicts simultaneously. To further improve the search capability of the algorithm, a cooperative co-evolution algorithm is introduced to divide the complex problem into several low dimensional sub-problems [23]. Furthermore, a dynamic grouping strategy based on the conflict between flights is designed to improve search efficiency and to avoid premature convergence. The well-known multi-objective evolutionary algorithm based on decomposition (MOEA/D) is then employed to tackle each sub-problem separately [24]. Computational results using real traffic data from the Chinese air route network demonstrate that the proposed approach achieved better non-dominated solutions in a more efficiently manner than the existing approaches, such as the multi-objective genetic algorithm (MOGA) [25], NSGAII [26], or MOEA/D. The results also show that our approach can provide satisfactory solutions for controllers in a more practical sense.

The rest of this paper is organized as follows. Firstly, the problem is formulated in Section 2. Section 3 presents the details of our solution approach. The results of computational experiments are presented and analyzed in Section 4. Finally, some conclusions and future research directions are drawn in Section 5.

2 Problem formulation

The problem described in this paper can be formulated as follows. Let W denotes the set of waypoints in the considered airspace, then the waypoint sequence of the trajectory of flight i is $\left\{W_j{}^i\right\}_{j=0,\dots,nw_i,}$ $W_j{}^i \in \mathbb{R}^2$ where j is the index of the waypoint in the sequence, nw_i is the number of waypoints in the path of flight i. There are n flights $(F_1, F_2 \dots F_n)$ in total with specific flight plans. The velocity of flight i in each segment is $\left\{V_j{}^i\right\}_{j=0,\dots,nw_i,}$ $V_j{}^i \in \mathbb{R}_+{}^2$. Without consideration of the track error, the estimated arrival time at each waypoint of fight i can be obtained by [27]

$$T_j{}^i = \frac{\left\|W_j{}^i - W_{j-1}{}^i\right\|}{v_j{}^i} + T_{j-1}{}^i, \quad j = 1, \dots, nw_i, \tag{1}$$

where $T_0{}^i = 0$ and $W_0{}^i$ is the first waypoint of the path of flight i. The flight distance s of flight i at time t is

$$s^i(t) = v_j{}^i(t - T_{j-1}{}^i) + s^i(T_{j-1}{}^i), \quad t \in (T_{j-1}{}^i, T_j{}^i]. \tag{2}$$

The current position p of flight i at time t is

$$p^i(t) = p^i(T_{j-1}{}^i) + v_j{}^i(t - T_{j-1}{}^i)\frac{(W_j{}^i - p^i(T_{j-1}{}^i))}{\left\|W_j{}^i - p(T_{j-1}{}^i)\right\|}, \tag{3}$$

where $s^i(T_0{}^i) = 0$, and $p^i(T_0{}^i) = W_0{}^i$.

Under the operation of the sector-based air traffic management, the track error of flights in general obeys a Gaussian distribution where the mean is zero, and the horizontal standard deviation δ_s^2 is defined by

$$\delta_s^2(t) \sim r_s^2 t^2, \tag{4}$$

and the lateral standard deviation is described by

$$\delta_c^2(t) \sim \min\{r_c^2 s^2(t), \bar{\delta}_c^2\}, \tag{5}$$

where $\bar{\delta}_c^2$ is the maximum of the lateral standard deviation. We can see that the horizontal standard deviation and the lateral standard deviation will increase as the time grows, and generally $\delta_s(t)$ is larger than $\delta_c(t)$. In addition, the vertical standard deviation is a constant.

However, under the operation of 4D trajectory, the accuracy of the flight path could be greatly improved. Moreover, with the help of the flight management system, flights can arrive at each waypoint with higher precision. Therefore, in this paper, both the horizontal standard deviation and the lateral standard deviation are considered to be constant and are defined by δ_s and δ_c, respectively. In addition, the estimated arrival time at each waypoint is assumed to obey a Gaussian distribution with zero mean and δ_{tw} as the standard deviation.

Suppose that the angle between the current velocity of flight i and x axis is θ_j in the plane coordinate system, and in the body coordinate system it can be denoted by

$$R(\theta_j) = \begin{pmatrix} \cos\theta_j & -\sin\theta_j \\ \sin\theta_j & \cos\theta_j \end{pmatrix}. \tag{6}$$

Hence, the predicted position of flight i at time t can be obtained by

$$X^i(t) = p^i({T_{j-1}}^i) + {v_j}^i(t - {T_{j-1}}^i + \delta_{tw}^2)\frac{({W_j}^i - p^i({T_{j-1}}^i))}{\left\|{W_j}^i - p({T_{j-1}}^i)\right\|} + D, \tag{7}$$

where D is a covariance matrix, and $D = R(\theta)\bar{D}R(\theta)^{\mathrm{T}}$, with $\bar{D} = \left(\begin{smallmatrix} \delta_s^2 & \\ & \delta_c^2 \end{smallmatrix}\right)$ if

$$CD = {v_j}^i \frac{({W_j}^i - p^i({T_{j-1}}^i))}{\left\|{W_j}^i - p({T_{j-1}}^i)\right\|}\delta_{tw}^2 + D. \tag{8}$$

Then, $X^i(t)$ can be defined by

$$X^i(t) \sim N(P^i(t), CD). \tag{9}$$

Considering the flight set F in a time window, the distance function between any two flights i and j is denoted by

$$\mathrm{dist}_{ij}(t) = \left\|X_i(t) - X_j(t)\right\|. \tag{10}$$

It is assumed that the positions of flights are not relevant, so $\mathrm{dist}_{ij}(t)$ obeys a Gaussian distribution as follows:

$$\mathrm{dist}_{ij}(t) \sim N(P^i(t) - P^j(t), 2CD). \tag{11}$$

Then, the conflict probability $PC_{ij}(t)$ of two flights i and j at time t can be computed by

$$PC_{ij}(t) = \int\limits_{\mathrm{dist}_{ij} < \varepsilon_{ij}} p_{ij}^{d_t}(y)\mathrm{d}y, \tag{12}$$

where $p_{ij}^{d_t}(y)$ is the probability density function of $\mathrm{dist}_{ij}(t)$. The conflict situation (CS) of all flights in the considered airspace can be defined by

$$\mathrm{CS} = \sum_{i=1}^{n}\sum_{j>i}^{n}\mathrm{MPC}_{ij}, \tag{13}$$

where MPC_{ij} is the maximum conflict probability of two flights, and it can be described by

$$\mathrm{MPC}_{ij} = \underset{t\in[T_{ij}^1, T_{ij}^2]}{\mathrm{MAX}}(PC_{ij}(t)). \tag{14}$$

Hence, the first objective is formulated to minimize the total maximum conflict probability, and it can be defined by

$$\min\ f_1 = \mathrm{CS}. \tag{15}$$

In this work, the ground delay method is used to avoid conflict at waypoints, which is an effective way by delaying flights while they are still on the ground before departure. However, in order to reduce the cost for airlines, the sum of flight delays is formulated as the second objective which is defined by

$$\min\ f_2 = \frac{1}{n}\sum\nolimits_i^n \delta_i, \tag{16}$$

where δ_i presents the departure delay of flight i, and $\delta_i \in [0\ \delta_{\max}/ts]$, where $\delta_{\max}$ is the maximum allowable delay. It means that the delay of any flight is limited by a maximum value in order to prevent some flights being postponed for too long. ts is the time step for time sampling.

It can be demonstrated that the LCA problem is a large-scale combination optimization problem with two objectives. Furthermore, the variables and constraints are tightly coupled because of conflict avoidance.

3 Optimization framework

In order to solve the abovementioned optimization problem in an efficient manner and to avoid premature convergence, an efficient multi-objective optimization framework is proposed in this section. Firstly, CC algorithm is introduced to divide the complex problem into several low dimensional sub-problems. Towards this aim, a dynamic grouping strategy based on the conflict between flights is designed as a heuristic strategy. Then, the well-known multi-objective evolutionary algorithm based on decomposition (MOEA/D) is employed to solve each sub-problem. The framework is described in Algorithm 1.

Algorithm 1 The framework of the proposed method

```
Initialize the population g = 0.
//Main loop:
while g < maxgen do
    Evaluate all individuals in the population.
    Compute the non-dominated solutions.
    //cooperative co-evolution.
    Divide the decision variables into groups based on the dynamic grouping strategy.
    Decision variables in each group generate its subpopulation.
    for each subpopulation do
        Use the MOEA/D framework with a genetic algorithm.
        Evaluate all individuals in the subpopulation, and compute the non-dominated solutions.
    end for
    Obtain the non-dominated solutions.
    g = g + 1.
end while
```

In the following subsections, some important mechanisms, such as the dynamic grouping strategy, subcomponent optimization, adaptive crossover, and mutation operators are elaborated in more details.

3.1 The dynamic grouping strategy

The cooperative co-evolution algorithm has two critical steps. In this section, we mainly describe the dynamic grouping strategy which is used to divide flights into groups based on conflicts.

In order to describe if two flights conflict with each other, a matrix C [28] is adopted in this work as defined below:

$$C = \begin{pmatrix} C_{11} \cdots C_{1n} \\ \vdots \quad \vdots \quad \vdots \\ C_{n1} \cdots C_{nn} \end{pmatrix} \tag{17}$$

where

$$C_{ij} = \begin{cases} 1, & \text{if } F_i \text{ and } F_j \text{ conflict, } i \neq j; \\ 0, & \text{otherwise;} \end{cases} \quad i, j = 1, \ldots, n. \tag{18}$$

First, if there is no conflict among any two flights, the random grouping strategy will be employed, which randomly divides the flights into ns groups with the same size.

Secondly, if there are at least two flights which conflict with each other, i.e.,

$$\exists i \neq j, \quad \text{s.t.} \quad C_{ij} = 1, \tag{19}$$

then, the flights are divided into sn groups based on the dynamic grouping strategy which can be defined by

$$\text{group}_k = \left({F_k}^{(1)}, {F_k}^{(2)}, \ldots, {F_k}^{(m_k)} \right), \ 1 \leqslant k \leqslant sn, \ 1 \leqslant m_k < n, \ \sum_{k=1}^{sn} m_k = n, \tag{20}$$

where ${F_k}^{(j)}$ denotes the jth flight in group_k and m_k indicates the number of flights in group_k.

The flights in each group satisfy

$$\forall a \in \text{group}_k, \ \forall b \in \text{group}_l, \quad \text{s.t.} \quad C_{ab} = 0, \tag{21}$$

and flights from different groups satisfy

$$\forall a \in \text{group}_k, \ \forall b \in \text{group}_l, \quad \text{s.t.} \quad C_{ab} = 0. \tag{22}$$

3.2 Subcomponent optimization

In this work, the fast genetic algorithm (GA) is proposed as the global search method [28].

Another critical point is the optimization of each group. In this paper, a fast GA is incorporated into the MOEA/D framework.

The sub-population of each group includes ps individuals indicating the possible solutions of flights in this group. Hence the sub-population is a matrix defined by

$$\text{subpop}_k = \left\{ {f_k}^{(1)}, {f_k}^{(2)}, \ldots, {f_k}^{(ps)} \right\}, \quad 1 \leqslant k \leqslant sn, \tag{23}$$

where ${f_k}^{(i)} (1 \leqslant i \leqslant ps)$ is a vector which can be defined by

$${f_k}^{(i)} = ({\delta_k}^{(i1)}, {\delta_k}^{(i2)}, \ldots, {\delta_k}^{(im_k)}), \quad 1 \leqslant m_k < n \quad \sum_{k=1}^{sn} m_k = n, \tag{24}$$

where ${\delta_k}^{(ij)}$ denotes the delay time slot of flight ${F_k}^{(j)}$ of chromosome j in group_k.

The general framework of MOEA/D [24] is shown in Algorithm 2.

The adaptive crossover and mutation operators are specially designed for the LCA problem based on the fitness of each gene in the individual. The fitness takes the ground delay and conflict probability of flights into account. The fitness of each flight in group_k is defined by

$$\text{fit}_k^j = \frac{1 - \delta_k^j / \delta_{\max}}{1 + cs_{kj}} \ (1 \leqslant j \leqslant m_k), \tag{25}$$

where cs_{kj} is the total conflict probability of flight j with other flights.

The mechanism of the adaptive crossover is shown in Figure 1. In this example, A and B are parents in sub-population k. If $\text{fit}_k^{A_1} > \text{fit}_k^{B_1}$, the two children will inherit from A_1 accordingly, and if $\text{fit}_k^{B_1} > \text{fit}_k^{A_1}$, they inherit from B_1. Otherwise, the genes of children are obtained in a way as follows:

$$\begin{aligned} CA_1 &= \text{floor}(\alpha A_1 + (1 - \alpha) B_1), \\ CB_1 &= \text{floor}(\alpha B_1 + (1 - \alpha) A_1), \end{aligned} \tag{26}$$

where α is the parameter of the linear combination.

For adaptive mutation operator, as can be seen from Figure 2, if $gf_k^j < \varepsilon$, the gene j mutates with a probability of p_k.

Algorithm 2 Algorithmic flow of MOEA/D with GA

Input:

(1) A stopping criterion;

(2) np: the number of the sub-problems;

(3) An uniform spread of n weight vectors: $\lambda^1, \dots, \lambda^{np}$;

(4) T: the number of the weight vectors in the neighborhood of each weight vector;

Output: Approximation to the PF and PS.

Procedure:

Step 1 Initialization:

Step 1.1 Compute the Euclidean distances between the weight vectors and work out the T closest weight vectors to each weight vector. For each $i = 1, \dots, np$, set $B(i) = \{i_1, \dots, i_T\}$, where $\lambda^{i_1}, \dots, \lambda^{i_T}$ are the T closest weight vectors to λ^i.

Step 1.2 Generate an initial population $x^1, \dots, x^{np}$. Calculate the fitness values of the population.

Step 1.3 Initialize $z = (z_1, \dots, z_m)$, where $z_j = \min_{1 \leqslant i \leqslant n} f_j(x^i)$.

Step 2 Update:

For $i = 1, \dots, np$

Step 2.1 Selection of the mating pool:

Generate a random number which is uniformly distributed in [0, 1]. Set

$$P = \begin{cases} B(i), & \text{if rand} < \delta; \\ \{1, \dots, np\}, & \text{otherwise.} \end{cases}$$

Step 2.2 Reproduction:

Set $r_1 = i$, and randomly select two indexes k, l from P, and then generate a new solution y using mutation and crossover operators of GA.

Step 2.3 Update of the reference point: For each $j = 1, \dots, m$, if $z_j > f_j(y)$, then set $z_j = f_j(y)$.

Step 2.4 Replacement of solutions

Step 3 Stopping Criterion:

If the stopping criteria is satisfied, then stop the algorithm and output PF and PS. Otherwise, go to **Step 2**.

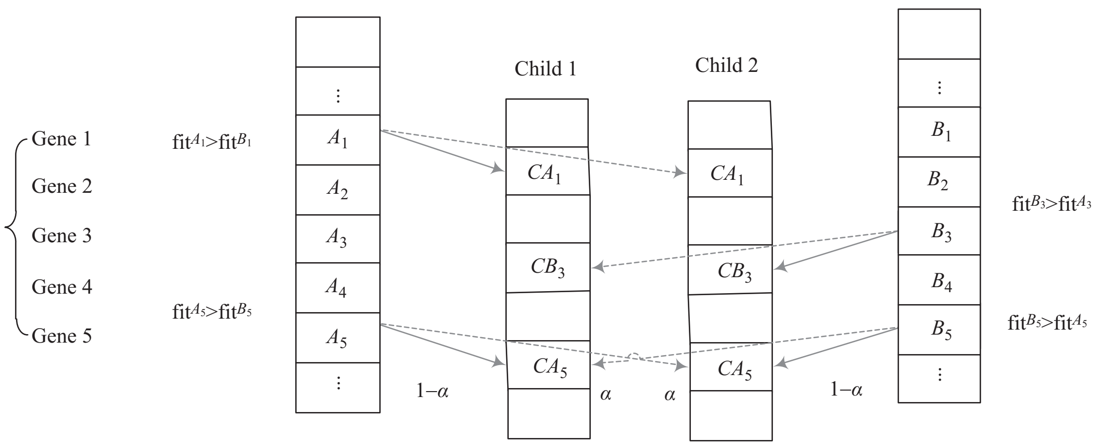

Figure 1 Adaptive crossover operator.

4 Experimental studies

4.1 Database and experimental setup

The national route network of China consists of 1706 air route segments, 940 waypoints and 150 airports as shown in Figure 2. The air traffic data was obtained from Civil Aviation Administration of China (CAAC) for a whole day of 7 October, 2009. It is worth mentioning that the takeoff and landing phases of flights are truncated within a given radius (usually 10 n mile) around airports. The traffic around airports is managed following specific procedures imposed by the terminal control area (TCA) control services in these zones.

The minimum safe time interval is equal to $\tau = 60$ s. $\delta_{\max}$ is set to be 90 min, the value interval of δ is 0.25 min, and $\varepsilon = 0.3$.

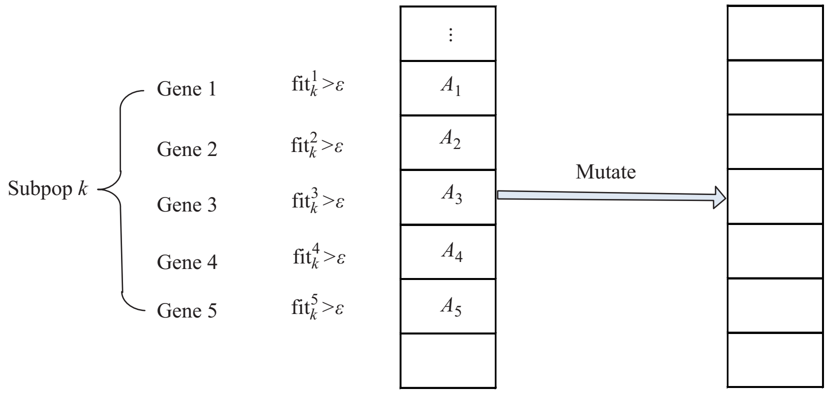

Figure 2 Adaptive mutate operator.

Table 1 Parameters of the experiments

Parameters	Description	Value
ps	Population size	100
maxgen	Max generation	500
p_c	Crossover probability	0.8
p_m	Mutate probability	0.1

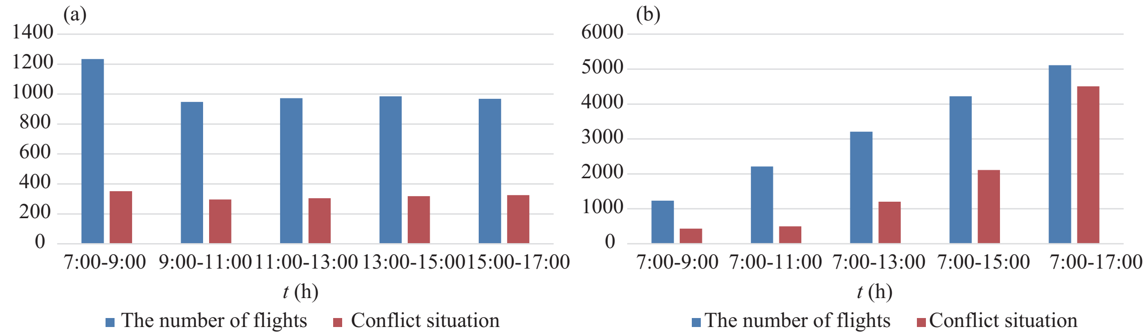

Figure 3 (Color online) (a) The relationship between the number of flights and the conflict situation in every two hours; (b) the relationship between the number of flights and the conflict situation as the considered time accumulates.

In order to compare with the proposed MOCC, MOEA/D, MOGA [29], NSGA2 [30] are selected, and all these algorithms were implemented in C++ in this work. Computational experiments were carried out on a computer with an E5620 2.4 GHz CPU with 12 GB RAM. For each algorithm, the results were collected and analyzed based on 15 independent runs.

The parameters used in all experiments are listed in Table 1, and they are often adopted in other algorithms [18].

4.2 The depiction of conflict situation

Next, the relationship between the number of flights and the conflict situation in the considered airspace is depicted in Figure 3. We can see that there are about 1000 flights during every two hours in Figure 3(a). The number of flights from 7 a.m. to 9 a.m. is the largest. The total maximum conflict probability of all flights in each time period is about 300. In addition, in Figure 3(b), it can be seen that as the number of flights grows, the total maximum conflict probability increases quickly.

4.3 Comparison with the existing methods

In order to compare the performance of the above mentioned algorithms, two scenarios including 960 flights (represent the busiest one hour) and 1664 flights (represent the busiest three hours) are considered.

Table 2 Comparison of different algorithms for 960 flights (I_H, γ, Δ)

Algorithms	I_H	γ	Δ
MOGA	3994	98.99	1.313
NSGA2	5468	68.93	1.213
MOEA/D	6378	55.23	1.263
MOCC	6731	43.63	1.010

Table 3 Comparison of different algorithms for 1664 flights (I_H, γ, Δ)

Algorithms	I_H	γ	Δ
MOGA	4119	86.30	1.1751
NSGA2	5249	56.73	1.2426
MOEA/D	5951	63.30	1.3448
MOCC	6467	57.45	0.8181

In addition, three typical metrics are adopted to evaluate the performance of the solutions obtained by each of the algorithms. The convergence metric (γ) [26], the spread metric (Δ) [31], and the hypervolume metric I_H is used [32, 33].

Tables 2 and 3 summarize the average values of I_H, I_D and Δ over 15 independent runs. The best results are highlighted in boldface in each row of the table. We can see from both tables that the proposed algorithm outperforms the other three algorithms in all metrics. Moreover, when the number of flights increases, it performs even better. Therefore, it is concluded that MOCC has superiority in solving large-scale problems such as the one in this paper.

Additionally, the non-dominated solutions with the least DTC and with the least CS obtained by the all algorithms over 15 runs are listed in Tables 4 and 5 under the two scenarios separately. It can be observed that both of the non-dominated solutions with the least DTC and the least CS obtained by MOCC are not dominated by the corresponding solutions of the other three algorithms when the number of flights is 960. In indeed, in most comparisons under this scenario, MOCC provides better solutions in both objectives. The same conclusion applies to the scenario when the number of flights is 1664.

Figure 4 shows the non-dominated solutions obtained by respective algorithms. Specifically, the non-dominated solutions of each algorithm were obtained over 15 runs. From Figure 4, it can be concluded that MOCC performs the best because its solutions dominate those obtained by other algorithms. Among all algorithms, MOGA has the worst performance in terms of convergence. MOEA/D performs better than NSGA2 in terms of convergence and diversity.

From the experimental results, we conclude that MOCC performs better than the other three methods for both scenarios. MOCC adopts an effective multi-objective optimization framework based on the CC (i.e. dynamic grouping) and MOEA/D, greatly improving its the search capability. The CC divides the complex problem into several low-dimension sub-problems, which makes the problem easier to solve. The sub-problems work cooperatively to obtain better solutions. Furthermore, the CC takes full advantage of the characteristics of the long-term conflict avoidance problem and is based on the conflict among flights, leading to improved search efficiency. The improved search performance is also due to the employment of the well known multi-objective evolutionary algorithm based on decomposition (MOEA/D) to solve each sub-problem.

4.4 Comparison between dynamic grouping strategy and other popular strategies

The experiment in this section is designed to further investigate the contribution of the proposed dynamic grouping strategy. The grouping strategy is a key issue in the CC-based framework. There are several popular grouping strategies, e.g. one-dimensional grouping strategy, splitting-in-half grouping strategy, and random grouping strategy [19]. In the following, two of these grouping strategies are compared with the proposed one. All grouping strategies are implemented within the same CC-based framework and share exactly the same settings. The two grouping strategies for comparison are briefly described as follows:

Table 4 Non-dominated solutions with the least CS and the least DTC for 960 flights

Algorithms	Solutions with least CS		Solutions with least DTC	
	CS	DTC	CS	DTC
MOGA	62.02	18.76	140.6	9.552
NSGA2	22.52	20.00	138.9	6.183
MOEA/D	15.17	13.40	109.4	3.256
MOCC	0.3841	15.42	108.5	2.692

Table 5 Non-dominated solutions with the least CS and the least DTC for 1664 flights

Algorithms	Solutions with least CS		Solutions with least DTC	
	CS	DTC	CS	DTC
MOGA	52.67	21.12	178.2	8.873
NSGA2	17.91	22.46	101.3	6.854
MOEA/D	31.21	14.03	137.3	3.202
MOCC	0.4173	15.44	116.2	2.734

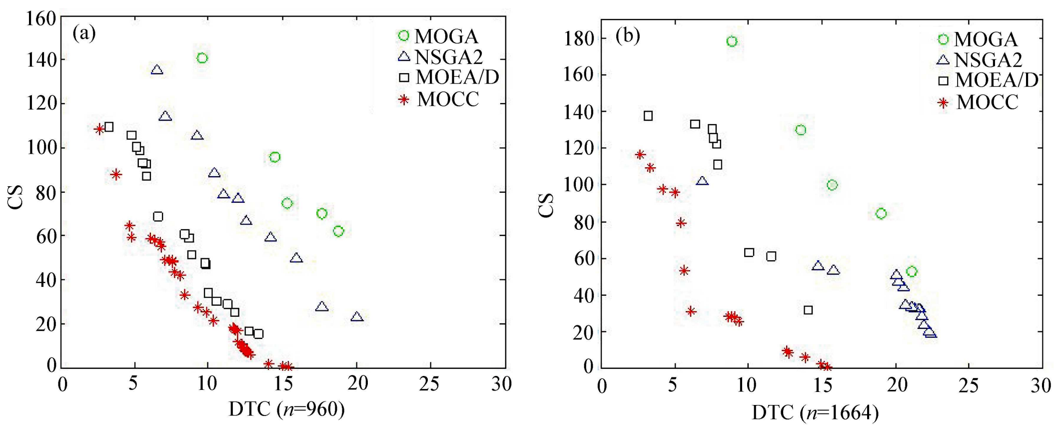

Figure 4 (Color online) Adaptive mutate operator.

Table 6 Comparison of different algorithms for 960 flights (I_H, γ, Δ)

Algorithms	I_H	γ	Δ
MOCC-SIH	5534	61.23	1.161
MOCC-RG	6028	54.38	1.072
MOCC	6731	43.63	1.010

Table 7 Comparison of different algorithms for 1664 flights (I_H, γ, Δ)

Algorithms	I_H	γ	Δ
MOCC-SIH	5370	62.68	1.1754
MOCC-RG	5875	60.45	1.0864
MOCC	6467	57.45	0.8181

- Splitting-in-half based strategy (SIH): Each sub-group contains half of the total aircraft.
- Random grouping strategy (RG): All the aircraft are randomly divided into several sub-groups.

Tables 6 and 7 show the average value of I_H, I_D and Δ over 15 independent runs of the algorithms for respective scenarios. The best value is highlighted in boldface in each row of the table. It can be concluded from both tables that the proposed algorithm outperforms the other three algorithms in terms of I_H, I_D and Δ. Hence, the dynamic grouping strategy has superiority in solving large-scale problems such as the one in this paper.

The non-dominated solutions with the least delay time cost (DTC) and the least CS obtained by all algorithms over 15 runs are listed in Tables 8 and 9. We can see that under both scenarios, the dynamic

Table 8 Non-dominated solutions with the least CS and the least DTC for 960 flights

Algorithms	Solutions with least CS		Solutions with least DTC	
	CS	DTC	CS	DTC
MOCC-SIH	58.67	17.47	154.4	8.754
MOCC-RG	10.54	16.98	113.5	4.785
MOCC	0.3841	15.42	108.5	2.692

Table 9 Non-dominated solutions with the least CS and the least DTC for 1664 flights

Algorithms	Solutions with least CS		Solutions with least DTC	
	CS	DTC	CS	DTC
MOCC-SIH	64.35	24.98	189.2	9.358
MOCC-RG	12.57	23.47	164.2	7.426
MOCC	0.4173	15.44	116.2	2.734

grouping strategy performs the best in both objectives.

The splitting-in-half grouping strategy cannot cope with this large-scale problem with more than half flights in each group still vulnerable to potential conflicts. Although the random grouping strategy can reduce such potential conflicts in the case of the two interacting flights in the same group, its performance will drop dramatically when there are more than two interacting flights. In general, the splitting-in-half and random grouping strategies represent a blind search mechanism and are more easily to be trapped in a local optimum. On the contrary, the proposed dynamic grouping strategy exploits the pattern reflected in potential conflicts among flights leading to an improved global search capability.

4.5 Application to real operations

In this section, we further investigate the applicability of the proposed approach in real operations, i.e. its ability to provide feasible solutions for the air traffic controllers to keep safe separation of flights.

It is worth mentioning that the proposed method in this paper is a pre-tactical approach which can be used to solve conflicts that happen in a time scale from several hours to a few days in advance. Therefore, we do not consider disturbances. More specifically, the computational time needed to get feasible solutions of MOCC is about 5 min and 17 min for scenario with 960 and 1664 flights, respectively. This is sufficient for a real pre-tactical application. About 30 non-dominated solutions in scenario 1 and 20 non-dominated solutions in scenario 2 are obtained. Practically, controllers may only need a few feasible solutions. Therefore, the computation time can be much shorter The computation time can be further reduced using more advanced parallel computation technology.

We also noticed that even for the scenario with 1664 flights, the average number of conflicts using MOCC is almost 0 and the average delay can be controlled within 15 min. Furthermore, as can be seen from Figure 4, when the average delay is within 10 min, the maximum number of flights will be under 20 which can be comfortably handled by air traffic controllers.

In conclusion, the proposed MOCC can largely improve the optimization capability and avoid local optima. It represents the best search and grouping strategy among all solution approaches dealing with the long-term conflict avoidance problem. Although the current version of MOCC cannot be applied to a real time application, it is sufficient for a pre-tactical management application.

5 Conclusion and future work

In this paper, a novel long-term conflict avoidance approach supporting the 4DT operation is proposed to provide better strategic flight flow management solutions. Taking the flights track error into consideration, the LCA problem is firstly formulated as a multi-objective problem minimizing the total delay and the number of conflicts simultaneously. Considering that the LCA problem is a large-scale combinatorial optimization problem with tightly coupled variables, in this work, CC algorithm is introduced to divide the complex problem into several low-dimensional sub-problems to further improve the searching capability. A dynamic grouping strategy based on the conflict between flights is proposed to improve the optimization efficiency and avoid premature convergence. To fully utilize the proposed grouping strategy, the well-known MOEA/D is employed in search of better solutions for each sub-problems. The proposed approach has been validated using real traffic data from Chinese air route network, and the results demonstrate that the proposed approach obtained better non-dominated solutions than the existing approaches including the MOGA, NSGA2, and MOEA/D. The results also show that our approach can provide satisfactory solutions for controllers under real operational scenarios.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. U1533119, U1433203), and Foundation for Innovative Research Groups of the National Natural Science Foundation of China (Grant No. 61221061).

Conflict of interest The authors declare that they have no conflict of interest.

References

1 Liu W Y, Hwang I. Probabilistic trajectory prediction and conflict detection for air traffic control. AIAA J Guid, Contr Dynam, 2011, 34: 1779–1789

2 Lv R L, Guan X M, Li X Y, et al. A large-scale flight multi-objective assignment approach based on multi-island parallel evolution algorithm with cooperative coevolutionary. Sci China Inf Sci, 2016, 59: 072201

3 Guan X M, Zhang X J, Han D, et al. A strategic flight conflict avoidance approach based on memetic algorithm. Chinese J Aeron, 2013, 27: 93–101

4 Kuchar J, Yang L. A review of conflict detection and resolution modeling methods. IEEE Trans Intell Transp Syst, 2000, 1: 179–189

5 Hwang I, Tomlin C. Protocol-based conflict resolution for finite information horizon. In: Proceedings of American Control Conference, Anchorage, 2002. 748–753

6 Archibald K, Hill J C, Jepsen N A, et al. A satisficing approach to aircraft conflict resolution. IEEE Trans Syst, Man, Cybern C, 2008, 38: 510–521

7 Tomlin C, Mitchell I, Ghosh R. Safety verification of conflict resolution manoeuvres. IEEE Trans Intell Transp Syst, 2001, 2: 110–120

8 Kosecka K, Tomlin C, Pappas G, et al. Generation of conflict resolution maneuvers for air traffic management. In: Proceedings of International Conference on Intelligent Robots and Systems, Grenoble, 1997. 1598–1603

9 Mao Z H, Dugail D, Feron E. Space partition for conflict resolution of intersecting flows of mobile agents. IEEE Trans Intell Transp Syst, 2007, 8: 512–527

10 Pallottino L, Feron E M, Bicchi A. Conflict resolution problems for air traffic management systems solved with mixed integer programming. IEEE Trans Intell Transp Syst, 2003, 3: 3–11

11 Mondoloni S, Conway S. An Airborne Conflict Resolution Approach Using a Genetic Algorithm. NASA Langley Technical Report, AIAA-2001-4054. 2001

12 Vivona R, Karr D, Roscoe D. Pattern based genetic algorithm for airborne conflict resolution. In: Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit, Keystone, 2006. AIAA-2006-6060

13 Pechoucek M, Sislak D. Agent-based approach to free-flight planning, control, and simulation. IEEE Intell Syst, 2009, 24: 14–17

14 Hill J C, Johnson F R, Archibald J K, et al. A cooperative multi-agent approach to free flight. In: Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems, Utrecht, 2005. 1083–1090

15 Rong J, Geng S, Valasek J, et al. Air traffic control nego-tiation and resolution using an on board multi-aircraft system. In: Proceedings of the 21st Digital Avionics System Conference, Irvine, 2002. 69–80

16 Durand N, Allignol, C. 4D-Trajectory deconfliction through departure time adjustment. In: Proceedings of Conference on International Air Traffic Management R&D Seminar, Napa, 2009. 1–10

17 Durand N, Alliot J M, Noailles J. Automatic aircraft conflict resolution using genetic algorithms. In: Proceedings of the 1996 ACM symposium on Applied ComputingSymposium on Applied Computing, Pennsylvania, 1996. 289–298

18 Su J, Zhang X J, Guan X M. 4D-Trajectory conflict resolution using cooperative coevolution. In: Proceedings of International Conference on Information Technology and Software Engineering, Beijing, 2012. 387–395

19 Yang Z, Tang K, Yao X. Large scale evolutionary optimization using cooperative co-evolution. Inform Sci, 2008, 178: 2985–2999

20 Ray T, Yao X. A cooperative coevolutionary algorithm with correlation based adaptive variable partitioning. In: Proceedings of the 11th Congress on Evolutionary Computation, Montreal, 2009. 983–989

21 Omidvar M N, Li X D, Yao X. Cooperative coevolution with delta grouping for large scale non-separable function optimization. In: Proceedings of the IEEE Congress on Evolutionary Computation, Barcelona, 2010. 1762–1769

22 Sayed E, Essam D, Sarker R A. Dependency identification tech-nique for large scale optimization problems. In: Proceedings of the IEEE Congress on Evolutionary Computation, Brisbane, 2012. 1–8

23 Potter M, Jong K D. A cooperative coevolutionary approach to function optimization. In: Proceedings of the 3rd Conference on Parallel Problem Solving from Nature, London, 1994. 249–257
24 Zhang Q, Li H. MOEA/D: A multi-objective evolutionary algo-rithm based on decomposition. IEEE Trans on Evolution Comput, 2007, 11: 712–731
25 Delahaye D, Sofiane O, Puechmorel S. Airspace congestion smoothing by multi-objective genetic algorithm. In: Proceedings of ACM Symposium on Applied Computing, Santa Fe, 2005. 907–912
26 Deb K, Pratap A, Agarwal S. A fast and elitist multi-objective genetic algorithm: NSGA-II. IEEE Trans Evolution Comput, 2002, 6: 182–197
27 Prandini M, Hu J H, Lygeros J, et al. A probabilistic approach to aircraft conflict detection. IEEE Trans Intell Transp Syst, 2000, 1: 199–220
28 Zhang X J, Guan X M, Hwang I, et al. A hybrid distributed-centralized conflict resolution approach for multi-aircraft based on cooperative co-evolutionary. Sci China Inf Sci, 2013, 56: 128202
29 Zitzler E, Thiele L, Laumanns M. Performance assessment of multiobjective optimizers: An analysis and review. IEEE Trans on Evolution Comput, 2003, 7: 117–132
30 Fleischer M. The measure of Pareto optima: applications to multiobjective metaheuristics. In: Proceeding of the 2nd International Conference on Evolutionary Multi-objective Optimization, Faro, 2003. 519–533
31 Wilcoxon F. Individual comparisons by ranking methods. Biometrics Bull, 1945, 1: 80–83
32 Delahaye D, Sofiane O, Puechmorel S. Airspace congestion smoothing by multi-objective genetic algorithm. In: Proceedings of ACM Symposium on Applied Computing, Santa Fe, 2005. 907–912
33 Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans on Evolution Comput, 2002, 6: 182–197