Distributed Flight Routing and Scheduling for Air Traffic Flow Management

Yicheng Zhang, *Student Member, IEEE*, Rong Su, *Senior Member, IEEE*, Qing Li, Christos G. Cassandras, *Fellow, IEEE*, and Lihua Xie, *Fellow, IEEE*

***Abstract*— Air traffic flow management (ATFM) is an important component in an air traffic control system and has significant effects on the safety and efficiency of air transportation. In this paper, we propose a distributed ATFM strategy to minimize the airport departure and arrival schedule deviations. The scheduling problem is formulated based on an en-route air traffic system model consisting of air routes, waypoints, and airports. A cell transmission flow dynamic model is adopted to describe the system dynamics under safety related constraints, such as the capacities of air routes and airports, and the aircraft speed limits. Our ATFM problem is formulated as an integer quadratic programming problem. To overcome the computational complexity associated with this problem, we first solve a relaxed quadratic programming problem by a distributed approach based on Lagrangian relaxation. Then a heuristic forward-backward propagation algorithm is proposed to obtain the final integer solution. Experimental results demonstrate the effectiveness of the proposed scheduling strategy.**

***Index Terms*— Air traffic flow management, Lagrangian relaxation, subgradient method, forward-backward propagation.**

I. INTRODUCTION

AIR traffic delays have been costing billions of dollars to airlines each year [1]. The situation is expected to become even worse, if no satisfactory solution can be found soon. Due to limited space for further infrastructural expansion, improving efficiency of air traffic management becomes critical for the aviation industry to cope with the expected demand surge. In this paper we aim at improving efficiency of air traffic flow management (ATFM) by reducing the arrival and departure schedule deviations in the air traffic system.

There are many techniques proposed in the literature and applied in the current practice [2], [3]. In [4] and [5] an ATFM strategy to optimize the airport capacity utilization is formulated as a mixed integer linear programming problem to alleviate the consequences of congestion. However, the proposed formulation is only for ground delays without consideration of the airborne delays. In [6], the proposed control algorithm achieves a global optimum in the sense of eliminating airborne delays. In [7] an air traffic flow management strategy for the ground-holding policy problem is proposed and a minimum cost flow algorithm is adopted. This strategy is only defined for a simplified single destination airport scenario and is not applicable for multiple-origins multiple-destinations problems. In [8], [9], and [10] similar ATFM models to handle both ground delays and airborne delays are proposed with an Integer Program (IP) formulation. These models provide a complete representation of all phases of each flight and expeditious aircraft movement. The distinctive feature of the model is that it allows rerouting decisions. However, as these models only have constraints on the minimal travelling time in a sector, they do not provide any information on the maximal travelling time or the flight trajectories.

In terms of modelling, there have been several major types proposed in the literature, see a detailed comparison given in [11]. Lagrangian models [12], [13] are used to describe flight trajectories of individual aircraft, which is usually computationally infeasible for a large network. Aggregate traffic models and Eulerian models [14], [15] are used to describe average behaviours of a group of aircraft, as described by the concept of *flows*. An aggregation approach typically provides a lower order, fixed-resolution model of the airspace, while the Eulerian approach provides a flexible resolution model [16]. A multi-commodity Eulerian-Lagrangian large-capacity cell transmission model for en-route air traffic is proposed in [17], which adopts the basic idea of cell transmission models described in [18] and [19] within a standard multi-commodity flow model equipped with origin-destination (OD) pairs to avoid difficulties in describing flow merging and diverging.

In this paper we adopt an Eulerian-Lagrangian model similar to [17], but with a richer set of constraint types than other existing flow models such as [17], [20], and [10]. For example, we consider a variety of aircraft types such as small, medium and large passenger/cargo aircraft and unmanned aerial vehicles (UAVs) that are expected to be more popular in the near future, and take the aircraft speed lower and upper limits into account when applying air route capacity constraints on air route volume dynamics. We formulate our ATFM problem as an integer quadratic programming (IQP) problem owing to the integer values of the decision variables of air route shifts within each discrete time interval, and aim to minimize the total airport departure and arrival schedule deviations

Manuscript received May 5, 2016; revised October 21, 2016 and January 2, 2017; accepted January 14, 2017. Date of publication February 14, 2017; date of current version October 3, 2017. This work was supported by the ATMRI-CAAS Project on Air Traffic Flow Management under Project ATMRI:2014-R7-SU. The Associate Editor for this paper was Z.-H. Mao.

Y. Zhang, R. Su, and L. Xie are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: yzhang088@e.ntu.edu.sg; rsu@ntu.edu.sg; elhxie@ntu.edu.sg).

Q. Li is with the Air Traffic Management Research Institute, Nanyang Technological University, Singapore 639798 (e-mail: liqing@ntu.edu.sg).

C. G. Cassandras is with the Division of Systems Engineering, Boston University, Brookline, MA 02446 USA (e-mail: cgc@bu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2017.2657550

in an air traffic network. To overcome the exponential-time complexity in IQP, we first relax it into a standard convex QP, which, although having a polynomial-time complexity [21], requires a distributed scheduling strategy based on Lagrangian relaxation [22] and the subgradient method [23], aiming for a good trade-off between the quality of scheduling and the computational complexity. After solving the relaxed QP problem, we then propose a novel heuristic forward-backward propagation algorithm to achieve integer solutions. Compared with existing flow-based ATFM approaches, we have made the following contributions: (1) an IQP ATFM formulation with an Eulerian-Lagragian flow model and more types of constraints, (2) a Lagrangian Relaxation based distributed optimization approach to solve a QP-relaxed ATFM problem, and (3) a heuristic forward-backward propagation algorithm to obtain an integer solution.

The remainder of the paper is organized as follows. An ATFM problem is formulated as an IQP problem in Section II. A distributed air traffic flow routing and scheduling strategy is proposed in Section III, which solves a QP-relaxed problem. A heuristic propagation algorithm is presented in Section IV, which aims to derive an integer solution to the original IQP problem. Experimental results are shown in Section V. Conclusions are drawn in Section VI.

II. An ATFM Problem Formulation

A. Introduction of an En-Route Air Traffic Network

We focus on an en-route part of an air traffic network, which consists of (the descending and ascending part of) airports and pre-defined air routes within concerned sectors. Currently, there are two different types of flight navigation systems: Required Navigation Performance (RNP) and Area Navigation (or Random Navigation (RNAV)). Although the latter is cheaper and possibly more flexible for airlines, it imposes a major safety concern, especially over spaces which lack of sufficient ground radar coverage. Motivated by the fact that the RNP may eventually become the dominant one for the aviation industry [24] owing to a major safety concern about RNAV, we consider an air traffic network which consists of pre-defined routes. Each air route consists of a sequence of waypoints (or control points), and any two consecutive waypoints are connected by one *link*, which is one segment of an air route. A waypoint is a reference point in physical space used for the purpose of radar guided navigation. Each airport is simplified as a set of one departure link, one arrival link, and a holding link. We do not consider any ground operations in this paper, but simply assume that the departure and arrival handling capacities in each concerned airport are known in advance. An illustration for an en-route air traffic network is depicted as a directed graph shown in Fig. 1. Although in the picture there is at most one directed link between any two nodes (i.e., waypoints), it is possible to use two links with opposite directions to denote one air route allowing bi-directional flights but with safe vertical separations.

It is interesting to note that an air traffic management system is usually distributed by nature, where multiple Area Control Centers (ACCs) running in parallel to control individual aircraft in their own responsible airspace (usually defined as Flight Information Regions (FIRs)), whereas communicating with other neighbouring ACCs to relay important messages. This natural distributed infrastructure will later facilitate our distributed flight routing and scheduling strategy to lower the computational burden for real-time operations.

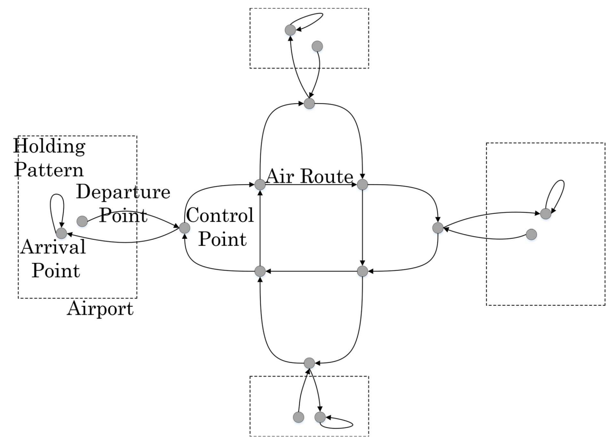


Fig. 1. Simplified system model.

B. Air Route Segmentation

Aircraft in the same sector must be well separated from each other to ensure safety. Each aircraft must keep a safe distance from other aircraft ahead, above, under or aside. In real-world applications, the vertical separation is implemented by introducing flight levels. Aircraft with different cruising speeds will take different flight levels. Aircraft at the same flight level have similar speeds. The current practice usually enforces a separation of 5-50 nautical miles depending on the actual flying space, e.g., over land with well radar coverage or over sea with little radar coverage. This separation will result in specific link capacities. In this paper we assume that the separation distance in each link is known and fixed in advance. We adopt a discrete-time cell transmission link dynamic model in this framework. Within a pre-chosen sampling period Δ, the distance L_C that an aircraft is able to cover is determined by the minimum acceptable cruising speed (denoted as $\underline{S}$) and the maximum cruising speed (denoted as $\overline{S}$), i.e.,

$$\underline{S}\Delta \leq L_C \leq \overline{S}\Delta. \tag{1}$$

In this work we choose L_C as the distance that an aircraft can cover within Δ with an economic cruising speed, which is between $\underline{S}$ and $\overline{S}$, and call L_C one *link segment*. For different types of aircraft with different cruising speeds, their segment lengths are different. In the current practice flights with significantly different cruising speeds should fly at different levels, and it is preferrable for aircraft not to change their flight levels after they are assigned. In this paper we simplify the setting further by making the following assumption.

Assumption 1: Each type of aircraft has a pre-determined set of flight levels, and each flight level will only be assigned to aircraft with the same heading and similar cruising speeds.

In other words, the segmentation of each flight level in each link is uniquely pre-determined. With this assumption

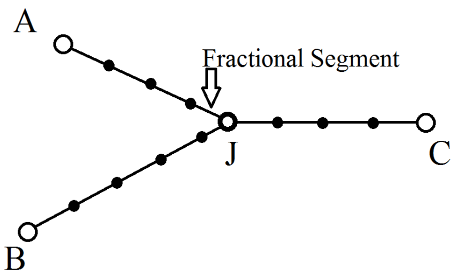


Fig. 2. Fractional Segments in air-routes.

we can treat each flight level of a link as one link between two corresponding waypoints, and each type of aircraft can only access some of those outgoing links at each waypoint, which match the aircraft's pre-determined accessible flight levels. From now on we do not explicitly mention flight levels but only links. Each link is partitioned into a set of whole segments with possibly one fractional segment, which covers two consecutive link levels, as shown in Fig 2, where the link level $A - J$ consists of three segments and a fractional segment, whereas the link level $B - J$ has four segments and a fractional segment.

Since we use a cell transmission model, it is not possible for us to track every single aircraft in the flow. So the choice of Δ needs to ensure the following assumption holds:

Assumption 2: All aircraft that fly into a segment during the period t will fly out of the segment during the period $t+1$.

This assumption of "aircraft hopping among segments" essentially rules out the possibility of dealing with multiple types of aircraft with large speed differences in each link level, which fortunately holds owing to Assumption 1. For a whole segment, all the airplanes enter this segment at time t will move out at time $t+1$. However, for a fractional segment, the airplanes entering it at time t may not be able to stay in this segment until time $t+1$ but hop from this segment to the next connected segment in the same period.

To estimate the flow entering and exiting a fractional segment, the aircraft are required to be uniformly distributed in each segment. This assumption is roughly true when the length of segment is short enough. In our model, the segment length is related to the sampling time, thus by adjusting the sampling time we can achieve this assumption via a suitable air traffic control strategy, which is nevertheless outside the scope of this paper. With this assumption of uniform distribution, the number of aircraft leaving one whole segment i and entering/traversing another whole segment j via a fractional segment k during the interval t can be calculated as follows.

$$f_{ij}(t) = \left\lceil \frac{L_C - L_k}{L_C} f_{ik}(t) \right\rceil, \tag{2}$$

where $\lceil\cdot\rceil$ denotes the ceiling function, whose value is the smallest integer value larger than the argument. From now on we call $f_{ij}(t)$ a *shift* of aircraft in t. Clearly, $f_{ij}(t)$ must be an integer number within each interval Δ. The motivation behind equation (2) can be explained as follows. When aircraft from the segment i traverse the fractional segment k and move into the whole segment j, those pilots will not know that the segment k is a fractional segment. So they still consider the segment k as a whole segment and want to ensure that the uniform distribution assumption hold within one virtual whole segment (during one time interval Δ), which covers the entire segment k and part of the segment j. That's why in equation (2) we have the expression $(L_C - L_k)/L_C$, which describes the percentage of aircraft which fly into the segment j, as the ratio of the part of that virtual whole segment overlapping the segment j (i.e., $L_C - L_k$) and the virtual whole segment L_C. We assume that after segmentation, no two fractional segments are neighbours, which can be easily satisfied by having a partial segment only at the end of each link. Equation (2) is equivalent to the following inequalities:

$$\begin{aligned} f_{ij}(t) &\geq \frac{L_C - L_k}{L_C} f_{ik}(t), \\ f_{ij}(t) &\leq \frac{L_C - L_k}{L_C} f_{ik}(t) + 1, \\ f_{ij}(t) &\in \mathbb{N}. \end{aligned}$$

The fractional segment model is introduced to obtain reasonable partitions in each air link. However, it may lead to a high concentration of aircraft at the entrance of the downstream segment next to these fractional segments, i.e., the merging of aircraft temporarily create non-uniform distribution in the downstream segment. To ensure that this temporary concentration of aircraft will not cause any problem for the air traffic control part, which is based on the output of flow management, we impose a constraint that the density at the entrance of the downstream segment should not surpass the maximum density of the downstream segment. More explicitly, assume the capacity of segment i is C_i, the length of this segment is L_i. Then the maximal number of flights that could possibly enter this segment during time interval t is C_i and the maximal flight density ρ_i to pass the entrance of this segment is defined as $\rho_i = \frac{C_i}{L_i}$. To ensure the density at the entrance of the downstream segment should not surpass its maximal density during each time t, the incoming flights from the upstream segments should satisfy the constraints below,

$$\sum_{j \in U_i \cup U_{i,F}} \frac{f_{ji}(t)}{L_C - L_{ji}} \leq \frac{C_i}{L_i}, \tag{3}$$

where U_i is the set of all upstream segments directly connected with the segment i, $U_{i,F}$ is the set of all upstream whole segments connecting to the segment i via some fractional segments, and L_{ji} is the length of the fractional segment connecting the whole segment j and the whole segment i. For $j \in U_i$, $L_{ji} := 0$, because there is no fractional segment between segment j and segment i.

Comparing with the air traffic system model description proposed in [10], our model description has some advantages which make this model more realistic for an en-route air traffic system. Firstly, our model takes the minimum flight speed into consideration which is not considered in [10]. The minimal speed is an important characteristic to ensure the flight safety. Secondly, we consider RNP navigation, which allows all concerned aircraft to fly through a fixed en-route network. In contrast, [10] considers sectors and only the entrance and exit of the sectors are fixed, i.e., RNAV navigation is adopted,

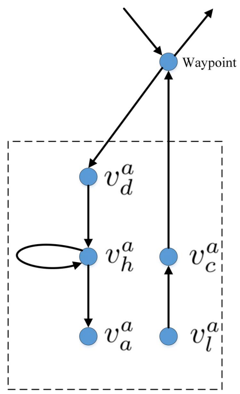


Fig. 3. The airport model.

allowing aircraft to fly arbitrarily in each sector, instead of following a fixed en-routed network. Thirdly, the flow merging shown in Fig. 2 is treated more realistically in our paper.

C. Statement of Air Traffic Flow Routing and Scheduling Problem

1) Notations: Before we describe our air flow routing and scheduling problem, some necessary notations are listed below.

- $\mathbb{N}, \mathbb{R}^+$ – The sets of natural numbers and non-negative real numbers, respectively.
- $G = (V, E)$ – The directed graph is to describe the air traffic network, where vertices and edges denote waypoints (together with all segmentation points) and air links, respectively. From now on, we will use link and segment interchangeably.
- $\mathcal{A}$ – The set of all concerned airports, where each airport $a \in \mathcal{A}$ consists of five waypoints as shown in Fig 3, i.e., $v_d^a \in V$ – the first descending fix, $v_h^a \in V$ – the aggregated entrance of holding pattern, $v_a^a \in V$ – the first approaching fix, $v_l^a \in V$ – the last departure fix, $v_c^a \in V$ – the last climbing fix.
  There is a self-loop at v_h^h denoting aircraft circulating in the holding pattern, which creates arrival delays. No link connects v_a^a and v_l^a because we do not consider ground operations between aircraft arrival and departure, and treat v_a^a as a sink of the network and v_l^a as a source of the network. For notational simplicity, we use $f_{in}^{a,P}(\phi, t)$ to denote the number of incoming (or arriving) (ϕ, P)-aircraft, i.e., aircraft of the type ϕ with the OD (origin - destination) pair P, via the link (v_h^a, v_a^a) at t, and $f_{out}^{a,P}(\phi, t)$ for the number of outgoing (or departing) (ϕ, P)-aircraft via the link (v_l^a, v_c^a) at interval t. For each airport, the number $r^{a,P}(\phi, t)$ of the scheduled arriving (ϕ, P)-aircraft via the link (v_h^a, v_a^a) and the number $s^{a,P}(\phi, t)$ of the scheduled departing (ϕ, P)-aircraft via the link (v_l^a, v_c^a) at t are assumed known.
- Φ – The set of aircraft types.
- $g : \Phi \times E \to 2^E$ – The assignment of feasible downstream links for each type of aircraft in each link.
- $i := (v, v') \in E \subseteq V \times V$ – denotes the directed air link from waypoint v to waypoint v'. Some parameters for each air link include the length, the capacity and the connections among each other.
  - L_i – The length of the link i.
  - $C_i(t)$ – The capacity of the link i at t.
  - $U_i(\phi) := \{j \in E | i \in g(\phi, j)\}$ – Upstream links connected with i at v for the type ϕ aircraft.
  - $D_i(\phi) := \{j \in E | j \in g(\phi, i)\}$ – Downstream links connected with i at v' for the type ϕ aircraft.
- $\underline{S}(\phi, i)$ – The lower speed limit of the type ϕ aircraft in the link i.
- $\overline{S}(\phi, i)$ – The upper speed limit of the type ϕ aircraft in the link i.
- Δ – The sampling interval.
- $H_p \subseteq \mathbb{N}$ – The set of labels of all discrete intervals. By default, $H_p := \{0, 1, 2, \cdots, |H_p|\}$.
- $\mathcal{C} \subseteq A \times A - \{(a, a), a \in A\}$ – The set of origin-destination (OD) pairs. For each $P = (a, a') \in \mathcal{C}$, let $P[1] = a$ and $P[2] = a'$.
- $N_i^P(\phi, t)$ – The volume (i.e., the number) of (ϕ, P)-aircraft of the link i at t.
- $f_{ij}^P(\phi, t)$ – The number (or the shift) of (ϕ, P)-aircraft leaving the link i and entering/traversing the link j at t. By convention, if $j \notin g(\phi, i)$, i.e., the type ϕ aircraft cannot access the link j after traversing the link i (owing to the flight level constraint mentioned above), then $f_{ij}^P(\phi, t) = 0$ for all $t \in H_p$.

2) Constraints: We consider the following constraints: the network dynamics constraints, the link capacity constraints, the shift limits constraints, which are described below.

C1 - Network Dynamics Constraints: The network dynamics describe the relationship between the air traffic shifts and the air link volumes. For all $t \in H_p$, $i \in E$, $\phi \in \Phi$ and $P \in \mathcal{C}$,

$$N_i^P(\phi, t+1) = N_i^P(\phi, t) + \Big[f_{i,in}^P(\phi, t) - f_{i,out}^P(\phi, t)\Big], \quad (4)$$

where $f_{i,in}^P(\phi, t)$ and $f_{i,out}^P(\phi, t)$ denote respectively the numbers of incoming and outgoing (ϕ, P)-aircraft in the link i during the time interval t. More explicitly,

$$f_{i,in}^P(\phi, t) = \sum_{j \in U_i(\phi) \cup U_{i,F}(\phi)} f_{ji}^P(\phi, t), \quad (5a)$$

$$f_{i,out}^P(\phi, t) = \sum_{k \in D_i(\phi) \cup D_{i,F}(\phi)} f_{ik}^P(\phi, t), \quad (5b)$$

where the sets $U_{i,F}$ and $D_{i,F}$ denote respectively all upstream whole segments and downstream whole segments connecting with the segment i via some fractional segments for the type ϕ aircraft.

C2 - Link Capacity Constraint: Owing to the head-and-tail separation requirement imposed on all aircraft in the network, each link (or segment) has its own capacity. In general, different links may have different separation requirements, captured by a variable $m_{sep}(i, t)$, where $i \in E$ and $t \in H_p$, which also suggests that the separation distance is time variant, owing to possibly the time variant weather conditions. For example, $m_{sep}(i, t)$=5 NM is common in en-route airspace, while $m_{sep}(i, t) = 3$ NM is common in terminal airspace at lower altitudes. This time variant separation distance function is assumed known in advance in this paper for a routing and scheduling purpose, from which the link capacity can be defined as follows:

$$C_i(t) = \frac{L_i}{m_{sep}(i,t)} \tag{6}$$

where $C_i(t)$ is the capacity of the link i at t and L_i is the length of the link i. The total number of aircraft in the link i should not be greater than the link capacity at each t, i.e.,

$$(\forall t \in H_p) \sum_{P\in\mathcal{C}} \sum_{\phi\in\Phi} N_i^P(\phi,t) \le C_i(t). \tag{7}$$

C3 - Link Shift Limit Constraints: With the constraints of the cruising speeds for different types of aircraft, the outgoing link shift should also be bounded by the product of the link density and the speed limits, i.e., for all $t \in H_p$, $i, j \in E$, $\phi \in \Phi$, $P \in \mathcal{C}$,

$$\begin{aligned} \frac{N_i^P(\phi,t)}{L_i}\underline{S}(\phi,i)\Delta \le & \sum_{j\in D_i+D_{i,F}} f_{ij}^P(\phi,t) \\ & \le \frac{N_i^P(\phi,t)}{L_i}\overline{S}(\phi,i)\Delta, \end{aligned} \tag{8}$$

where $\frac{N_i^P(\phi,t)}{L_i}$ denotes the link density of (ϕ, P)-aircraft with a uniform distribution. To ensure the existence of an integer solution to the link shift assignments, it is necessary that

$$\left\lceil \frac{N_i^P(\phi,t)}{L_i}\underline{S}(\phi,i)\Delta \right\rceil \le \left\lfloor \frac{N_i^P(\phi,t)}{L_i}\overline{S}(\phi,i)\Delta \right\rfloor,$$

meaning that one integer aircraft shift assignment is feasible.

C4 - Flight Density Limit Constraints: As mentioned in the network segmentation, the density of the merging air flows should be no bigger than the entrance density of the downstream link. Then we have the following:

$$\sum_{\phi\in\Phi,P\in\mathcal{C}} \sum_{j\in U_i(\phi)\cup U_{i,F}(\phi)} \frac{f_{ji}^P(\phi,t)}{L_C - L_{ji}} \le \frac{C_i(t)}{L_i}. \tag{9}$$

C5 - Airport Handling Capacity Constraints: In practice no aircraft can depart earlier than its scheduled departure time. For this reason, for each airport $a \in \mathcal{A}$ we have the following constraints about aircraft departure and arrival shifts in each airport. For all $t \in H_p$,

$$(\forall P \in \mathcal{C})(\forall \phi \in \Phi) \sum_{l=0}^{t} f_{out}^{a,P}(\phi,l) \le \sum_{l=0}^{t} s^{a,P}(\phi,t), \tag{10a}$$

$$\sum_{P\in\mathcal{C},\phi\in\Phi} f_{in}^{a,P}(\phi,t) \le C_{in}^a(t) \tag{10b}$$

$$\sum_{P\in\mathcal{C},\phi\in\Phi} f_{out}^{a,P}(\phi,t) \le C_{out}^a(t), \tag{10c}$$

where $C_{in}^a(t)$ and $C_{out}^a(t)$ are airport handling capacities at t for flight arrivals and departures, respectively, which are assumed to be known in advance. Constraint (10.a) states that the accumulated departure shift in each airport a at each time t should not surpass the scheduled ones. Constraints (10.b) and (10.c) state that the actual total arrival and departure shifts at each time t should not surpass the airport handling capability.

3) Objective Function: Our objective is to minimize the total deviation from the original arrival and departure schedules as well as the chances of landing in airports different from the originally planned ones. Based on the aforementioned notations, the objective function can be formulated as follows,

$$\begin{aligned} \min \sum_{t\in H_p,\phi\in\Phi,P\in\mathcal{C}} \sum_{a\in A} \Big\{ & \big[f_{in}^{a,P}(\phi,t) - r^{a,P}(\phi,t)\big]^2 \\ & + \big[f_{out}^{a,P}(\phi,t) - s^{a,P}(\phi,t)\big]^2 + M \sum_{P[2]\ne a} f_{in}^{a,P}(\phi,t) \Big\}, \end{aligned} \tag{11}$$

where the positive constant M in the last term is chosen to be very large, denoting the extremely high penalty on landing aircraft to an airport different from their originally planned destinations. The first and second terms in the cost function denote the total deviation from the original arrival and departure schedules, respectively.

We summarize what we have developed and state below the **Air Traffic Flow Routing and Scheduling Problem (ATFRSP)**:

$$\begin{aligned} \min \sum_{t\in H_p,\phi\in\Phi,P\in\mathcal{C}} \sum_{a\in A} \Big\{ & \big[f_{in}^{a,P}(\phi,t) - r^{a,P}(\phi,t)\big]^2 \\ & + \big[f_{out}^{a,P}(\phi,t) - s^{a,P}(\phi,t)\big]^2 + M \sum_{P[2]\ne a} f_{in}^{a,P}(\phi,t) \Big\} \end{aligned} \tag{12a}$$

subject to

$$N_i^P(\phi,t+1) = N_i^P(\phi,t) + \Big[f_{i,in}^P(\phi,t) - f_{i,out}^P(\phi,t)\Big] \tag{12b}$$

$$f_{i,in}^P(\phi,t) = \sum_{j\in U_i(\phi)\cup U_{i,F}(\phi)} f_{ji}^P(\phi,t) \tag{12c}$$

$$f_{i,out}^P(\phi,t) = \sum_{k\in D_i(\phi)\cup D_{i,F}(\phi)} f_{ik}^P(\phi,t) \tag{12d}$$

$$(\forall j \in U_{i,F}(\phi))\ f_{ji}^P(\phi,t) = \left\lceil \frac{L_C - L_k}{L_C} f_{jk}^P(\phi,t) \right\rceil$$

where k is the fractional segment connecting j and i (12e)

$$(\forall p \in D_{i,F}(\phi))\ f_{ip}^P(\phi,t) = \left\lceil \frac{L_C - L_q}{L_C} f_{iq}^P(\phi,t) \right\rceil$$

where q is the fractional segment connecting i and p (12f)

$$\sum_{\phi\in\Phi,P\in\mathcal{C}} \sum_{j\in U_i(\phi)\cup U_{i,F}(\phi)} \frac{f_{ji}^P(\phi,t)}{L_C - L_{ji}} \le \frac{C_i(t)}{L_i} \tag{12g}$$

$$\sum_{P\in\mathcal{P}} \sum_{\phi\in\Phi} N_i^P(\phi,t) \le C_i(t) \tag{12h}$$

$$\begin{aligned} \frac{N_i^P(\phi,t)\Delta}{L_i}\underline{S}(\phi,i) \le & \sum_{j\in D_i(\phi)\cup D_{i,F}(\phi)} f_{ij}^P(\phi,t) \\ & \le \frac{N_i^P(\phi,t)\Delta}{L_i}\overline{S}(\phi,i) \end{aligned} \tag{12i}$$

$$(\forall P \in \mathcal{C})(\forall \phi \in \Phi) \sum_{l=0}^{t} f_{out}^{a,P}(\phi,l) \le \sum_{l=0}^{t} s^{a,P}(\phi,t), \tag{12j}$$

$$\sum_{P\in\mathcal{C},\phi\in\Phi} f_{in}^{a,P}(\phi,t) \le C_{in}^{a}(t) \tag{12k}$$

$$\sum_{P\in\mathcal{C},\phi\in\Phi} f_{out}^{a,P}(\phi,t) \le C_{out}^{a}(t), \tag{12l}$$

$$N_i^P(\phi,t), \quad f_{ij}^P(\phi,t) \in \mathbb{N} \tag{12m}$$

In ATFRSP the control variables are those shifts of aircraft $f_{ij}(\phi,t)$ for $i,j \in E$ and $t \in H_p$. These shifts will allow air traffic controllers to instruct each individual aircraft how to adjust its speed in each time interval t. The ATFRSP is an integer quadratic programming (IQP) problem. Owing to the high complexity involved in solving this IQP problem, we will first relax it into a convex quadratic programming (QP) problem, which will be solved by a distributed algorithm based on Lagrangian relaxation, and then use a heuristic algorithm to obtain a final integer solution.

III. Distributed Flow Routing and Scheduling

We first relax all integer decision variables in the ATFRSP into real numbers. This converts the ATFRSP into a standard convex QP problem. We partition the whole air traffic network into sub-networks. Each airport or waypoint only belongs to one sub-network, and so do most air links, except for a few shared by two sub-networks. Formally speaking, we consider the network as a directed graph $G = (V, E)$, where the vertex set V contains one special node called *ext* denoting the *external* of the entire network, and the edge set is $E \subseteq V \times V - \{(ext, ext)\}$ denoting the set of all directed air links, i.e., each air link $(\upsilon, \upsilon') \in E$ represents an air traffic flow either from one waypoint υ to another waypoint υ', or from the external source $\upsilon = ext$ to a waypoint υ' (which represents an incoming boundary link), or from waypoint υ to the external source $\upsilon' = ext$ (which represents an outgoing boundary link). Let $\mathcal{S}$ be a partition of $\upsilon - \{ext\}$, i.e., each waypoint belongs to one sub-network, and let $\mathcal{L}(\mathcal{S})$ denote all air links belonging to $\mathcal{S}$. We now make the following modification to the network graph G: for each link $(\upsilon, \upsilon') \in E$ with $\upsilon \in S \in \mathcal{S}$, $\upsilon' \in S' \in \mathcal{S}$ and $S \neq S'$, we add a node $b_{\upsilon,\upsilon'}$ to V, which represents a boundary of S and S', and replace $(\upsilon, \upsilon') \in E$ by two new edges $(\upsilon, b_{\upsilon,\upsilon'})$ and $(b_{\upsilon,\upsilon'}, \upsilon')$, which denotes two disjoint air segments, whose union is the original link (υ, υ'), and $(\upsilon, b_{\upsilon,\upsilon'})$ is placed in $\mathcal{S}$ and $(b_{\upsilon,\upsilon'}, \upsilon')$ belongs to $\mathcal{S}'$. After the modification, let B be the collection of all such boundary nodes, and E' be the new edge set. Then the new network graph is $G' = (V' = V \cup B, E')$, where $E' \subseteq (V \cup B) \times (V \cup B) - (\{(ext, ext)\} \cup B \times B)$. For any two different sub-networks, they can only share some boundary points in B and, of course, the external node ext.

Suppose the whole network is partitioned into n_k sub-networks denoted as $\{S_k \in \mathcal{S} | k = 1, \cdots, n_k\}$. The boundary constraints are the consistent constraints for the air traffic flow on the boundary air-routes, i.e., at any time interval $t \in H_p$, the incoming shift should be equal to the outgoing shift on the boundary air links, $f^{in}_{b_{\upsilon,\upsilon'},\upsilon'} = f^{out}_{\upsilon,b_{\upsilon,\upsilon'}}$, where $(\upsilon, b_{\upsilon,\upsilon'}) \in \mathcal{L}(S)$ and $(b_{\upsilon,\upsilon'}, \upsilon') \in \mathcal{L}(S')$ denote the boundary air segments in two adjacent sub-networks. Based on the previous terminologies, the proposed ATFSP can be written

---

Algorithm 1 Subgradient Algorithm for Lagrangian Dual Problem (14)

1) Pick $\lambda^0 \in \mathbb{R}$;
2) In round $r \geq 0$, solve each $H(\lambda^r, S)$ $(S \in \mathcal{S})$ in parallel;
3) Update $\lambda^{(r+1)}_{b_{\upsilon,\upsilon'}}$ as follows,
$$\lambda^{(r+1)}_{b_{\upsilon,\upsilon'}} = \lambda^{r}_{b_{\upsilon,\upsilon'}} + \alpha^{r}_{b_{\upsilon,\upsilon'}}(f^{in}_{(b_{\upsilon,\upsilon'},\upsilon')} - f^{out}_{(\upsilon,b_{\upsilon,\upsilon'})})$$
where $\alpha^r_{b_{\upsilon,\upsilon'}} \geq 0$ is the step size at round r;
4) Iterate on r until $\lambda^r_{b_{\upsilon,\upsilon'}}$ converges.

---

in the formulation as follows,

$$\begin{aligned}&\min \sum_{S\in\mathcal{S}} J(S)\\ &\text{subject to } \Psi(S)\\ &\qquad \forall((\upsilon, b_{\upsilon,\upsilon'}),(b_{\upsilon,\upsilon'},\upsilon') \in E') f^{in}_{(b_{\upsilon,\upsilon'},\upsilon')} = f^{out}_{(\upsilon,b_{\upsilon,\upsilon'})},\end{aligned}$$

where $\Psi(S)$ is the set of constraints associated with sub-network S, and the sub-network objective function $J(S)$ is defined as follows,

$$\begin{aligned}J(S) = \min \sum_{t\in H_p,\phi\in\Phi,P\in\mathcal{C}} \sum_{a\in A(S)} &\Big\{\big[f^{a,P}_{in}(\phi,t) - r^{a,P}(\phi,t)\big]^2\\ &+ \big[f^{a,P}_{out}(\phi,t) - s^{a,P}(\phi,t)\big]^2\\ &+ M \sum_{P[2]\neq a} f^{a,P}_{in}(\phi,t)\Big\}.\end{aligned}$$

By using Lagrangian relaxation, we can remove the boundary constraints and obtain the following Lagrangian dual problem:

$$\begin{aligned}&\max_{\{\lambda_{b_{\upsilon,\upsilon'}}\in\mathbb{R}|b_{\upsilon,\upsilon'}\in B\}} \min \sum_{S\in\mathcal{S}} J(S) + \lambda_{b_{\upsilon,\upsilon'}}(f^{out}_{(\upsilon,b_{\upsilon,\upsilon'})} - f^{in}_{(b_{\upsilon,\upsilon'},\upsilon')})\\ &\text{subject to } \Psi(S), \quad \forall S \in \mathcal{S}\end{aligned} \tag{13}$$

Let λ be the vector consisting of all $\{\lambda_{b_{\upsilon,\upsilon'}} | b_{\upsilon,\upsilon'} \in B\}$ and define $H(\lambda, S)$ as follows:

$$\begin{aligned}\min\ J(S) &- \sum_{b_{\upsilon,\upsilon'}\in B:\upsilon'\in S} \lambda_{b_{\upsilon,\upsilon'}} f^{in}_{(b_{\upsilon,\upsilon'},\upsilon')}\\ &+ \sum_{b_{\tilde{\upsilon},\tilde{\upsilon}'}\in B:\tilde{\upsilon}\in S} \lambda_{b_{\tilde{\upsilon},\tilde{\upsilon}'}} f^{out}_{(\tilde{\upsilon},b_{\tilde{\upsilon},\tilde{\upsilon}'})}\\ \text{subject to }\ &\Psi(S)\end{aligned}$$

Then the Lagrangian dual problem (14) can be rewritten in the following separable form,

$$\max_{\lambda\in\mathbb{R}} \sum_{S\in\mathcal{S}} H(\lambda, S) \tag{14}$$

Problem (14) can be solved by a standard iterative subgradient method shown in Algorithm 1.

Because the QP-relaxed ATFRSP is convex, Algorithm 1 converges in polynomial time. In addition, the duality gap is zero because the Slater condition holds. Thus, all boundary equalities hold.

IV. Forward-Backward Propagation for Integer Solutions of ATFRSP

After solving the QP-relaxed version of the ATFRSP, we need to compute an integer solution, i.e., all decision variables of the link volumes and link shifts must be integers. It is well known that finding a globally optimal integer solution is NP-hard. Thus, we aim to use a heuristic approach called *forward-backward propagation* algorithm to generate an integer solution, which is shown as follows.

The key step that determines the termination speed and the quality is to select a proper upstream or downstream link to reduce the concerned shift. Due to the heuristic nature, we propose a simple procedure to undertake this selection task.

Procedure 1: Selecting links for shift reduction

1) Inputs: (1) An air traffic network modelled by a directed graph $G = (V, E)$, (2) all link volumes and shifts, and (3) a concerned link i with $P \in \mathcal{C}$, $\phi \in \Phi$, $t \in H_p$ and a gap value $\gamma \in \mathbb{N}$, which needs to be absorbed.
2) Initialization: For each airport $a \in A$ we partition links into different tiers, according to their distances towards either a or a', where the distance of each link is defined by the length of the shortest path from a to the link for the backward propagation purpose, or from the link to a for the forward propagation purpose. Let $\xi_b(i, a)$ and $\xi_f(i, a)$ denote the tier numbers of the link i associated with a in backward propagation and forward propagation, respectively.
3) When backward propagation is required for the link i, i.e., when the value $N_j^P(\phi, t)$ needs to be reduced, e.g., when $C_j(t) < N_j^P(\phi, t)$, we pick $j \in U_i(\phi) \cup U_{i,F}(\phi)$ with $C_j(t) - N_j^P(\phi, t) > 0$ and $P[1] = a$ such that
$$\xi_b(j, a) = \min_{q \in U_i(\phi) \cup U_{i,F}(\phi): N_q^P(\phi,t) < C_q(t) \wedge P[1]=a} \xi_b(q, a).$$
When multiple choices for j are available, pick one with the highest available capacity margin, i.e., $C_j(t) - N_j^P(\phi, t) > 0$. If the gap value γ is too big for link j to absorb, i.e., $\gamma > C_j(t) - N_j^P(\phi, t)$, namely γ is larger than the capacity margin of link j, choose the second link, whose tier number is the minimum among all remaining links, and continue this step until the gap is completely absorbed. This step essentially tries to bring the reduction to the original airport, because there are too many aircraft in the system and ground delay can be applied there.
4) When forward propagation is required for the link i, i.e., when $N_j^P(\phi, t)$ needs to be increased, e.g., when $N_j^P(\phi, t) < 0$, we pick $j \in D_i(\phi) \cup D_{i,F}(\phi)$ with $N_j^P(\phi, t) > 0$ and $P[2] = a$ such that
$$\xi_f(j, a) = \min_{q \in D_i(\phi) \cup D_{i,F}(\phi): N_q^P(\phi,t) > 0 \wedge P[2]=a} \xi_f(q, a).$$
When multiple choices for j are available, pick one with the highest volume value. If the gap γ is too big for the link j to absorb, i.e., $\gamma > N_j^P(\phi, t)$, namely γ is larger than the volume of link j, choose the second one, whose tier number is the minimum among all remaining links, and continue this step until the gap is absorbed. This step essentially tries to bring the reduction to the destination airport, because there are too few aircraft in the system. □

With Procedure 1, we present a forward-backward propagation algorithm, aiming to find an integer solution to ATFRSP.

Algorithm 2: Forward - Backward Propagation

1) **Input:** Let $\{\tilde{f}_{ij}^P(\phi, t) \in \mathbb{R}^+ | i \in E \wedge j \in D_i \cup D_{i,F} \wedge P \in \mathcal{C} \wedge \phi \in \Phi \wedge t \in H_p\}$ be the solution to the QP-relaxed problem. For all $i \in E$, $N_i^P(\phi, 0)$ is known. For all $a \in \mathcal{A}$, $P \in \mathcal{C}$, $\phi \in \Phi$ and $t \in H_p$, $C_{in}^a(t)$, $C_{out}^a(t)$, $r^{a,P}(\phi, t)$ and $s^{a,P}(\phi, t)$ are also known.
2) **Initialization:** For each $t \in H_p$ and $i \in E$, round down all link shifts $\tilde{f}_{ij}^P(\phi, t)$, i.e., $f_{ij}^P(\phi, t) := \lfloor \tilde{f}_{ij}^P(\phi, t) \rfloor$.
3) **Iteration:** For each time interval $t = 1, \cdots, |H_p|$
   a) Update link volumes: for all $i \in E$, $P \in \mathcal{C}$ and $\phi \in \Phi$, $N_i^P(\phi, t) = N_i^P(\phi, t-1) + \left[f_{i,in}^P(\phi, t-1) - f_{i,out}^P(\phi, t-1) \right]$, where $N_i^P(\phi, 0)$ is known.
   b) For each $i \in E$, $P \in \mathcal{C}$, $\phi \in \Phi$ and $j \in U_{i,F}(\phi)$, check holdness of (12e). If
   $$f_{ji}^P(\phi, t) \neq \left\lceil \frac{L_C - L_k}{L_C} f_{jk}^P(\phi, t) \right\rceil,$$
   reduce either $f_{ji}^P(\phi, t)$ or $f_{jk}^P(\phi, t)$ to make the equality hold. Go back to Step (3.a).
   c) For each $i \in E$, $P \in \mathcal{C}$, $\phi \in \Phi$ and $p \in D_{i,F}(\phi)$, check holdness of (12f). If
   $$f_{ip}^P(\phi, t) \neq \left\lceil \frac{L_C - L_q}{L_C} f_{iq}^P(\phi, t) \right\rceil,$$
   reduce either $f_{ip}^P(\phi, t)$ or $f_{iq}^P(\phi, t)$ to make the equality hold. Go back to Step (3.a).
   d) For each $i \in E$, $P \in \mathcal{C}$ and $\phi \in \Phi$, check holdness of (12i). If
   $$\frac{N_i^P(\phi, t)\Delta}{L_i} \underline{S}(\phi, i) > \sum_{j \in D_i(\phi) \cup D_{i,F}(\phi)} f_{ij}^P(\phi, t),$$
   determine the maximum gap value $\gamma \in \mathbb{N}$ with
   $$\frac{(N_i^P(\phi, t) - \gamma)\Delta}{L_i} \underline{S}(\phi, i) \leq \sum_{j \in D_i(\phi) \cup D_{i,F}(\phi)} f_{ij}^P(\phi, t).$$
   Pick $j \in U_i(\phi) \cup U_{i,F}(\phi)$ based on Procedure 1 and reduce $f_{ji}^P(\phi, t-1)$ to lower $N_i^P(\phi, t)$. Continue the selection until the gap γ is absorbed for link i, then set t:=t-1, and go to Step (3.a). If
   $$\sum_{j \in D_i(\phi) \cup D_{i,F}(\phi)} f_{ij}^P(\phi, t) > \frac{N_i^P(\phi, t)\Delta}{L_i} \overline{S}(\phi, i),$$
   reduce $f_{ij}^P(\phi, t)$ to make "$\leq$" hold, and go to (3.a).
   e) Check holdness of (12.h) on link volumes. If
   $$(\exists i \in E) \sum_{P \in \mathcal{C}, \phi \in \Phi} N_i^P(\phi, t) > C_i(t),$$
   pick $j \in U_i(\phi) \cup U_{i,F}(\phi)$ based on Procedure 1 and reduce $f_{ji}^P(\phi, t-1)$ to lower $N_i^P(\phi, t)$. Continue

the selection until the gap is absorbed, i.e., "$\leq$" holds for link i, then set $t := t - 1$ and go to Step (3.a). If

$$(\exists i \in E)(\exists P \in \mathcal{C})(\exists \phi \in \Phi)N_i^P(\phi, t) < 0,$$

pick $j \in D_i(\phi) \cup D_{i,F}(\phi)$ by Procedure 1 and reduce $f_{ij}^P(\phi, t-1)$ to increase $N_i^P(\phi, t)$. Continue the selection until the gap is absorbed, i.e., "$\geq$" holds for the link i., then set $t : t - 1$ and go to Step (3.a).

f) Set $t := t + 1$ and go to Step (3.a).

4) **Output:** The integer values of link volumes and shifts.

Theorem 1: Algorithm 2 terminates to an integer solution to ATFRSP.

Proof: Due to the "hopping assumption" (i.e., Assumption 2), whenever we want to reduce a link volume at t, it suffices to reduce its upstream link shifts at $t - 1$. This ensures that Steps (3.d) and (3.e) are feasible in each iteration, i.e., those gaps can be absorbed. Because all link volumes and shifts are finite, which are monotonically non-increasing during Step (3), Algorithm 2 terminates in a finite number of steps when no more changes on the volumes and shifts are needed. To see that the output of Algorithm 2 is an integer solution to ATFRSP, it is clear that the constraint (12g), (12j)-(12m) hold automatically after rounding down in Step (2), and constraints (12.b)-(12.d) hold in Step (3.a). When the algorithm terminates, clearly, constraints (12e), (12f), (12h) and (12i) all hold. Thus, all constraints in the ATFRSP hold. ■

Algorithm 2 relies on Procedure 1. The Step (2) of Procedure 1 is required only once, and can be done in $O(|A| \cdot |E|^2)$, where $|\cdot|$ denotes the size of a set, because for each $a \in A$ the shortest path problem within a directed graph can be solved in $O(|E|^2)$, where we treat each link of G as a node and nodes of G as links. For each iteration, Step (3.a) requires $|E| \cdot |\mathcal{C}| \cdot |\Phi|$ updates. Step (3.b) requires at most $|E| \cdot |\mathcal{C}| \cdot |\Phi|$, and so is Step (3.c). Step (3.d) requires no more than $2|E| \cdot |\mathcal{C}| \cdot |\Phi|$ comparisons. In case of a backward propagation, it requires at most K_b checks for upstream links, where K_b is the maximum number of the adjacent upstream links for each link. For any directed graph without self-loops, $K_b \leq |V| - 1$, where the equality holds when the graph is complete. Step (3.e) requires at most $2|E|$ comparisons, and in case of a backward propagation or forward propagation, it requires at most K_b or K_f checks for upstream or downstream links, where K_f is the maximum number of the adjacent downstream links for each link. For any directed graph without self-loops, $K_f \leq |V| - 1$, where the equality holds when the graph is complete. The total number of iterations is no more than $\psi := |H_p| + \sum_{t \in H_p, i \in E, P \in \mathcal{C}, \phi \in \Phi} N_i^P(\phi, t)$, as in each extra iteration, at least one link volume will reduce one. So the total worst-case time complexity is $O(|A| \cdot |E|^2 + \psi(4|E||\mathcal{C}||\Phi| + 4|V| + 2|E|) = O(|A| \cdot |E|^2 + \psi|E||\mathcal{C}||\Phi|)$, which is pseudo-polynomial time. Nevertheless, considering that the capacity of any real air traffic network is upper bounded, ψ is bounded. Thus, the algorithm can be considered as polynomial time. Shortly, in the experimental part we will see that the actual complexity for real applications is much lower than the worst-case complexity.

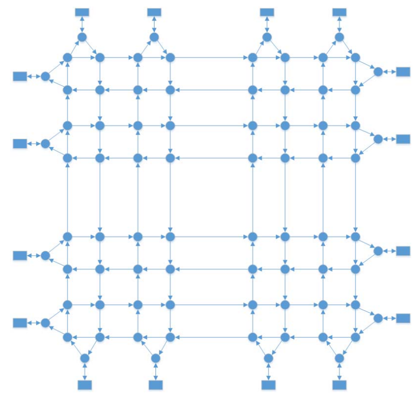


Fig. 4. A 8-8 air traffic system.

TABLE I
NUMBER OF DECISION VARIABLES

System Scale	Decision Variables ($\|H_p\| = 4$)	Decision Variables ($\|H_p\| = 12$)	Decision Variables ($\|H_p\| = 24$)
2to2	1616	4848	9696
4to4	21728	65184	130368
8to8	307904	923712	1847424
12to12	1485216	4455648	8911296

V. EXPERIMENTAL RESULTS

A. Setup of the Air Traffic Network

The layout of the air traffic network in our case study is shown in Fig 4. The airports are connected to the network as the boundary nodes of the air traffic grid. The detailed structure of each airport strictly follows the one depicted in Fig 3.

The numbers of decision variables for different scale systems are shown in Table I. For an 8-by-8 air traffic system with the prediction horizon $|H_p| = 4$, it has 16 airports, 80 waypoints and 176 air-routes, resulting in 307904 decision variables, which is difficult to solve in a centralized way.

B. Experimental Results of Centralized ATFRSP

The optimization problem is solved by CPLEX based on MATLAB on a PC with an Intel Core(TM) i7-4770 @3.40GHz CPU and RAM 8GB. The sampling time is chosen as 5 minutes and the prediction horizon is chosen as $|H_p| = 12, 24, 48, \cdots, 288$, respectively. The prediction horizon $|H_p| = 72$ refers to 6 hours, which is sufficiently long for air traffic in the ASEAN region. We choose several different air traffic grid scales, $n_h = n_v = 2, 4$ or 8, where n_h and n_v denotes respectively the numbers of individual cells horizontally and vertically, where each cell consists of 4-5 waypoints and 0-2 airports depending on whether it is an internal cell or a boundary cell. After applying the

TABLE II

EXPERIMENTAL RESULTS FOR CENTRALIZED APPROACH

$n_h = n_v$	$\|H_p\|$	IQP Solver Processing Time	Relaxed QP Processing Time
2	12	0.02s *	0.02s *
2	24	0.02s *	0.02s *
2	48	0.03s *	0.03s *
2	72	0.13s	0.05s *
2	144	0.16s	0.09s
2	288	0.55s	0.17s
4	12	0.03s *	0.02s *
4	24	0.06s *	0.02s *
4	48	0.23s	0.05s *
4	72	1.44s	0.34s
4	144	1.84s	0.64s
4	288	3.06s	1.36s
8	12	2351.5s	0.17s
8	18	**	0.26s
8	24	-	-

1) * The computational time may not be accurate for these small scale problem because of the minimal clock cycle of the operating system.
2) - The centralized problem cannot be complied, owing to the huge memory consumption.
3) ** The computational time is more than 2 hours.

TABLE III

EXPERIMENTAL RESULTS FOR DISTRIBUTED APPROACH

$n_h = n_v$	$\|H_p\|$	Relaxed QP Processing Time
8	12	0.20s
8	18	0.45s
8	24	3.64s
8	36	13.20s
8	48	16.43s
8	60	133.26s

centralized IQP solver CPLEX, the experimental results are shown in Table II.

From Table II we can see that the centralized approach is only applicable to a system with a small scale and a short prediction horizon. For example, the processing time for solving an 8-by-8 system with a prediction horizon of one hour is 2351.5s, which is does not meet the real-time requirement. Solving the relaxed QP problem requires much less processing time. However, when handling a large scale air traffic system with a long prediction horizon, the centralized approach requires more memory than what a normal computer can provide.

C. Experimental Results of Distributed ATFRSP

We revisit the same case study by applying our proposed distributed air traffic flow routing and scheduling strategy together with the heuristic propagation algorithm (Algorithm 2) with the same PC configuration mentioned before. The results are shown in Table III. All air traffic networks in this test are divided into four 4-by-4 small-scale sub-networks.

Table III clearly indicates that the proposed distributed approach can solve larger-scale problems with longer prediction horizons. For example, the processing time for solving an 8-by-8 system with $|H_p| = 60$ is around 2 minutes, which is shorter than the sampling time of 5 minutes, making a real-time solution feasible. Our experimental work indicates

TABLE IV

THE QUALITY COMPARISON BETWEEN THE DISTRIBUTED APPROACH AND CENTRALIZED IQP APPROACH

$n_h = n_v$	$\|H_p\|$	Total scheduled departure and arrival flights	Centralized IQP Approach Flight Deviation	Distributed Approach plus FB Propagation Flight Deviation	%
4	12	936	286	296	3.5%
4	18	1368	432	440	1.9%
4	24	1800	553	600	7.8%
4	36	2664	882	902	1.1%
4	48	3528	1183	1214	2.6%
4	60	4392	1468	1532	4.4%

that the processing time for larger-scale networks with longer prediction horizons depend on the processing time for solving the local optimization problem associated with each sub-network. Thus, how to handle each sub-network efficiently becomes critically important. Some metaheuristics may be adopted here to overcome this computational challenge, which will be discussed in our future work.

To evaluate the quality of solutions attainable from our distributed approach in comparison with the solutions derivable from solving the centralized IQP, we consider a network with $n_h = n_v = 4$, and $12 \leq |H_p| \leq 60$, and the number of aircraft in the network is about 1.5 time of the network capacity, i.e., the network experiences a heavy traffic "jam", leading to delays of about 1/3 flights. The results shown in Table IV to Table VII are based on the average performance indices and processing time over 20 runs. The experimental results for the number of total departure and arrival deviations are shown in Table IV, which indicate that the difference between the outcomes of the centralized approach and our distributed approach are close to each other, and the largest difference is less than 8%. This suggests that our distributed approach has achieved a tremendous gain in reducing computational complexity with an acceptable degree of quality degradation.

We also provide some experimental results for the forward-backward propagation algorithm to illustrate its usefulness. The numbers of tiers in some networks with different scales are shown in Table V. These numbers indicate the maximal depths of forward propagations and backward propagations. We test the forward-backward propagation algorithm with different levels of "congestions" in an air traffic network with $n_h = n_v = 4$ and $H_p = 5$, where the concept of "congestion" is measured by the percentage of vacancies in the air links. If the vacancy is over 50% of a link capacity, it means the traffic load is light and no congestion in this link. However, if the vacancy is less than 20% of the link capacity, we consider it as being congested. The different air traffic scenarios are obtained by changing the departure rates of aircraft in the airports. Three different scenarios are considered in this paper. In scenario 1, the departure rates are low and no congestion exists in this air traffic network. In scenario 2, the volumes of about 50% links will reach 80% of their link capacities. In scenario 3, the volumes of almost all links in the network will reach about 80% of their link capacities. The test results are shown in Table VI.

TABLE V
NUMBER OF TIERS IN SYSTEMS WITH DIFFERENT SCALES

$n_h = n_v$	4	8	12	16
Number of Tiers	5	7	9	11

TABLE VI
TEST RESULTS OF THE FORWARD-BACKWARD ALGORITHM UNDER DIFFERENT TRAFFIC SCENARIOS

Scenario	Number of backward propagations	Maximal depth of backward	Number of backward propagations reached airports
1	0	0	0
2	24	2	0
3	40	2	10
Scenario	Number of forward propagations	Maximal depth of forward	Number of forward propagations reached airports
1	0	0	0
2	4	1	0
3	43	2	5

TABLE VII
EXPERIMENTAL RESULTS FOR DISTRIBUTED APPROACH

$n_h = n_v$	$\|H_p\|$	Relaxed QP Processing Time	Propagation Processing Time	Total
8	12	0.32s	0.10s	0.42s
8	18	0.65s	0.13s	0.78s
8	24	3.50s	0.13s	3.63s
8	36	11.10s	0.92s	12.02s
8	48	18.52s	1.58s	20.10s
8	60	129.77s	2.69s	132.46s

From Table VI we can see that if no traffic congestions exist in the network (as shown in Scenario 1), there is no need to do any forward or backward operations in the traffic grid when undertaking the heuristic algorithm to find an integer solution of the ATFRSP, because the dynamic equation can always be satisfied and no capacity constraints are violated. In Scenario 2, as some traffic congestion exists, less than 10% of the dynamic equations need to undertake forward or backward propagation to obtain values within relevant capacity constraints. As the number of backward and forward propagations reaching airports equals 0, these propagations will not significantly affect the quality of the final integer solution. In Scenario 3, around 29% of links need to do forward or backward propagations and about 18% of these propagations will reach the airports, which will cause some extra time delays in the final results.

The processing time for the forward backward propagation algorithm for these three scenarios are similar. Each of them takes from 0.10s to 2.69s to be solved, as shown in Table VII. As a conclusion, the time for solving the forward-backward propagation is significantly less than the time for solving the QP-relaxed problem, which certainly takes much less time than solving the centralized IQP problem. Thus, the total time consumption to obtain an integer solution via our distributed approach, which combines the Lagrangian relaxation and the heuristic propagation algorithm, is viable for solving a large-scale air traffic flow routing and scheduling problem.

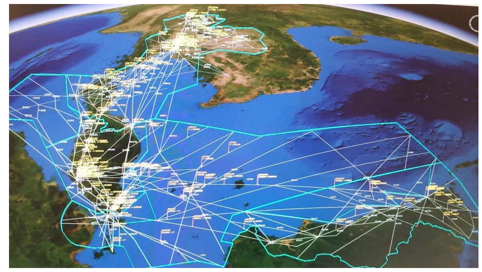


Fig. 5. A simplified air traffic system model for ASEAN region.

TABLE VIII
EXPERIMENTAL RESULTS FOR THE SIMPLIFIED ASEAN AIR TRAFFIC NETWORK

Items	NO. of Decision Variables)	Computation Times
Kota Kinabalu FIR	92736	24.15s
Kuala Lumpur FIR	92736	29.36s
Bangkok FIR	910656	121.63s
Singapore FIR	836640	117.31s
Propagation (Algorithm 2)		2.21s

D. A Case Study Based on ASEAN Air Traffic Network

To further illustrate the effectiveness of our distributed approach, we apply it to a simplified air traffic network shown in Fig 5, which is part of the ASEAN network. The concerned network is constructed based on the en-routed map provided by ICAO [25], [26] for the ASEAN region, which consists of 43 airports, 147 waypoints and 484 air links. There are 166 OD pairs considered in this case study with a time horizon of 4 hours, and the sampling interval Δ is 5 minutes.

To apply our distributed approach, we partition the concerned network into four regions simply based on the FIR settings from the en-route map, i.e., the Kota Kinabalu FIR, the Kuala Lumpur FIR, the Bangkok FIR and the Singapore FIR. The outcome is shown in Table VIII, where the second column lists the number of decision variables in each region, which clearly indicate the scale of the problem. The third column lists the computation time associated with each region and the forward-backward propagation algorithm (i.e., Algorithm 2). Notice that computation in different regions can be carried out in parallel, as each region is usually equipped with a local computational device. The time for the (centralized) update of the multipliers is negligible, owing to the advancement of information and communication technologies. Thus, in real applications the actual computation time for solving that QP-relaxed ATFRSP problem is roughly equal to the maximum regional computation time, which is 121.63s in our case study. Compared with the computation time of running Algorithm 2, which is 2.21s, it is clear that the computational bottleneck is to solve that QP-relaxed ATFRSP problem, even though it is convex. To overcome this computational obstacle, we have been exploring possibilities of embedding meta-heuristic approaches in the Lagrangian multiplier method. Nevertheless, the results shown in Table VIII clearly indicate that our proposed distributed approach is viable for dealing with ATFRSP in a realistic air traffic network.

VI. Conclusion

In this paper we have proposed a realistic air traffic flow management model and formulated an air traffic flow routing and scheduling problem (ATFRSP) as an integer quadratic programming problem, which aims to minimize the deviation from the originally planned departure and arrival shifts. The key idea underlying the problem formulation is to use airport ground delays (via adjusting the departure shifts) and en-route flight routing to minimize the network-wise schedule deviation. To overcome the computational complexity, we have proposed a distributed approach, which first relaxes the original IQP problem into a convex QP problem, solvable by a distributed strategy based on Lagrangian relaxation, and then feeds the outcome of that QP-relaxed problem into a heuristic propagation algorithm to derive a final feasible integer solution. Our experimental results have indicated that this distributed strategy can solve fairly large flow routing and scheduling problems. We will explore meta-heuristics in local computation to improve the computational efficiency of our approach even further.

References

[1] M. Ball *et al.*, "Total delay impact study: A comprehensive assessment of the costs and impacts of flight delay in the United States," Nat. Center Excellence Aviation Oper. Res., Tech. Rep. 01219967, 2010.

[2] A. Agustín, A. Alonso-Ayuso, L. F. Escudero, and C. Pizarro, "Mathematical optimization models for air traffic flow management: A review," *Stud. Inform. Univ.*, vol. 8, no. 2, pp. 141–184, 2010.

[3] D. Bertsimas and S. Stock, "The multi-airport flow management problem with en route capacities," *Oper. Res.*, vol. 46, no. 3, pp. 406–422, 1998.

[4] E. P. Gilbo, "Airport capacity: Representation, estimation, optimization," *IEEE Trans. Control Syst. Technol.*, vol. 1, no. 3, pp. 144–154, Sep. 1993.

[5] E. P. Gilbo, "Optimizing airport capacity utilization in air traffic flow management subject to constraints at arrival and departure fixes," *IEEE Trans. Control Syst. Technol.*, vol. 5, no. 5, pp. 490–503, Sep. 1997.

[6] C. G. Panayiotou and C. G. Cassandras, "A sample path approach for solving the ground-holding policy problem in air traffic control," *IEEE Trans. Control Syst. Technol.*, vol. 9, no. 3, pp. 510–523, May 2001.

[7] M. Terrab and A. R. Odoni, "Strategic flow management for air traffic control," *Oper. Res.*, vol. 41, no. 1, pp. 138–152, 1993.

[8] D. Bertsimas, G. Lulli, and A. Odoni, "The air traffic flow management problem: An integer optimization approach," in *Integer Programming and Combinatorial Optimization*. Berlin, Germany: Springer, 2008, pp. 34–46.

[9] D. Bertsimas, G. Lulli, and A. Odoni, "An integer optimization approach to large-scale air traffic flow management," *Oper. Res.*, vol. 59, no. 1, pp. 211–227, 2011.

[10] D. Bertsimas and S. S. Patterson, "The traffic flow management rerouting problem in air traffic control: A dynamic network flow approach," *Transp. Sci.*, vol. 34, no. 3, pp. 239–255, 2000.

[11] B. Sridhar and P. K. Menon, "Comparison of linear dynamic models for air traffic flow management," in *Proc. IFAC World Congr.*, 2005, pp. 13–18.

[12] D. Bertsimas and S. S. Patterson, "The air traffic flow management problem with enroute capacities," *Oper. Res.*, vol. 46, no. 3, pp. 406–422, 1998.

[13] K. D. Bilimoria, B. Sridhar, S. R. Grabbe, G. B. Chatterji, and K. S. Sheth, "FACET: future ATM concepts evaluation tool," *Air Traffic Control Quart.*, vol. 9, no. 1, pp. 1–20, 2001.

[14] A. M. Bayen, R. L. Raffard, and C. J. Tomlin, "Eulerian network model of air traffic flow in congested areas," in *Proc. IEEE Amer. Control Conf.*, vol. 6. Jun./Jul. 2004, pp. 5520–5526.

[15] A. M. Bayen, R. L. Raffard, and C. J. Tomlin, "Adjoint-based control of a new eulerian network model of air traffic flow," *IEEE Trans. Control Syst. Technol.*, vol. 14, no. 5, pp. 804–818, Sep. 2006.

[16] B. Sridhar, T. Soni, K. Sheth, and G. Chatterji, "Aggregate flow model for air-traffic management," *J. Guid., Control, Dyn.*, vol. 29, no. 4, pp. 992–997, 2006.

[17] D. Sun and A. M. Bayen, "Multicommodity Eulerian–Lagrangian large-capacity cell transmission model for en route traffic," *J. Guid., Control, Dyn.*, vol. 31, no. 3, pp. 616–628, 2008.

[18] C. F. Daganzo, "The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory," *Transp. Res. B, Methodol.*, vol. 28, no. 4, pp. 269–287, 1994.

[19] C. F. Daganzo, "The cell transmission model, part II: Network traffic," *Transp. Res. B, Methodol.*, vol. 29, no. 2, pp. 79–93, 1995.

[20] K. S. Lindsay, E. A. Boyd, and R. Burlingame, "Traffic flow management modeling with the time assignment model," *Air Traffic Control Quart., Int. J. Eng. Oper.*, vol. 1, no. 3, pp. 255–276, 1993.

[21] N. Megiddo, "Linear programming (1986)," *Annu. Rev. Comput. Sci.*, vol. 2, no. 1, pp. 119–145, 1987.

[22] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999.

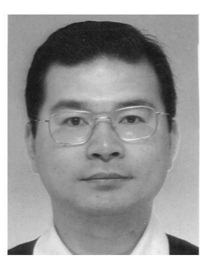
[23] A. Nedić, D. P. Bertsekas, and V. S. Borkar, "Distributed asynchronous incremental subgradient methods," *Stud. Comput. Math.*, vol. 8, pp. 381–407, 2001.

[24] "Roadmap for performance-based navigation: Evolution for area navigation (RNAV) and required navigation performance (RNP) capabilities," Federal Aviation Admin., Washington, DC, USA, Tech. Rep., Jul. 2006, version 2.0.

[25] *Air Routes (CAAS)*, accessed on Oct. 9, 2015. [Online]. Available: http://www.caas.gov.sg/caasWeb2010/export/sites/caas/en/Regulations/Aeronautical_Information/AIP/enroute/enr3.html

[26] *En-Route Chart (CAAS)*, accessed on Oct. 9, 2015. [Online]. Available: http://www.caas.gov.sg/caas/en/Regulations/Aeronautical_Information/AIP/enroute/enr6.html

Yicheng Zhang (S'15) received the master's degree from University of Science and Technology of China in 2014. He is currently working toward the Ph.D. degree with Nanyang Technological University, Singapore. His research interests include optimization method, modelbased fault diagnosis, smart grid, and intelligent transportation systems.



Rong Su (M'11–SM'14) received the B.E. degree in automatic control from University of Science and Technology of China in 1997, and the M.A.Sc. and Ph.D. degrees in electrical engineering from University of Toronto in 2000 and 2004, respectively. Since then he was affiliated with University of Waterloo and Technical University of Eindhoven before he joined Nanyang Technological University in 2010. He has authored or co-authored over 110 publications in journals, book chapters, and conference proceedings, and has two patents in his research fields. His research interests include discrete event systems, supervisory control, model-based fault diagnosis, multi-agent systems, optimization, and scheduling with applications in green buildings, flexible manufacturing, power management, and intelligent transportation systems. He is an Associate Editor of *Journal of Discrete Event Dynamic Systems: Theory and Applications, Journal of Control and Decision*, and *Transactions of the Institute of Measurement and Control*, and is the Chair of the IEEE Control Systems Society Technical Committee on Smart Cities.

Qing Li received the Ph.D. degree in transportation engineering from Nanyang Technological University in 2013. From 2012 to 2015, she was a Research Associate/Fellow with the Computer Aided Engineering Laboratory, Nanyang Technological University, where she is currently a Research Fellow with the Air Traffic Management Research Institute. Her research interests include traffic simulation, traffic flow management, routing and scheduling algorithm, and 3-D visualization.

Christos G. Cassandras (F'96) received the B.S. degree from Yale University, New Haven, CT, USA, in 1977; the M.S.E.E. degree from Stanford University, Stanford, CA, USA, in 1978; and the M.S. and Ph.D. degrees from Harvard University, Cambridge, MA, USA, in 1979 and 1982, respectively. He was with ITP Boston, Inc., Cambridge, from 1982 to 1984, where he was involved in the design of automated manufacturing systems. From 1984 to 1996, he was a Faculty Member with the Department of Electrical and Computer Engineering, University of Massachusetts Amherst, Amherst, MA, USA. He is currently a Distinguished Professor of Engineering with Boston University, Brookline, MA, USA; the Head of the Division of Systems Engineering; and a Professor of Electrical and Computer Engineering. He has authored over 350 refereed papers and five books in his research fields. His research interests include discrete event and hybrid systems, cooperative control, stochastic optimization, and computer simulation, with applications to computer and sensor networks, manufacturing systems, and transportation systems. He is a member of Phi Beta Kappa and Tau Beta Pi. He is also a fellow of the International Federation of Automatic Control. He was a recipient of several awards, including the 2011 IEEE Control Systems Technology Award, the 2006 Distinguished Member Award of the IEEE Control Systems Society, the 1999 Harold Chestnut Prize (IFAC Best Control Engineering Textbook), a 2011 prize and a 2014 prize for the IBM/IEEE Smarter Planet Challenge competition, the 2014 Engineering Distinguished Scholar Award at Boston University, several honorary professorships, a 1991 Lilly Fellowship, and a 2012 Kern Fellowship. He was an Editor-in-Chief of IEEE TRANSACTIONS ON AUTOMATIC CONTROL from 1998 to 2009. He serves on several editorial boards and has been a Guest Editor for various journals. He was the President of the IEEE Control Systems Society in 2012.

Lihua Xie (F'07) received the B.E. and M.E. degrees from Nanjing University of Science and Technology, in 1983 and 1986, respectively, and the Ph.D. degree from University of Newcastle, Australia, in 1992, all in electrical engineering. Since 1992, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, where he is currently a Professor and served as the Head of Division of Control and Instrumentation, from 2011 to 2014. He has held teaching appointments with the Department of Automatic Control, Nanjing University of Science and Technology, from 1986 to 1989, and a Changjiang Visiting Professorship with South China University of Technology from 2006 to 2011. He has published over 300 journal papers and co-authored two patents and seven books in his research fields. His research interests include robust control and estimation, networked control systems, multi-agent networks, and unmanned systems.

Dr. Xie He has served as an Editor of the IET Book Series in Control and an Associate Editor of a number of journals, including IEEE TRANSACTIONS ON AUTOMATIC CONTROL, *Automatica*, IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS- II: EXPRESS BRIEFS. He is a fellow of IFAC.