A Flow-based Flight Scheduler for En-route Air Traffic Management

Li, Qing; Zhang, Yicheng; Su, Rong.

2016

Li, Q., Zhang, Y., & Su, R. (2016). A Flow-based Flight Scheduler for En-route Air Traffic Management. IFAC-PapersOnLine, 49(3), 353-358.

https://hdl.handle.net/10356/83839

https://doi.org/10.1016/j.ifacol.2016.07.059

Available online at www.sciencedirect.com

ScienceDirect

IFAC-PapersOnLine 49-3 (2016) 353–358

A Flow-based Flight Scheduler for En-route Air Traffic Management

Qing Li*, Yicheng Zhang, and Rong Su****

**Air Traffic Management Research Institute, Nanyang Technological University, Singapore 639798.*
(e-mail: liqing@ntu.edu.sg)
***School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798.*
(e-mails: yzhang088@e.ntu.edu.sg, rsu@ntu.edu.sg)

Abstract: This paper addresses a network-wide scheduling problem for generating, adjusting and optimizing flight schedules in en-route airspace. The aim of scheduling is twofold: to support flight plans by providing flight schedules reliable on optimal flow restrictions; and to assist en-route traffic control by ensuring safe and ordered trajectory of traffic. Firstly, we design an algorithm to generate continuous-time flight schedules based on discrete-time flow assignments of aircrafts. This algorithm is further enhanced with consideration of sequence optimization for merging traffic. A flight scheduling system has been developed in a simulation environment.

Keywords: air traffic flow management; scheduling; sequencing; en-route traffic control; simulation

1. INTRODUCTION

With trade liberalization and globalization of air services, air traffic is boosted by strong growth of air transport over the past decades. The demand for use of limited resources in Air Traffic System (ATS) spurred by frequent airborne movements may increase beyond acceptable range. In the case that traffic volume exceeds capacity of the traffic system, delays and traffic congestion would arise (C. Gwiggner and S. Nagaoka, 2010). Air Traffic Flow Management (ATFM) is a problem of optimization of resource allocation to manage capacity-demand imbalances in airspace and airports (V. Tošic et al., 1995). Recently, the ATFM related research is focusing on: the optimization problem for the multiple airports network (D. Sun and A.M. Bayen, 2008, D. Bertsimas et al., 2011); and the predictive models with regards to stochastic capacity profiles, e.g. weather conditions (S. Wang, 2009, D. Bertsimas and S.S. Patterson, 2000). Such research at present reflects the capability of ATFM to handle complexity and uncertainty in a large-scale system. At the same time, the network-wide ATFM become more significant. The International Civil Aviation Organization (ICAO) encourages ATFM to establish collaborative decision-making between multiple operators (airports, airlines, air service provider and manufacturer) and multiple operational levels (local, regional and global) (ICAO, 2014). Flow management is strongly required to collaborate with flight plan from both strategic and operational perspectives (C. Gwiggner and S. Nagaoka, 2010). An example of the collaboration is flight scheduling, which is to decide when and where aircrafts will fly. This gives rise to a research problem: how the flow management strategies assist air traffic managers to plan and operate flight schedules?

Fight scheduling using traffic flow techniques has been developed in past research, which refers to the topic of "traffic flow scheduling". Such research on the optimal scheduling problem employs different approaches to formula the scheduling model, e.g. integer linear programming (J. Rios and K. Ross, 2010), mixed integer linear programming (A.D. Aspremont and L.E. Ghaoui, 2007), and mixed non-linear integer programming (S. Yan et al., 2007). On the other hand, the flow model have a common feature: the flow variables are discrete space–discrete time aggregated (D. Sun and A.M. Bayen, 2008). The scheduling is inherently to assign flows for the number of aircrafts that can fly within a certain link/sector at a given time. Such flows obtained cannot directly be applied in practice for flight planner. We need a method to discompose the aggregated flows into individual, time-spatial continued flights.

The disaggregation problem is solved by an algorithm that decomposes the flows into arc chains, each of which represents an aircraft's route (R. Ahujanti and J. Orlin, 1993). The idea is essentially to do a routing which generates paths for individual flights in accordance with flow assignments. This algorithm is also adapted by other researchers to solve the flow disaggregation problem (S. Yan and H.F. Young, 1996, C.H. Tang et al., 2008). At the same time, a study for rerouting management proposes a random heuristic method combined with Integer Programming Packing Formulation to generate non-aggregated individual flight paths (D. Bertsimas and S.S. Patterson, 2000). There is not a routing but also a scheduling process, so that the routes generated meet scheduling constraints, e.g. departure time is less than or equal to scheduled departure time. Although the target of disaggregation is achieved, these two methods mentioned above have limitations: 1. the solution is not unique due to the randomness of routing: it may result in different routes; 2. the solution may not be reliable because it merely considers the flow and OD constrains, where the operational aspects of aircrafts, e.g. separation, speed and sequence, are ignored.

Against the background, this paper is motivated. Here we focus on a flight scheduling problem for generating, adjusting and optimizing flight schedules in en-route space, which is implemented into two phases. Firstly, a scheduling algorithm is developed to generate a set of time-spatial continuous individual flights from aggregated aircraft flows. The second phase is concerned with schedule adjustment and optimization, for a special case: merging sequence. En-route traffic control

10.1016/j.ifacol.2016.07.059

is to maintain safe and ordered trajectory of aircrafts. In this sense, this paper is meaningful for providing flight planner optimal flow-based flight schedules, as well as assisting en-route traffic controller to maintain safe and ordered trajectory of traffic.

Compared with the mentioned methods for flow disaggregation, our scheduling method is enhanced with operational feasibility, i.e. how to allocate time separations and how to order the sequence for aircrafts flying within a link. Technological contributions include: solve the randomness resulted from path dispatching; and avoid the solution not unique through optimal techniques for merge sequencing.

The rest of this paper is organized as follows. Section 2 describes the scheduling algorithms for generating the flow-based schedules. Section 3 presents the part of scheduling adjustment and optimization, especially for merging sequence. Section 4 includes a simulation case with a series of scenarios for illustrating the effect of scheduling. Finally, conclusions are drawn in Section 5.

2. SCHEDULE GENERATION

2.1 Preliminaries and scheduling solution

The en-route traffic network can be represented by basic elements, i.e. nodes (airports and waypoints) and links (J. Härri et al., 2009). The flight schedule is scoped in the en-route airspace, excluding departure/arrival time slots on ground. Thus, the airports are simplified as nodes, referring to the origin/destination of a flight route.

The input for scheduling is traffic flow rate, which is assumed from a general flow model. Each flow rate represents the number of aircrafts that can fly within a link at a given time. The flow is directional, and defined as

$$\{F_{ij} = (j_1, j_2, j_3, f_{ij}) | i = 1,2,\cdots,n, j = 1,2,\cdots,m\}. \quad (1)$$

It means that the value f_{ij} of aircrafts coming from node , will fly over node N_{j_2} toward node N_{j_3} at time step T_i. Fig.1. gives an example for illustration: the left table lists the directions of all flow rates each with a format "from…via…to…", where the number in the table refers to the node ID, and the node with ID -1 refers to the airport; and the right table is a matrix of values of flow rates, where *j* is flow ID and *i* is time step ID.

F_{ij} ($i = 1,2,\cdots,n$ →; $j = 1,2,\cdots,m$ ↓)

j_1	j_2	j_3	f_{ij}				
-1	1	3	2	0	0	0	0
-1	2	3	3	0	0	0	0
1	3	4	0	2	0	0	0
2	3	4	0	3	0	0	0
3	4	5	0	0	2	0	0
3	4	6	0	0	3	0	0
			T_1	T_2	T_3	T_4	T_5

Fig. 1.Input flows.

The output is flight schedules, each of which includes the sequence of nodes along the aircraft's path also the time stamps at which the aircraft reaches these nodes. The scheduling is executed by two steps: firstly, an algorithm is used to dispatch aircrafts, i.e. to discompose aggregated flows into individual aircraft routes; and then another algorithm is used to allocate times for each aircraft passing its route, where schedules are achieved and time-separation between flying aircrafts is near-uniform distribution.

2.2 Algorithm for route dispatching

For each aircraft, it departs from its origin, flies over the air traffic network via certain waypoints and finally reaches the destination. Route dispatching is to pick up available aircrafts from aggregated flows, meanwhile dispatching them to corresponding routes, until all flow restrictions are satisfied. The key notations are listed as below:

- $P_k(o_k, d_k, f_k)$: The aircraft with origino_k, destination d_k, and the last node it flies over is f_k.
- $R_k = \{R_{k,1}, R_{k,2}, R_{k,3}, \cdots\}$: The route of P_k, where $R_{k,l}$ is a node it passes.
- Q_{j_2}: The queue on the node N_{j_2}, i.e. a waiting list to record the aircrafts available to be dispatched.
- D_{j_3}: the destinations of flights passing node N_{j_3}.

The flowchart for algorithm is described in Fig.2. The algorithm iteratively executes a dispatching process beginning from T_1 and moving forward to end of time steps. In the flowchart, a part high-lighted by dish lines refers to the one-time dispatching at T_i . To satisfy the flow rate $F_{ij} = (j_1, j_2, j_3, f_{ij})$, the system needs to dispatchf_{ij}aircrafts from the node N_{j_2} to the nodeN_{j_3}. All aircrafts will experience three states, i.e. taking-off, flying and landing, to finish the dispatching:

- $j_1 = -1$: aircrafts (f_{ij}) are generated from the node N_{j_2} , and joint the queue on the node N_{j_3} . Each aircrafts$P_k(o_k, d_k, f_k)$is initialized as $o_k = j_2, f_k = j_2, R_k = \{j_2\}$, i.e. it starts at the nodeN_{j_2}.
- $j_3 = -1$: aircrafts (f_{ij}) coming from the node N_{j_1} land on the node N_{j_2}. Each aircrafts$P_k(o_k, d_k, f_k)$is finalized as $R_k = R_k \cup \{d_k\}$, i.e. it reaches the destination N_{j_2}.
- $j_1 \neq -1, j_3 \neq -1$: aircrafts (f_{ij}) coming from the node N_{j_1} joint the queue on the node N_{j_3} . Each aircrafts $P_k(o_k, d_k, f_k)$ is updated as $f_k = j_2, R_k = R_k \cup \{j_2\}$, i.e. it passes the node N_{j_2}.

If $j_1 \neq -1, j_3 \neq -1$, suppose that the subset Q_{j_1,j_2,j_3} denotes the aircrafts $\{P_k(o_k, d_k, f_k)\} \subseteq Q_{j_2}$ that come from the node $N_{j_1}(f_k = j_1)$ and can fly to the node $N_{j_3}(d_k \in D_{j_3})$:

$$Q_{j_1,j_2,j_3} = \{P_k(o_k, d_k, f_k) \in Q_{j_2} | f_k = j_1, d_k \in D_{j_3}\}. \quad (2)$$

$\|Q_{j_1,j_2,j_3}\|$ is the number of aircrafts in Q_{j_1,j_2,j_3}. If $\|Q_{j_1,j_2,j_3}\| \geq f_{ij}$, the first f_{ij}aircrafts in Q_{j_1,j_2,j_3} will be dispatched from Q_{j_2} to Q_{j_3}.

Else $j_3 = -1$, suppose that the subset $Q_{j_1,j_2,-1}$ denotes the aircrafts$\{P_k(o_k, d_k, f_k)\} \subseteq Q_{j_2}$ that come from the node $N_{j_1}(f_k = j_1)$ with destination to the node N_{j_2} $(d_k = j_2)$:

$$Q_{j_1,j_2,-1} = \{P_k(o_k, d_k, f_k) \in Q_{j_2} | f_k = j_1, d_k = j_2\}. \quad (3)$$

$\|Q_{j_1,j_2,-1}\|$is the number of aircrafts in $Q_{j_1,j_2,-1}$.If $\|Q_{j_1,j_2,-1}\| \geq f_{ij}$, the first f_{ij}aircrafts in $Q_{j_1,j_2,-1}$ will be removed from Q_{j_2} and landed on the node N_{j_2}.

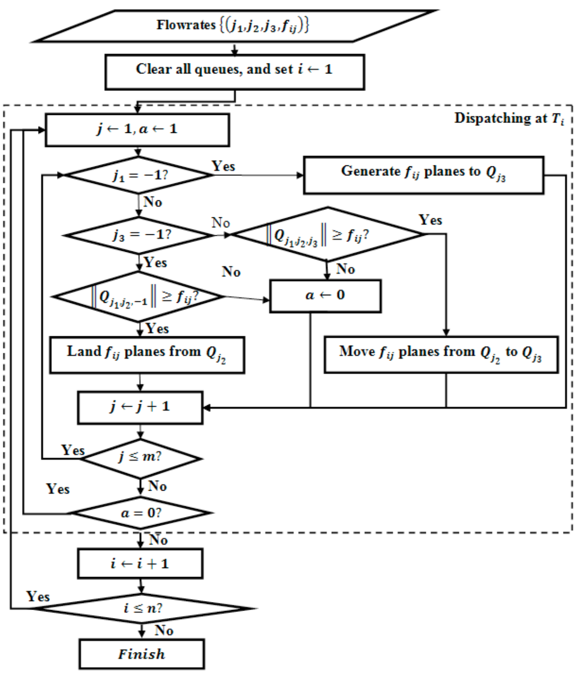


Fig. 2.Flowchart of dispatching algorithm.

Notice that, an initialization for Origin-Destination (OD) is needed, after an aircrafts P_k is generated (i.e. taking-off),

$$f_k = o_k, R_k = \{o_k\}. \quad (4)$$

Node N_{j_3} has a list of destinations D_{j_3}, each of which refers to the destination of a flight passing this node. In the flowchart, D_{j_3} is used for enabling aircrafts satisfy their OD constraints. For example, an aircrafts$P_k(o_k, d_k)$ can fly to the node N_{j_3} if and only if its destination is in D_{j_3}, i.e.$d_k \in D_{j_3}$. Using this way, the dispatching algorithm, compared with the previous method (D. Bertsimas and S.S. Patterson, 2000), eliminates the randomness of routing.

2.3 Algorithm for scheduling

After dispatched, each aircraft P_k has a route $R_k = \{R_{k,1}, R_{k,2}, R_{k,3}, \cdots, R_{k,n_k}\}$ representing n_k nodes the flight passed. The time for$P_k$passes the l-th node $N_{R_{k,l}}$on its route is $\tau_{k,l}$. Scheduling is to decide the a time sequence τ_k, which is assigned with

$$\tau_k = \{\tau_{k,1}, \tau_{k,2}, \tau_{k,3}, \cdots, \tau_{k,n_k}\}. \quad (5)$$

It is known that P_k passes $N_{R_{k,l}}$ due to a flowrate $F_{ij} = (j_1, R_{k,l}, j_3, f_{ij})$ at time T_i. Denote $\lambda_{i,R_{k,l}}$ as the summary of all flowrates via$N_{R_{k,l}}$ during the time interval $[T_i, T_{i+1}]$ as

$$\lambda_{i,R_{k,l}} = \sum_{(j_1,R_{k,l},j_3,f_{ij})} f_{ij}. \quad (6)$$

Thus, there are $\lambda_{i,R_{k,l}}$aircrafts pass the node $N_{R_{k,l}}$ during the time interval $[T_i, T_{i+1}]$, and all these $\lambda_{i,R_{k,l}}$ aircrafts are in the queue $Q_{R_{k,l}}$ at the time T_i. The target of the scheduling is to control these $\lambda_{i,R_{k,l}}$ aircrafts passing the node $N_{R_{k,l}}$ with a uniform time separation. In that way, the aircrafts of same type flying on the same level can keep uniform distribution with each other. Such separation is good for safety control, as well as to avoid traffic congestion. However, this is a quite ideal situation and is hardly achieved in reality. Alternatively, a near-uniform separation calculation is proposed as follows.

A time separation is determined as

$$S_{i,R_{k,l}} = \frac{T_{i+1} - T_i}{2\lambda_{i,R_{k,l}}}. \quad (7)$$

The x-thaircraft passes the node $N_{R_{k,l}}$ at time

$$T_i + (2x - 1)S_{i,R_{k,l}}. \quad (8)$$

Referring to Fig.3, we have

- A time separation $S_{i,R_{k,l}}$ is added between the first aircraft and the aircraft ahead,
- A time separation $S_{i,R_{k,l}}$ is added between the last aircraft and the aircraft behind, and
- A time separation $2S_{i,R_{k,l}}$ is set between two aircrafts.

Notice that in Fig.3

- During time interval $[T_{i-1}, T_i]$, $\lambda_{i-1,R_{k,l}}$ aircrafts pass the node $N_{R_{k,l}}$, and the last time separation is $S_{i-1,R_{k,l}}$.
- During time interval $[T_i, T_{i+1}]$, $\lambda_{i,R_{k,l}}$aircrafts pass the node $N_{R_{k,l}}$, and the last/first time separation is $S_{i,R_{k,l}}$.
- During time interval $[T_{i+1}, T_{i+2}]$, $\lambda_{i+1,R_{k,l}}$ aircrafts pass the node $N_{R_{k,l}}$, and the first time separation is $S_{i+1,R_{k,l}}$.

As such, during time interval $[T_i, T_{i+1}]$,

- The time separation between the first aircraft and the aircraft ahead is $S_{i-1,R_{k,l}} + S_{i,R_{k,l}}$, and
- The time separation between the last aircraft and the aircraft behind is $S_{i,R_{k,l}} + S_{i+1,R_{k,l}}$.

Considering a special case, where $\lambda_{i-1,R_{k,l}} = \lambda_{i,R_{k,l}} = \lambda_{i+1,R_{k,l}}$, we have $S_{i-1,R_{k,l}} = S_{i,R_{k,l}} = S_{i+1,R_{k,l}}$. It means that all aircrafts during time interval $[T_{i-1}, T_{i+2}]$ have the same time separation.In this case, the uniform distribution is achieved.

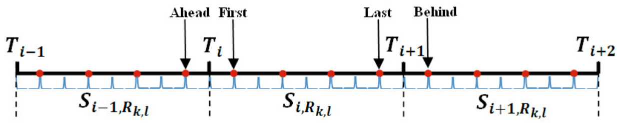


Fig. 3.Near-uniform time separation.

The travelling speed for the airplane P_k from node $N_{R_{k,l}}$ to node $N_{R_{k,l+1}}$can be calculated by

$$v_{k,l} = \frac{\|N_{R_{k,l+1}} - N_{R_{k,l}}\|}{\tau_{R_{k,l+1}} - \tau_{R_{k,l}}}. \quad (9)$$

Where $\|N_{R_{k,l+1}} - N_{R_{k,l}}\|$is the distance between the two nodes $N_{R_{k,l}}$ and $N_{R_{k,l+1}}$. This speed gives a reference for traffic controller. With proper speed control, the aircrafts can have a near-uniform distribution when flying within each link.

3. SCHEDULE ADJUSTMENT AND OPTIMIZATION

In Section 2, we obtain time-spatial continuous flight schedules with constrains on flows (number) and separation (position). Nevertheless, we cannot guarantee that the trajectory of aircrafts flying within in a certain region is ordered, based on the generated schedules. Considering a case shown in Fig.4, two streams of aircrafts are merging to a downstream link. Current scheduling algorithms will randomly pick up aircrafts into the merging sequence. It means that the order of aircrafts after merging is not unique. In practice, such flight planning is not reliable for safe en-route traffic control, because it may make aircrafts loss of track(K.R. Baker and D. Trietsch, 2013).

It is universally agreed that the sequencing solution should be reasonable and fair, e.g. using FCFS. However, this idea is less to address an optimal perspective in terms of throughput. Therefore, we provide an approach combining with fairness consideration with optimization constrains, to enable ordered trajectories of aircrafts in the scheduling process.

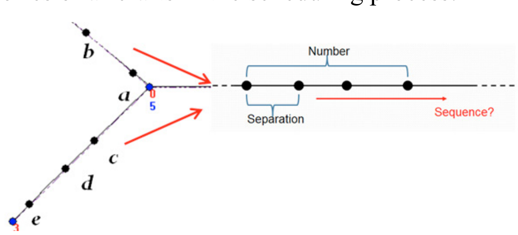


Fig. 4.Merging sequence.

3.1 Flight priority

During the time interval $[T_i, T_{i+1}]$, $\lambda_{i,R_{k,l}}$aircrafts pass the node $N_{R_{k,l}}$ following an order $O_{i,R_{k,l}}$. Denote $O_{i,R_{k,l}}(k)$ is the order of aircraft P_k in $O_{i,R_{k,l}}$, then

$$\tau_{k,l} = T_i + (2O_{i,R_{k,l}}(k) - 1)S_{i,R_{k,l}}. \quad (10)$$

Thus, the time forP_k passes the l-th node $N_{R_{k,l}}$on its route depends on the order of aircraftP_k in $O_{i,R_{k,l}}$.

The order $O_{i,R_{k,l}}$ can be set as flight priority. For example, a aircraftP_k with the highest flight priority, i.e. $O_{i,R_{k,l}}(k) = 1$, then

$$\tau_{k,l} = T_i + S_{i,R_{k,l}}. \quad (11)$$

During the scheduling process, this aircraft will always be the first one to pass a merging node. Obviously, its travelling time will be the minimal.

3.2 Optimization for multiple aircrafts

The travelling time for a aircraft P_k in the system is denoted as$\|\tau_k\| = \tau_{k,n_k} - \tau_{k,1}$. Suppose $K = \{k_1, k_2, \cdots\}$ are indices for aircrafts with higher priorities. Here we aim to reduce the average travelling time of those aircrafts. It can be achieved by solving the following objective function,

$$\min_{O_{i,j}} \sum_{k \in K} \|\tau_k\|, \quad (12)$$

Where $O_{i,j}$ are aircrafts' queue order on the merging node N_j at time step T_i.

In practice, aircrafts may be required to fly over a certain region as fast as possible, e.g. military zones. Under this consideration, we attempt to further optimize the average flight time within a certain region. Suppose L is a local area, and N_L are indices for all the merging nodes in this local regions, and $K = \{k_1, k_2, \cdots\}$are indices for aircrafts with higher priorities. We aim to reduce the travelling time for passing this region.

The time for P_k passes the l-th node $N_{R_{k,l}}$on its route is $\tau_{k,l}$. Thus, its travelling time in the local region is $\tau_{k,l_1} - \tau_{k,l_2}$, with $R_{k,l_1}, R_{k,l_2} \in N_L, R_{k,l_1+1}, R_{k,l_2-1} \notin N_L$.

The local optimization can be achieved by solving the following objective function,

$$\min_{\substack{O_{i,j} \\ j \in N_L}} \sum_{\substack{k \in K \\ R_{k,l_1}, R_{k,l_2} \in N_L \\ R_{k,l_1+1}, R_{k,l_2-1} \notin N_L}} (\tau_{k,l_1} - \tau_{k,l_2}). \quad (13)$$

4. SIMULATION

We have developed a simulation system to implement flight scheduling, using Microsoft Visual Studio 2008 under Windows 7 environment. It is an integrated multi-functional simulation platform, consisting of the graphic display module, the data processing module, the simulation control module, the algorithm module. The system supports real-time simulation which can be run on advanced microcomputers or graphic workstations. Currently, we are utilizing this system for air traffic simulation for the Association of Southeast Asian Nations (ASEAN).

In this section, we give a simulation case for an en-route traffic space (10 airports with 26 waypoints)for a region of ASEAN. All flights take off from Changi Airport in Singapore, and fly to other 9 airports, i.e. Kuala Lumpur, Penang, Phuket, Bangkok, Ho Chi Minh City, Kota Kinabalu, Kuching, Jakarta and Surbaya. The scheduling method is demonstrated in the following three scenarios.

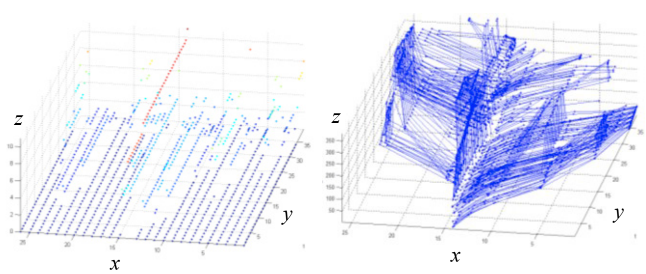
4.1 Scenario 1: visualized individual flights

This scenario is to present a normal scheduling process, i.e. how to obtain individual flights from aggregated flows. Input flows are generated from a flow model establish by our research team. Based on those data, the scheduling algorithm takes 120 milliseconds to generate 381 flights for 4 hours' simulation time. Table 1 lists the configuration of the computer and the scheduling performance.

Table 1. Configuration and performance

Computer	HP workstation xw4600
Operation system	Windows 7 64bits
CPU	Intel® Core™2 Quad CPU Q9300 @ 2.50 GHz
RAM	4GB
Scheduling time	120 milliseconds

The inputs and outputs are visualized in the terms of 3D chart in Fig.5. We select data based on a partial map (26 nodes) within 39 time steps for illustration, considering the complete time-space data is huge. As shown in Fig.5(a), flows are aggregated in discrete space (x-axis) discrete time (y-axis), where z-axis refers to the value of flow rates. After scheduling, the disaggregation problem is effectively solved. As shown in Fig.5(b), each line represents an individual flight (z-axis) which is continued time (x-axis) and continued space (y-axis).



(a) (b)

Fig. 5.Discrete flows to continued flights.

4.2 Scenario 2: rescheduling with capacity change

This scenario is to present a rescheduling process under a stochastic capacity profile. Due to weather conditions, the capacity of a link is reduced to 50%. Table 2 lists a summary of simulation results. In total 231aircrafts are affected with either delay or route change. As shown Fig.6, there is an overview of changed trajectories after rescheduling. The aircrafts highlighted in purple colour refer to the rerouted aircrafts; and the ones in red colour are delayed. A more detailed picture is shown at the lower left corner. We can see that all aircrafts can be tracked, and moves with near-uniform separations with each other.

The system executes the rescheduling calculations in the background, paralleling with display of dynamic scene. The high efficiency provides the capability to real-time analyse of the impact of flow management actions.

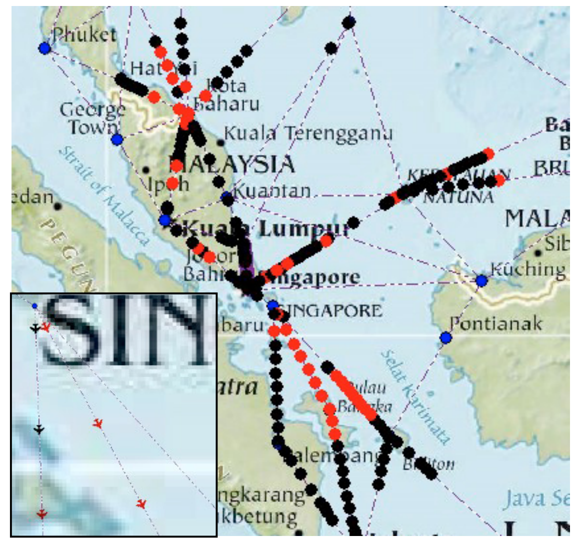


Fig. 6.Changes in flight trajectories after rescheduling.

Table 2.Simulation results

Simulation time period	4 hours
Total number of aircrafts generated	381
Number of aircrafts changed route	50
Number of delayed aircrafts	147
Average delay for delayed aircrafts	878 seconds

4.3 Scenario 3: sensitively analyses for flight priority

This scenario is to present a reasonable change of merging sequence when the priority of a certain flight is changed during the rescheduling process. An example is show in Fig.7. In Scenario 2, the aircraft (ID 62 in red) was delayed due to its low priority during rescheduling (Fig.7(a)). After merged into the downstream link, the aircraft is sequenced at last end of other aircrafts, with the merging sequence: 62-7-72 (Fig.7(b)). Now the aircraft is assigned with a higher priority. It is not delayed (Fig.7(c)), and sequenced in the head of others with the new merging sequence is 7-72-62 (Fig.7(d)).

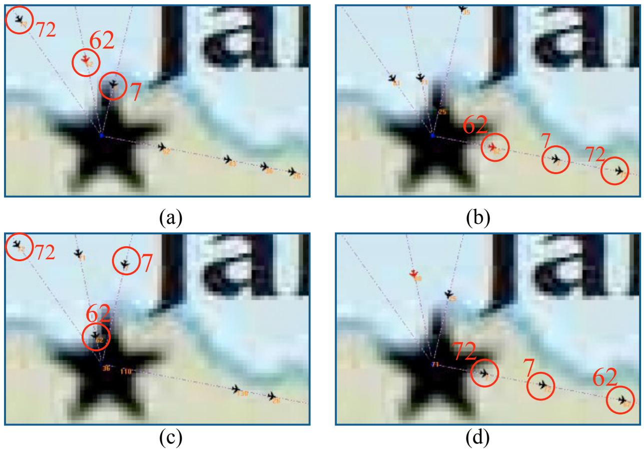


Fig. 7.Merging sequencing with flight priority.

5. CONCLUSIONS

A systematic scheduling framework is provided with capability of generating, adjusting and optimizing flight schedules in en-route airspace. It enhanced collaboration on en-route traffic control problems with flow-based flight planning. On one hand, it assists flight planner by providing flight schedules reliable on optimal flow restrictions; On the other hand, it ensures en-route air traffic controller by maintaining safe and ordered trajectory of traffic. This paper addresses an attempt to link strategic flow management with operational traffic control, which is encouraged by current ATFM in Collaborative Decision Making (CDM) process where multiple levels participate in flight decisions in the strategic, pre-tactical, and tactical phases (C.L. Wu and R.E. Caves, 2002).

The key achievement is to design algorithms to generate individual flights from aggregated flows, while enabling aircrafts keep reasonable separations and sequence. Despite flow-based, this scheduling method is independent of patterns of flow models, and thus supports a general problem of flow disaggregation. The sequencing problem is simply discussed and keeps some potential for further study. Simulation experiments demonstrate feasibility and reliability of the scheduling method, also proves a capacity of high-efficient computing, with fast response for real-time rescheduling.

REFERENCES

Ahujanti, R. & Orlin, J. (1993). *Network flow: theory, algorithms, and applications*, New Jersey: Prentice Hall.

Aspremont, A.D. & Ghaoui, L.E. (2007). A semidefinite relaxation for air traffic flow scheduling.*IEEE International Conference on Research, Innovation and Vision for the Future*. 103-107.

Baker, K.R. & Trietsch, D. (2013). *Principles of sequencing and scheduling*, John Wiley & Sons.

Bertsimas, D., Lulli, G. & Odoni, A. (2011). An integer optimization approach to large-scale air traffic flow management. *Operations research,* 59**,** 211-227.

Bertsimas, D. & Patterson, S.S. (2000). The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach. *Transportation Science,* 34**,** 239-255.

Gwiggner, C. & Nagaoka, S. (2010). Recent models in the analysis of air traffic flow. *Electronic Navigation Research Institute (ENRI). Tokyo*.

Härri, J., Filali, F. & Bonnet, C. (2009). Mobility models for vehicular ad hoc networks: a survey and taxonomy. *Communications Surveys & Tutorials, IEEE,* 11**,** 19-41.

ICAO (2014). Manual on Collaborative Air Traffic Flow Management. *In:* ORGANIZATION, T.I.C.A. (ed.).

Rios, J. & Ross, K. (2010). Massively parallel dantzig-wolfe decomposition applied to traffic flow scheduling. *Journal of Aerospace Computing, Information, and Communication,* 7**,** 32-45.

Sun, D. & Bayen, A.M. (2008). Multicommodity Eulerian-Lagrangian large-capacity cell transmission model for en route traffic. *Journal of guidance, control, and dynamics,* 31**,** 616-628.

Tang, C.H., Yan, S. & Chen, Y.H. (2008). An integrated model and solution algorithms for passenger, cargo, and combi flight scheduling. *Transportation Research Part E: Logistics and Transportation Review,* 44**,** 1004-1024.

Tošic, V., Babic, O., Cangalovic, M. & Hohlacov, D. (1995). A model to solve en route air traffic flow management problem: A temporal and spatial case. *unpublished report, University of Belgrade*.

Wang, S. (2009). Area Air Traffic Flow Optimal Scheduling Under Uncertain Weather.*International Conference on Computational Intelligence and Software Engineering*. 1-4.

Wu, C.L. & Caves, R.E. (2002). Research review of air traffic management. *Transport Reviews,* 22**,** 115-132.

Yan, S., Tang, C.H. & Lee, M.C. (2007). A flight scheduling model for Taiwan airlines under market competitions. *Omega,* 35**,** 61-74.

Yan, S. & Young, H.F. (1996). A decision support framework for multi-fleet routing and multi-stop flight scheduling. *Transportation Research Part A: Policy and Practice,* 30**,** 379-398.