JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 33, No. 3, May–June 2010

Disaggregation Method for an Aggregate Traffic Flow Management Model

Dengfeng Sun*
Purdue University, West Lafayette, Indiana 47906-2045
and
Banavar Sridhar† and Shon Grabbe‡
NASA Ames Research Center, Moffett Field, California 94035-1000

DOI: 10.2514/1.47469

A linear time-varying aggregate traffic flow model can be used to develop traffic flow management strategies using optimization algorithms. However, there are few methods available in the literature to translate these aggregate solutions into practical control actions involving individual aircraft. In this paper, a computationally efficient disaggregation algorithm is proposed by employing a series of linear program and mixed integer linear program methods, which converts an aggregate (flow-based) solution to a flight-specific control action. Numerical results generated by the optimization method and the disaggregation algorithm are presented and illustrated by applying them to generate traffic flow management schedules for a typical day in the U.S. National Airspace System.

Nomenclature

$A_i(k)$	=	maximum number of arrivals allowed in center i during time interval k
AQ_i	=	queue of arrival flights in center i
$a_i(k)$	=	number of arrivals in center i during time interval k, $\beta_{ii}(k)x_i(k)$
$C_i(k)$	=	maximum number of aircraft allowed in center i at time k
centerQ_i	=	queue of flights in center i
$D_i(k)$	=	maximum number of departures allowed in center i during time interval k
DQ_i	=	queue of departure flights in center i
$d_i(k)$	=	number of departures for center i during time interval k
dest(f)	=	destination center for flight f
$F_{ij}(k)$	=	maximum flows allowed between centers i and j during time interval k
(i, j)	=	directional link from center i to center j
N	=	number of centers
orig(f)	=	origin center for flight f
path(i, j, k)	=	kth shortest path from center i to center j
R_i	=	total number of scheduled arrivals in center i
$S_i(k)$	=	number of departures scheduled for center i during time interval k
T	=	time horizon of simulation and optimization
T_i	=	minimum time that a flight has to spend in center i if the flight goes through it
$[T_{ij}^0, T_{ij}^1]$	=	time interval during which weather impacts traffic flow between center i and center j
$[v]$	=	rounded value of v
x_0	=	initial values of the number of aircraft in centers
$x_i(k)$	=	number of aircraft in center i at the beginning of time interval k
$\beta_{ij}(k)$	=	fraction of aircraft in center i that enter center j during time interval k
ΔT	=	length of time intervals

I. Introduction

TODAY, air traffic flow prediction is done by propagating the trajectories of the proposed flights forward in time and using them to count the number of aircraft in a region of the airspace. Examples of systems that use this physics-based modeling approach for demand forecasting include the Center TRACON Automation System [1], the Future ATM Concepts Evaluation Tool (FACET) [2], and the Collaborative Routing Coordination Tool [3]. The accuracy of these predictions is impacted by departure and weather uncertainties [4,5]. These trajectory-based models predict the behavior of the U.S. National Airspace System (NAS) adequately for short durations of up to 20 min. With the short prediction accuracy, it is difficult to make sound strategic decisions on air traffic management.

Aggregate models simplify the analysis and design of many complex systems. The development of aggregate flow models for traffic flow management has been the subject of considerable interest since the first model [6] appeared in the literature. These models are based on the aggregation of traffic flows and simulate the behavior of volumes of flights. Compared with trajectory-based models, aggregate flow models are computationally efficient, and their complexity does not increase with the number of aircraft. The aggregate flow models previously published [7] represent and predict the traffic behavior to a high degree of accuracy and can be tailored to the time scales and regions of interest. A comparison of the characteristics of the different flow models can be found in previous studies [8,9]. The stability and response characteristics of the aggregate flow models are presented in Chatterji and Sridhar [10]. The aggregation in flow models generally results in the loss of information about the route structure of individual aircraft. Additional research is needed to translate the aggregate control policies into actual traffic flow management flight planning decisions involving aircraft departure times and routes. Currently, there is limited experience in the application of aggregate and reduced-order flow models to generating flight-level traffic flow management decisions. In a previous work by the same authors of this paper, a preliminary disaggregation algorithm was developed [11]. In this paper, a detailed disaggregation algorithm and more discussions are given in a full version. A larger number of experiments were conducted, taking into account more practical scenarios. The modeling-optimization-disaggregation framework has been implemented through FACET interfaces.

Received 2 October 2009; revision received 3 February 2010; accepted for publication 6 February 2010.

*Assistant Professor, School of Aeronautics and Astronautics. Member AIAA.
†Senior Scientist. Fellow AIAA.
‡Research Scientist. Member AIAA.

In this paper, an aggregate traffic flow model developed in a previous work [7] is implemented. The system parameters of the model are identified using historical air traffic data. Using filed flight plans as an input to the model, the traffic situation is predicted and compared with real data. A two-step approach is used to solve traffic flow management problems. The first step solves an optimization problem with the system dynamics of the aggregate model as part of the constraints. In the second step, a detailed computationally efficient disaggregation algorithm is designed to convert flow-based optimal solutions from the first step to flight-specific control actions, which is the key contribution of this paper. Application of the methodology to managing traffic on a typical day in the U.S. National Airspace System shows that the disaggregation algorithm generates control actions for individual flights while keeping the air traffic behavior very close to the optimal solution.

This paper is organized as follows. The next section briefly introduces the aggregate flow model. The optimization algorithm and the disaggregation algorithm are described in Secs. III and IV, respectively. Section V presents the application results of the optimization framework. Finally, concluding remarks are included in Sec. VI.

II. Aggregate Flow Model

A. Dynamic System Model

An aggregate traffic flow model was developed in a previous work [7]. It is a linear-time-variant dynamic system model (LDSM). The number of aircraft at different times in each center is represented by a state variable. The number of landings in a center and transitions from the center to the neighboring centers in an interval of time ΔT are assumed to be proportional to the number of aircraft in the center at the beginning of the interval. Using the principle of conservation of flow in a center, the number of aircraft in center i at the next instant of time $k+1$ can be related to the number of aircraft in i at k via the difference in the number of aircraft that came into the center and the number of aircraft that left the center as follows:

$$x_i(k+1) = x_i(k) - \sum_{j=1}^{N} \beta_{ij}(k)x_i(k) + \sum_{j=1, j\neq i}^{N} \beta_{ji}(k)x_j(k) + d_i(k) \tag{1}$$

or

$$x_i(k+1) = x_i(k) - \sum_{j=1, j\neq i}^{N} \beta_{ij}(k)x_i(k) - a_i(k) + \sum_{j=1, j\neq i}^{N} \beta_{ji}(k)x_j(k) + d_i(k)$$

The number of arrivals in center i during the kth time interval, denoted by $a_i(k) = \beta_{ii}(k)\cdot x_i(k)$, is a fraction of $x_i(k)$. In special cases, $\beta_{ii}(k) = 0$ implies that a transition must involve crossing an airspace boundary, and $\beta_{ii}(k) = 1$ implies physical transition from one time interval to another time interval. The departure within center i is denoted by $d_i(k)$, which is independent of $x_i(k)$. For simplicity of illustration, "during the kth time interval ΔT" will also be understood as "*at* time k" in this paper.

B. Implementation and Validation of Linear-Time-Variant Dynamic System Model

The inputs to the LDSM include departures $d_i(k)$ within center i at time k and the fractions $\beta_{ij}(k)$. The direct output from the LDSM is the aircraft count in the centers at each time step. It is also straightforward to generate other outputs, such as intercenter traffic flows, number of arrivals, etc., based on the information of $x_i(k)$, $\beta_{ij}(k)$, and $d_i(k)$.

To implement the LDSM, the fractions $\beta_{ij}(k)$ are obtained as the aggregated fractions of aircraft going from center i to j during the same kth time interval in each day, using historical air traffic data. In this study, the departures $d_i(k)$ are computed from filed flight plans (deterministic). This is different from the original model in [7], which includes both deterministic and stochastic components. The LDSM model is implemented in a deterministic manner because a deterministic optimization method (Sec. III) will be used for traffic flow management (TFM). The LDSM is implemented in C++ programming language. All centers of the U.S. airspace, oceanic centers, and part of the Canadian centers are included in the LDSM.

The LDSM is validated against real air traffic data. Figure 1 shows a comparison of real and simulated aircraft counts, with two time interval size values $\Delta T = 1$ and 15 min, in Oakland center (ZOA) from PDT 5:00PM on 23 August 2005 to PDT 4:59PM on 24 August 2005. For this validation, the fractions $\beta_{ij}(k)$ are obtained using the historical traffic data from 6 September 2001, and the departures $d_i(k)$ are generated using the flight data from 24 August 2005. Different dates were used in the validation to measure the quality attributes of the LDSM. It is shown that the LDSM correctly predicts the trends of the evolution of aircraft counts in each center, and the average relative absolute error between the simulation and real data is less than 2%. Similar results are observed for all continental centers in the United States. The maximum absolute error magnitude is 61 (number of aircraft).

It took less than 30 s to process one day of historical data (enhanced traffic management system flight track information) to compute β with the C++ code on a 1.8 GHz CPU, 2 GB RAM IBM ThinkPad T42 laptop with a Linux operating system. Performing a simulation of air traffic for a whole day (24 h) took approximately 5 s. Compared with most trajectory-based models that predict the behavior of the U.S. National Airspace System adequately for shorter durations, the LDSM is efficient at predicting the traffic situation for

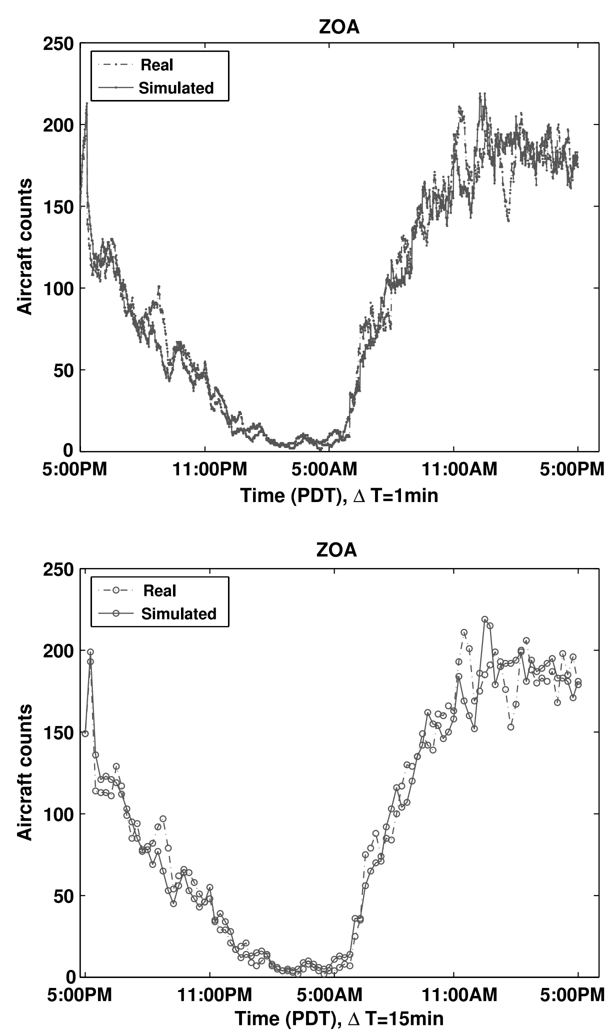

Fig. 1 Oakland center traffic.

a longer time horizon, which is helpful when making strategic decisions on air traffic management.

The aggregate flow model can be used to design traffic flow management strategies to minimize different objectives subject to various constraints. Several optimization methods have been designed for traffic flow management [12–16], most of which are based on networked air traffic flows. In the next section, a new optimization method will be proposed that fully uses the aggregate flow properties of the LDSM and is suitable for future disaggregation to generate flight-specific control.

C. Remarks

The aggregation in the LDSM can be done at different levels, for example, at a center level (in this paper), a sector level, or a user-defined level. For a sector-level LDSM, the state variable x_i represents the number of aircraft at different times in each *sector*, and $\beta_{ij}(k)$ represents the fraction of traffic flow from sector i to sector j at time k. Figure 2 shows a comparison of aircraft counts at a sector level (sector ZFW61). The optimization and disaggregation algorithms designed in this paper can be applied to different LDSMs accordingly.

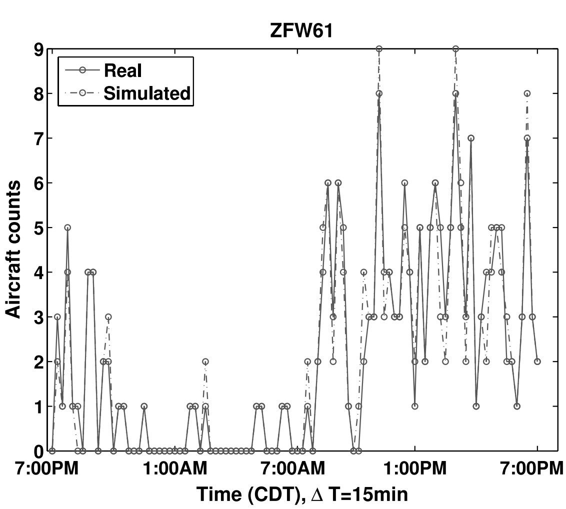

Fig. 2 Aircraft count in sector ZFW61.

III. Optimization Method

In this section, using the aggregated flow model, LDSM, an optimization method will be designed to optimize the parameters β and $d_i(k)$ in Eq. (1) for traffic flow management. The selection of $\beta_{ij}(k)$ and $d_i(k)$ is equivalent to a traffic flow management strategy based on choosing routes and aircraft departure times to minimize a cost function.

A TFM problem is formulated as follows:

$$\min_{\beta,d} \mathcal{J}(x) \tag{2}$$

subject to the initial condition (Sec. III.A.1), LDSM dynamics (Sec. III.A.2), center capacity constraints (Sec. III.A.3), intercenter flow constraints (Sec. III.A.4), departure constraints (Sec. III.A.5), arrival constraints (Sec. III.A.6), and minimum dwell time constraints (Sec. III.A.7). The constraints and the cost function (2) are described in the following subsections.

A. Constraints

1. Initial Condition

The initial condition

$$x(0) = x_0 \tag{3}$$

represents the number of aircraft in each center at the beginning ($k = 0$) of the optimization.

2. LDSM Dynamics

The LDSM is subject to the following dynamics:

$$x_i(k+1) = x_i(k) - \sum_{j=1}^{N} \beta_{ij}(k) x_i(k) + \sum_{j=1, j\neq i}^{N} \beta_{ji}(k) x_j(k) + d_i(k)$$
$$k \in \{0, \dots, T\}, \qquad i \in \{1, \dots, N\} \tag{4}$$

which is presented in Sec. II.

3. Center Capacity Constraint

The time-varying center capacity constraints are formulated as

$$0 \le x_i(k) \le C_i(k), \qquad k \in \{0, \dots, T\}, \qquad i \in \{1, \dots, N\} \tag{5}$$

which restricts the number of aircraft in a center to be below the maximum number of aircraft allowed.

4. Intercenter Flow Constraint

Intercenter flow constraints are formulated by

$$\beta_{ij}(k)x_i(k) + \beta_{ji}(k)x_j(k) \le F_{ij}(k), \qquad k \in \{T_{ij}^0, \dots, T_{ij}^1\} \tag{6}$$

which forces the flow between centers i and j below the maximum amount of flow allowed during the time interval $[T_{ij}^0, \dots, T_{ij}^1]$.

5. Departure Constraint

Departure-related constraints are as follows:

$$0 \le d_i(k) \le D_i(k), \qquad k\{0, \dots, T\}, \qquad i\{1, \dots, N\} \tag{7}$$

$$\sum_{k=0}^{T} d_i(k) = \sum_{k=0}^{T} = S_i(k), \qquad i \in \{1, \dots, N\} \tag{8}$$

where Eq. (7) restricts the number of departures below the departure capacity at each time step; Eq. (8) enforces all flights to depart. In this constraint, departure time k for $S(k)$ is defined as the scheduled departure time.

6. Arrival Constraint

In the arrival constraints,

$$0 \le a_i(k) \le A_i(k), \qquad k \in \{0, \dots, T\}, \qquad i \in \{1, \dots, N\} \tag{9}$$

$$\sum_{k=0}^{T} a_i(k) = R_i, \qquad i \in \{1, \dots, N\} \tag{10}$$

Equation (9) restricts the number of arrivals below the arrival capacity, and Eq. (10) enforces the total number of arrivals by time T (left-hand side of the equation) to be equal to the total number of scheduled arrivals (right-hand side of the equation).

7. Minimum Dwell Time Constraint

The amount of time an aircraft spends in a center depends on the design speed of the aircraft and the flight route in the center. For example, A is used to denote the starting point of a route in center i and B is used to denote the ending point, as shown in Fig. 3.

The minimum flight travel time along the route $A \to B$ can be formulated as:

$$\sum_{j\in N^B} \beta_{Bj}(k) x_{AB}(k) = 0, \qquad k \in \{0, \dots, T_{AB} - 1\} \tag{11}$$

$$\sum_{j\in N^A} \beta_{jA}(k) x_j(k) = 0, \qquad k \in \{T - T_{AB} + 1, \dots, T\} \tag{12}$$

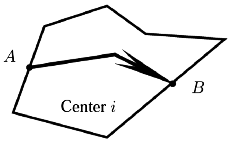

Fig. 3 Example of a route from A to B.

$$\sum_{k=T_{AB}}^{t+T_{AB}} \sum_{j\in N^B} \beta_{Bj}(k)x_{AB}(k) \leq \sum_{k=0}^{t} \sum_{j\in N^A} \beta_{jA}(k)x_j(k)$$

$$t \in \{0, \ldots, T - T_{AB}\} \tag{13}$$

where $x_{AB}(k)$ is the aircraft count on route $A \to B$ at time k, N^B is the set of downstream flows connecting B, and N^A is the set of upstream flows connecting A. T_{AB} is the minimum travel time on route $A \to B$.

In the current paper, a special case for this constraint is studied. The minimum dwell time is assumed to be identical for every flight flying through a center despite the fact that flights use different routes that need different travel times, that there are flights taking off and landing in the same center, and that flights taking off and landing have different dwell times than en route flights, etc. For the special case, this constraint can be formulated as follows:

$$\sum_{j=1}^{N} =\beta_{ij}(k)x_i(k) = 0, \qquad a_i(k) = 0$$

$$k \in \{0, \ldots, T_i - 1\}, \qquad i \in \{1, \ldots, N\} \tag{14}$$

$$\sum_{j=1}^{N} \beta_{ji}(k)x_j(k) = 0, \qquad d_i(k) = 0$$

$$k \in \{T - T_i + 1, \ldots, T\}, \qquad i \in \{1, \ldots, N\} \tag{15}$$

$$\sum_{k=T_i}^{t+T_i} \left(\sum_{j=1}^{N} \beta_{ij}(k)x_i(k) + a_i(k) \right) \leq \sum_{k=0}^{t} \left(\sum_{j=1}^{N} \beta_{ji}(k)x_j(k) + d_i(k) \right)$$

$$t \in \{0, \ldots, T - T_i\}, \qquad i = \{1, \ldots, N\} \tag{16}$$

The formulation of minimum dwell time constraints for the general case introduces one constraint for each route, whereas the special case has one constraint for each center; therefore, it is expected that the general case will take more computational time for the optimization.

Details about this constraint are omitted for brevity and will be presented in Appendix A.

8. Remarks

1) In all constraints, the fraction variables $\beta_{ij}(k)$ always appear together with $x_i(k)$. Therefore, alternative variables $f_{ij}(k)$ are defined where $f_{ij}(k) = \beta_{ij}(k)x_i(k)$; $f_{ij}(k)$ is actually the amount of traffic flow that leaves center i and enters center j at time k. Replacing $\beta_{ij}(k)x_i(k)$ by $f_{ij}(k)$, all constraints (3–16) are linear in $x_i(k)$, $d_i(k)$, and $f_{ij}(k)$.

2) In the implementation of the optimization method, some flights will not land as T time horizon concludes, which causes infeasibility for optimization. In this case, those flights are treated as "background" flights, and their behavior is simulated using the LDSM, Eq. (1). Because of the existence of the background flights, the capacities $C_i(k)$, $A_i(k)$, $D_i(k)$, and $F_{ij}(k)$ are reduced as follows:

$$C_i(k) := C_i^o(k) - x_i^b(k) \qquad A_i(k) := A_i^o(k) - a_i^b(k)$$

$$D_i(k) := D_i^o(k) - d_i^b(k) \qquad F_{ij}(k) := F_{ij}^o(k) - x_i^b(k)\beta_{ij}^b(k)$$

where $\beta_{ij}^b(k)$ is computed from historical data for the LDSM, variables with superscript b represent background flights, and capacities with superscript o are the original capacities without any background flights.

B. Cost Function

The cost function $\mathcal{J}(x)$ in Eq. (2) can be defined in various forms according to specific objectives. For example,

$$\mathcal{J}(x) = \sum_{i=1}^{N} \sum_{k=0}^{T} x_i(k)$$

which is equivalent to minimizing the total flight time (see Appendix B).

When minimization of the departure delays is also part of the objective, the cost function can be formulated as

$$\mathcal{J}(x) = \underbrace{\sum_{i=1}^{N} \sum_{k=0}^{T} x_i(k)}_{J_0(x)} + \underbrace{\sum_{i=1}^{N} \sum_{t=0}^{T} \left(\sum_{k=0}^{t} S_i(k) - \sum_{k=0}^{t} d_i(k) \right)}_{J_1(d)}$$

where $J_0(x)$ is the total flight time, and $J_1(d)$ is the total departure delay.

Delays in different centers can also include designated weights to impose some requirements on the equity with which delays are distributed across centers [17]. The objective function can be formed as follows:

$$\mathcal{J}(x) = \sum_{i=1}^{N} \sum_{k=0}^{T} w_i^{\text{en route}}(k)x_i(k) + \sum_{i=1}^{N} \sum_{t=0}^{T} \left(\sum_{k=0}^{t} w_i^{\text{departure}}(k)(S_i(k) - d_i(k)) \right)$$

where $w_i^{\text{en route}}$ and $w_i^{\text{departure}}$ are the weights for en route delay and ground delay in center i, respectively.

The objective functions given here are subject to the same constraints described in Sec. III.A.

The example objective functions are linear, and the optimization problems are linear.

IV. Disaggregation Algorithm

The LDSM and the optimization method are formulated based on aggregated traffic flows. An individual flight's identity is not preserved. A disaggregation algorithm, which converts an optimal aggregate flow-based solution (d and β) to a flight-specific control action, is presented in this section. (This disaggregation algorithm works for solutions other than optimization as well: given the number of departures and intercenter transition flows, the algorithm converts the flows into flight-specific actions, if possible.) The optimization algorithm outputs for each center the number of departures, the number of aircraft going to each neighbor, and the number of aircraft that should be in it, during each time period. Based on these optimization results, the disaggregation algorithm then determines which flights should depart from a center during each time period, the optimal route (a sequence of centers), and the optimal transition times between centers along the optimal route for each flight. An example of an output from the disaggregation algorithm is as follows: the action for flight UA332 is to depart at 12:15PM from ZOA center, arrive at the border between ZOA–ZLA centers at 12:45PM (where ZLA is the Los Angeles center), land in ZLA at 1:30PM. The accuracy for the flight-specific control is determined by ΔT, that is, control actions are taken every ΔT instance. Similar control actions will be generated for all the flights.

The disaggregation algorithm includes three steps: preparation, disaggregation, and cleanup.

A. Preparation

In the "preparation" step, 1) the possible fractional number solutions from the optimization method will be rounded to integers for physical and practical purposes in air traffic control, 2) a departure flight queue will be built for flights taking off from each center, and 3) a graph will be designed for possible routing choices of each center pair.

1. Rounding

Round $d_i(k)$ and $f_{ij}(k) = \beta_{ij} x_i(k)$ to integers. This procedure is necessary for practical traffic control actions, that is, the controls are applied to an integer number of flights instead of "one-and-a-half" airplanes being directed to take off when $d_i(k) = 1.5$ for example.

2. Departure Queue

Generate a departure queue DQ_i for each center i by picking

$$\sum_{k=0}^{T} d_i(k)$$

flights scheduled to take off from the center:

$$\text{length}\,(DQ_i) = \sum_{k=0}^{T} d_i(k) = \sum_{k=0}^{T} S_i(k)$$

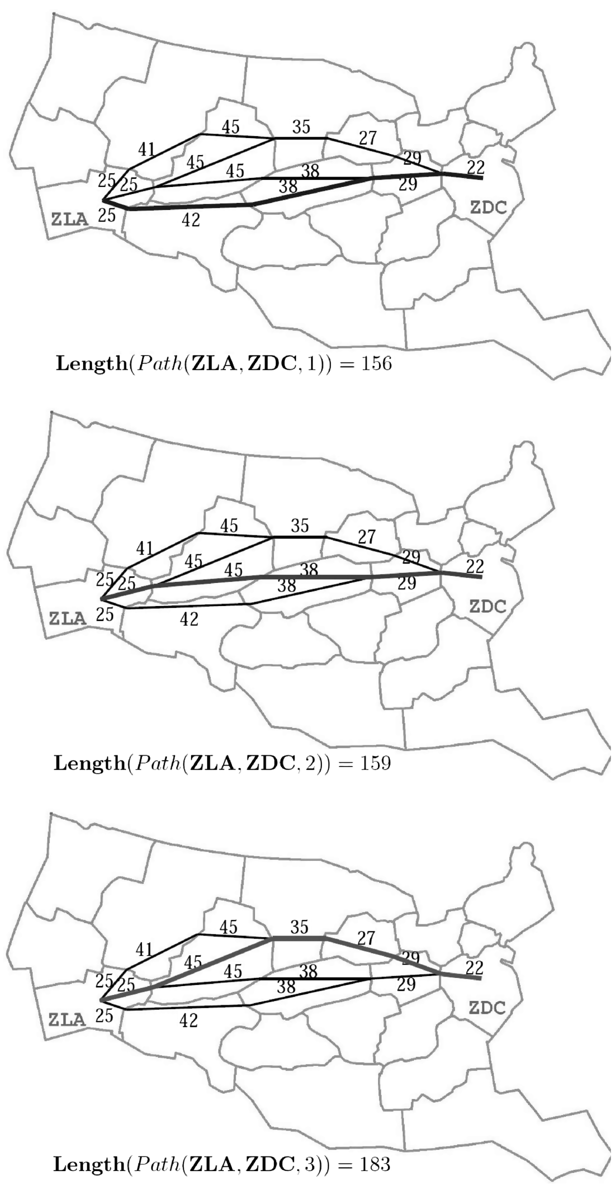

Fig. 4 Three possible paths from ZLA to ZDC. A number represents the length (in minutes) of a link.

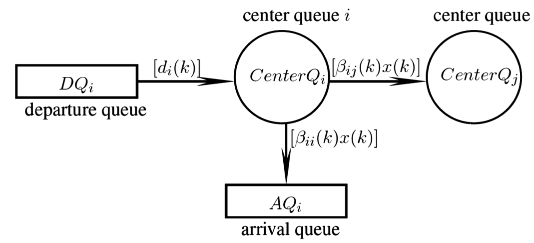

Fig. 5 Disaggregation step.

Flights with the earliest scheduled departure time are stored at the front of the departure queue. The order of takeoff for the flights (leaving the departure queue) will be first in/first out.

3. Graph for Paths

Generate a graph that includes possible route choices for each center pair. In this step, m possible flight paths (routes) for each origin–destination center pair are generated from historical air traffic data. In this paper, $m = 3$. The three paths are ranked from "shortest" to "longest" based on the average flight times; for paths originating from center i with a destination in center j:

$$\text{length (path}(i, j, 1)) \leq \text{length(path}(i, j, 2)) \leq \text{length(path}(i, j, 3))$$

Figure 4 shows an illustration of three possible paths from ZLA to ZDC.

B. Disaggregation

In the disaggregation step, each flight in the departure queue DQ_i will be directed to depart from its origination center at an optimal departure time. Every airborne flight in each center will be controlled to either land in the current center at an optimal time or to leave the current center and enter a successive center at an optimal time. The successive center is chosen optimally with the objective [Eq. (2)]. The procedure of disaggregation follows.

At each time k for each center i:

1) Take $[d_i(k)]$ (rounded value of $d_i(k)$) flights from each departure queue DQ_i and push these flights into the center flight queue centerQ_i (a queue containing all airborne flights in center i).

2) Disaggregate intercenter flows: optimally direct flights from one center to another or to land (pseudocode in Algorithm 1; an illustration shown in Fig. 5):

a) For each center flight queue centerQ_i, pop out the flights for which the destination is this center; push these flights in the arrival flight queue AQ_i, and record the current time k when the flights enter AQ_i.

b) Round intercenter flow $f_{ij}(k)$ to $[f_{ij}(k)]$. For each center $j \in$ neighbors of(i), find possible flights in centerQ_i that may go to their destinations through center j, that is, find a set of flights (denoted by fid) $fid \in$ centerQ_i, such that j is the second center on path$(i, \text{dest}(fid), m)$, $m = 1$, 2, and 3. path$(i, \text{dest}(fid), m)$ is

Algorithm 1 Disaggregation for intercenter flows

for $fid \in$ centerQ_i do
    if dest$(fid) = i$ then
        center$Q_i \cdot$ pop(fid)
        $AQ_i \cdot$ push(fid)
    end if
end for
solve optimal routing problem (17) and (18) for $r(fid, j)$
for $fid \in$ centerQ_i do
    for $j \in$ neighbors of (i) do
        if $r(fid, j) = 1$ then
            center$Q_j \cdot$ push(fid)
            center$Q_i \cdot$ pop(fid)
        end if
    end for
end for

defined in the preparation step (step a). Then find an optimal strategy of routing $n \approx [f_{ij}(k)]$ number of flights from center i to j, such that the overall cost for routing the flights in the current center to their destinations is minimized:

$$\min_{r(fid,j)} \sum_{fid \in \mathrm{center}Q_i} \sum_{\substack{m=1,2,3 \\ j=\text{second center on}\,\mathrm{path}(i,\mathrm{dest}(fid),m)}} r(fid,j) \cdot \mathrm{length}(\mathrm{path}(i,\mathrm{dest}(fid),m)) \tag{17}$$

subject to

$$r(fid,j) \in \{0,1\}; \qquad \sum_{fid \in \mathrm{center}Q_i} r(fid,j) \geq [\beta_{ij}(k)x_i(k)] - \alpha \tag{18}$$

where $r(fid,j) = 1$ (or 0) means flight fid is (or is not) routed from center i to j, and $\alpha = 1$ in the practical implementation. Even though $n \geq [f_{ij}(k)] - \alpha$ could theoretically lead to a very large number for n, in the aforementioned mathematical formulation for the optimal routing, the *minimization* in the objective function [see Eq. (17)] forbids large n. As observed in the results of the optimization, n equals $[f_{ij}(k)]$ for over 90% of the intercenter flows, and in the rest cases, $n = [f_{ij}(k)] - \alpha$. The reason why $n = [f_{ij}(k)]$ is not applied in the constraints is because the strict equality oftentimes causes infeasibility of the optimization problem, that is, it is possible that no such flight-specific disaggregation solution exists to route exactly $[f_{ij}(k)]$ number of flights from center i to j at time k.

c) For each flight fid in centerQ_i, if $r(fid,j) = 1$ by the optimization problem (17) and (18), pop out fid from centerQ_i and push it into centerQ_j.

3) For each arrival queue AQ_i, find all flights that are eligible (current eligibility criteria: it has to be in the arrival queue of center i for at least T_i units of time) to land. If the number of eligible flights is larger than $[a_i(k)]$, pop out $[a_i(k)]$ flights from AQ_i and let them land; otherwise, pop out all eligible flights from AQ_i and let these eligible flights land.

C. Cleanup

As a result of numerical rounding to $d_i(k)$, $f_{ij}(k)$, and $a_i(k)$, the center capacity constraint of Eq. (5) could be violated,[§] and some flights may be airborne (called "residual flights" in this study) at the end of optimization time horizon (4 h in the present study). (By observation of thousands of runs on different scenarios, the number of residual flights is less than 40, which constitutes less than 0.5% of the total flights.) A cleanup process is necessary to make sure that every flight in the departure queue DQ_i for each center i lands in its destination during the optimization time horizon. The procedure of cleanup follows.

For every residual flight fid:

1) Find the origin center orig(fid) and destination center dest(fid) for flight fid.

2) Find a minimum cost path from orig(fid) to dest(fid). The new "cost" of flying through center j includes the minimum dwell time T_j in the center and the maximum occupancy of the center during the disaggregation step (step b), $\max_k \bar{x}_j(k)$, where $\bar{x}_j(k)$ is the number of aircraft in center j at time k as a result of the disaggregation step (step b):

$$\mathrm{cost}\,(j) = \max_k \bar{x}_j(k) + T_j$$

Dijkstra's algorithm [18] is used to find the minimum cost path. Figures 6 and 7 illustrate an example of the new cost and the minimum cost path from ZLA to ZDC.

[§]From the disaggregation results, it is observed that the capacity is rarely violated in practice, including thousands of runs of different scenarios. In the very few cases in which the capacity is exceeded, the violation is very minor, namely, at most two aircraft more than the capacity, which is about 1% over the capacity.

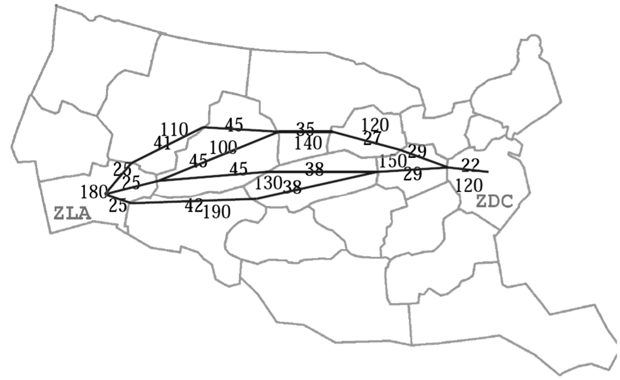

Fig. 6 Illustration of new costs for the links along possible paths from ZLA to ZDC. The small numbers (less than 100) are the original costs T_i (see Fig. 5); the big numbers (larger than 100) are the maximum occupancies of the centers from the disaggregation step (step b).

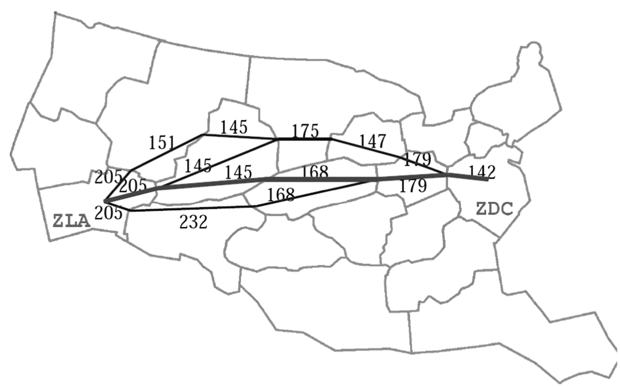

Fig. 7 Highlighted path is the minimum cost path from ZLA to ZDC based on the new link costs in Fig. 6.

3) Let flight fid fly along the aforementioned minimum cost path. (It is possible that an additional flight fid can break the capacity of a center through which the flight flies. However, in the experiments this has not happened; because there are less than 0.5% residual flights in total, the capacity breach is not considered a big threat.)

Remarks. The new cost for a minimum path can be defined in several ways. For example, it may be a time-varying cost and include the marginal capacities of the centers to accept additional flights, that is, $C_j(k) - \bar{x}_j(k)$ for center j at time k; in this case, the new cost for a flight going through center j is defined as

$$\mathrm{cost}\,(j,k) = C_j(k) - \bar{x}_j(k) + T_j$$

To account for the cases in which $C_j(k) - \bar{x}_j(k) \leq 0$, a very high cost is assigned for cost(j,k). For this problem, a time-varying minimum cost algorithm [19] can be applied to find the optimal path.

V. Application to Traffic Flow Management

The optimization method in Sec. III and the disaggregation algorithm in Sec. IV are implemented on the same platform as in Sec. II.B. In the implementation of the optimization model, ILOG CPLEX Concert optimization library is used to solve the optimization problem coded in C++ .[¶] Several application examples of the model and the algorithm are described in the rest of this section. For the airspace including 20 continental centers of the United States, 4 h TFM problems were solved in the examples. The states of the LDSM are updated every 15 min. (The number of variables are proportional to number of time stamps, which is inversely proportional to time interval ΔT for a fixed time horizon. Using a smaller time interval of

[¶]Data available online at http://www.ilog.com/corporate/training/acrobat/CPLEX.pdf [retrieved 12 June 2009].

less than 15 min will result in larger computational times than those presented in this paper.) All 8422 flights that departed from the continental centers during the time horizon were included in the study. The modeling, optimization, and disaggregation framework is implemented with FACET [2] through an application programming interface (API).

A. Minimization of Delays with Center Capacity Constraints

In the example presented in this section, the objective is to minimize the total flight time and the departure delays under the constraints of center capacities. The cost function is defined as follows:

$$\mathcal{J}(x) = \sum_{i=1}^{N}\sum_{k=0}^{T} x_i(k) + \sum_{i=1}^{N}\sum_{t=0}^{T}\left(\sum_{k=0}^{t} S_i(k) - \sum_{k=0}^{t} d_i(k)\right) \quad (19)$$

Figures 8 and 9 show four center count profiles in Cleveland (ZOB) and New York (ZNY) centers:

1) The profile labeled *Real* is the recorded real center aircraft counts.

2) The profile labeled *Simulated* is a prediction of center aircraft counts using the LDSM, that is, the fractions β in Eq. (1) are computed from historical data, and the departures $d_i(k) = S_i(k)$, where $S_i(k)$ are scheduled departures.

3) The *Optimized* profile is an output from the optimization with the objective function of Eq. (19). Notice that the optimized center counts are below the center capacity, which is the baseline capacity computed as the maximum observed center counts on a good weather day plus 10%. (The intercenter flow capacity, departure capacity, and arrival capacity are computed in the same manner.)

4) The *Disaggregated* profile is the output from the disaggregation algorithm, which generates flight-specific action.

5) A solid line *Capacity* represents the capacity of each center.

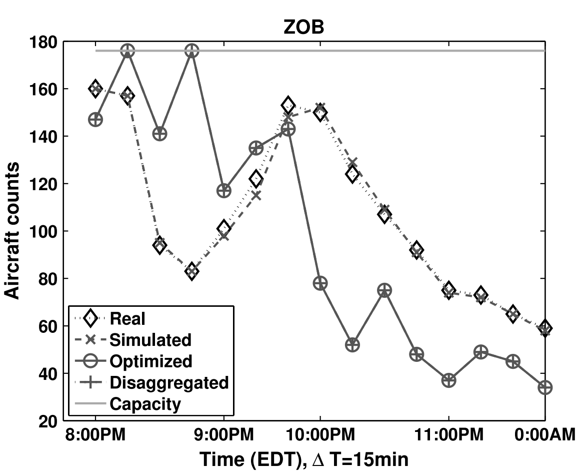

Fig. 8 Comparison of center count profiles in Cleveland center.

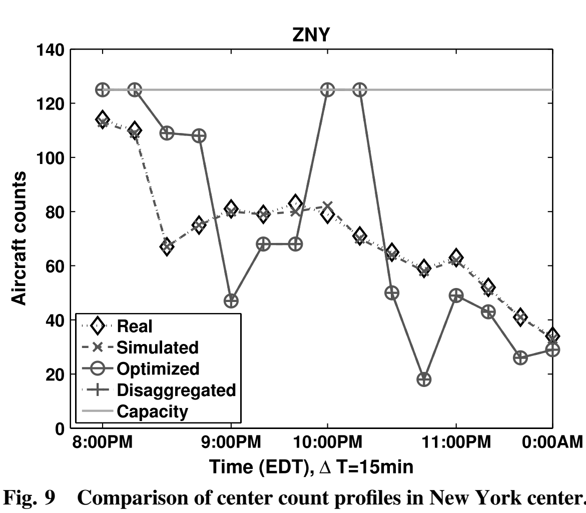

Fig. 9 Comparison of center count profiles in New York center.

Table 1 Comparison of the costs with different aircraft count profiles in all centers

Center counts (x)	Real	Simulated	Optimized	Disaggregated
$\mathcal{J}(x)$	1.28×10^7	1.3×10^7	9.1×10^6	1.0×10^7

Similar results are observed in the rest centers of the entire NAS. Table 1 summarizes the costs defined by Eq. (19) for the real, simulated, optimized, and disaggregated data, including all flights in 20 continental centers in the U.S. Clearly, the optimized solution has the minimal cost among the four, whereas the disaggregated profile obtained with flight-specific information has a larger cost. The simulated data are very close to the real data because the LDSM predicts flows and aircraft counts in centers very close to the real situation. Depending on the values of T_i in the minimum dwell time constraint in Eq. (16), the optimal solution has a lower cost compared with the real data and simulated data in general.

There is a big difference between the optimized and the real center count profiles, in particular, the former is larger than the latter in the beginning, reaching the capacity. This is as expected: because the objective is to minimize the total flight time, the optimization model will try to "push" the flights to fly as fast as they can, using the minimum dwell time in a center. Therefore, most centers will accommodate as many as possible (close to capacity) flights. However, the centers can not keep the amount of flights to capacity all times. For example, if center *A* can accommodate one more flight before reaching its capacity and there is one flight coming from center *B*, a neighbor of center *A*, routing this flight from *B* to *A* may not be the optimal option (for instance, going to *A* may result in a much longer path to the destination). Because of the difference in center counts between the optimized and real situations at the beginning of the planning time horizon, the optimized and the real count profiles can be very different later.

The average computational time for optimizations of several test cases in each day of May 2005 is shown in Fig. 10; the average computational time for disaggregation is in Fig. 11. The computations are performed on a Dell OptiPlex 760 desktop, with Intel® Core$^{\mathrm{TM}}$2 Duo processor E8400, 4GB RAM, and Linux operating system. Notice that the optimization and disaggregation can start at any time given initial conditions (current airborne flights' situation) and flight plans (incoming flights). The run time is much

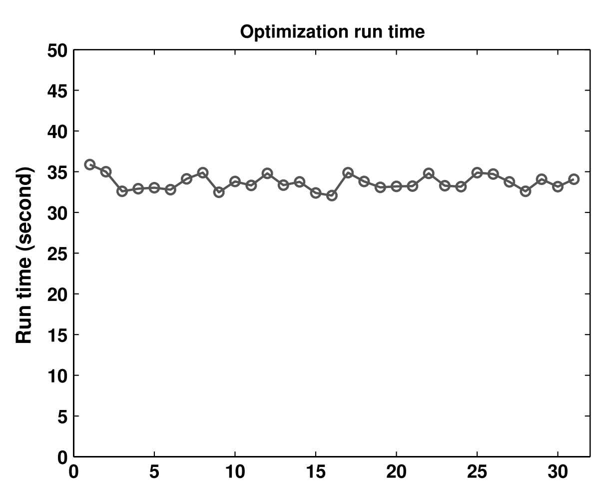

Fig. 10 Average computational time for optimization. Horizontal axis denotes 31 days in May 2005.

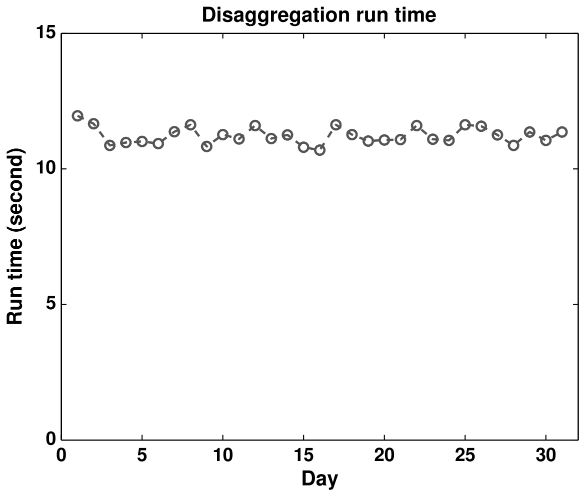

Fig. 11 Average computational time for disaggregation. Horizontal axis denotes 31 days in May 2005.

smaller than the planning horizon, which makes it possible to develop an operational tool for TFM practitioners.

B. Minimization of Delays Using a Sector-Level LDSM

In this section, an example of minimizing delays using a sector-level LDSM is presented. The same objective function as in Eq. (19) and the same constraints are used in this example. (Instead of center capacities, sector capacities are encoded in the optimization problem. T_i is now the minimum dwell time in sector i.) Fig. 12 shows four sector count profiles of sector ZFW48, using the same notations as in the previous section. The experiment is run for the entire continental U.S. airspace, including 385 high-altitude sectors. The average computational time for optimization is 7.2 min and for disaggregation about 2 min.

C. Optimal Flow Routing

Figure 13 displays a weather scenario in which Chicago (ZAU), Indianapolis (ZID), Kansas City (ZKC), and Memphis (ZME) centers are impacted by the weather, and their capacities are reduced correspondingly. Using the LDSM, it is observed that there is a major flow going through ZKC–ZAU–ZOB under normal weather conditions. Because of the weather impact, the center capacity of ZAU is assumed to have a 75% reduction [20]. Under this assumption, the optimization model and the disaggregation algorithm generate a routing strategy such that around 70% of the flights, which use the ZKC–ZAU–ZOB route under normal weather conditions, used a route ZKC–ZMP–ZOB instead. Figure 14 demonstrates this scenario: the dashed line represents the route under normal weather conditions, whereas the solid line represents the route under weather impact. In the current optimization-disaggregation work, equity is not considered, and the objective is to minimize the *overall* delay for the entire system. It is expected that extra cost, both in time and extra fuel, will occur for some flights; some flights might choose to divert, cancel, or be delayed rather than detouring so far. These factors will need to be studied in future research.

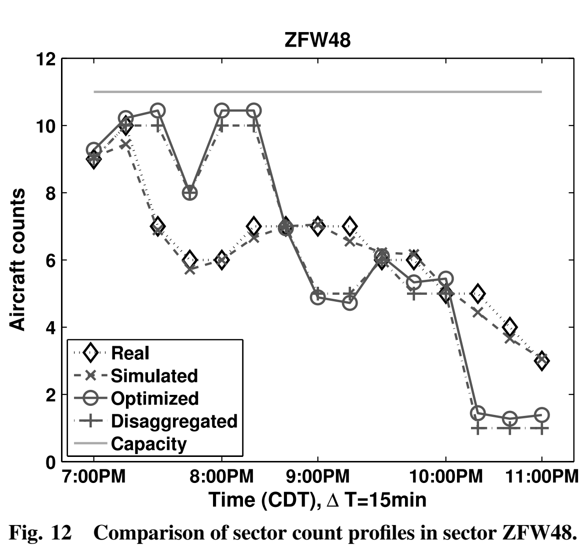

Fig. 12 Comparison of sector count profiles in sector ZFW48.

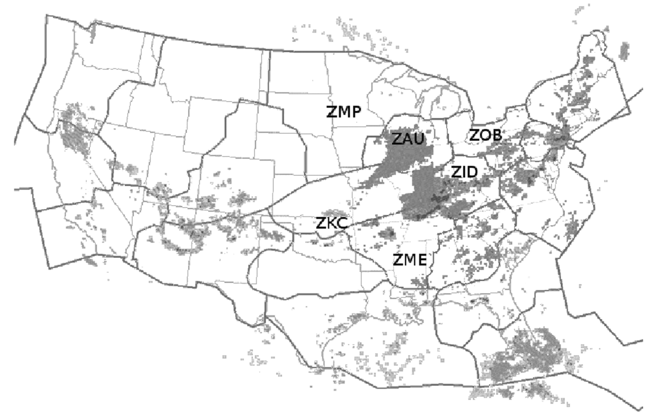

Fig. 13 Example of National Convective Weather Diagnostic weather report.

D. Implementation in FACET

The modeling, optimization, and disaggregation framework is implemented with FACET [2] using Java API. For each flight, one of the outputs from the disaggregation method is the optimal route defined by a sequence of centers, the optimal entry time to each center, and the optimal arrival time in destination. As an initial implementation, the physical entry and exit locations to each center are randomly picked with points on the boundary of the center, which can be refined in future study. Based on the entry/exit location and the optimal arrival/exit time, an optimal local route (inside a center) and an optimal speed are computed in FACET.

Figure 15 shows a snapshot of an implementation of this work in FACET. The gray lines represent the optimal route inside each center. Notice that the flight situation is updated every 15 min in optimization and disaggregation procedures; therefore, multiple flights can be directed to enter the same route at the same time (the beginning of a 15 min interval); as a consequence, each aircraft symbol in Fig. 15 may represent multiple flights at the same location. The actual number of aircraft in each center is equal to the *Disaggregated* numbers shown in Figs. 8 and 9, as an example for the ZOB and ZNY centers.

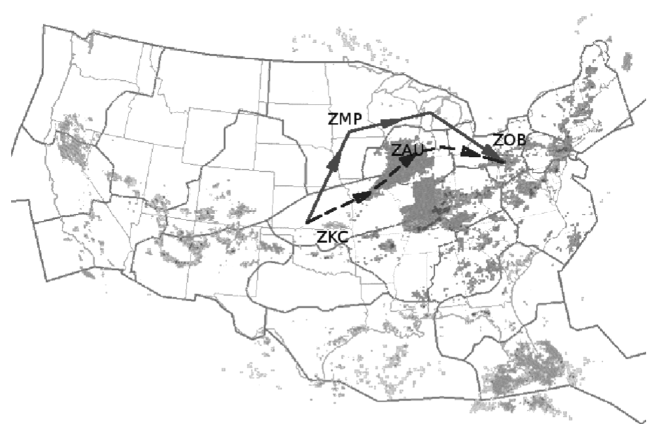

Fig. 14 Example of optimal flow routing (dashed line is the original route; solid line is a reroute).

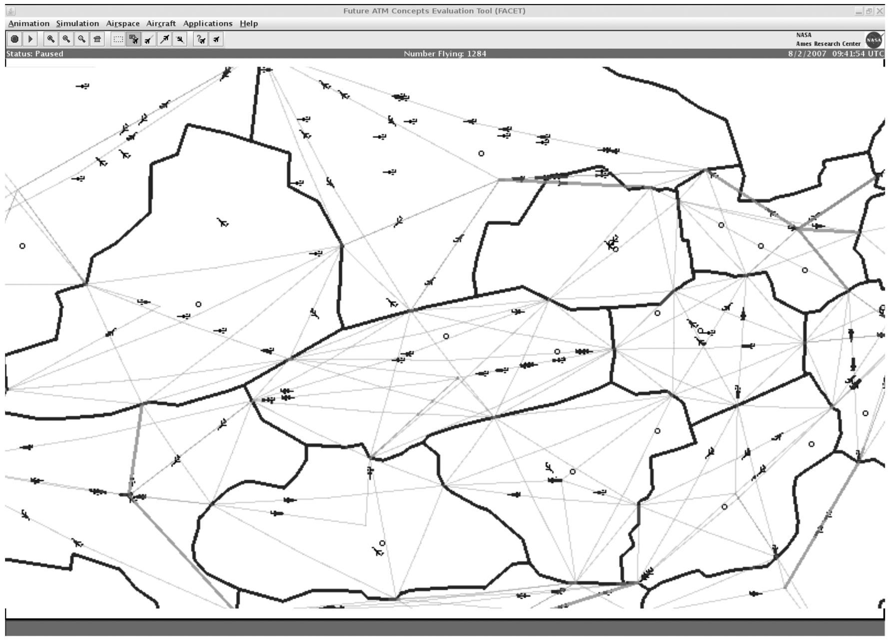

Fig. 15 Snapshot of FACET.

VI. Conclusions

In this paper, a linear time-varying aggregate air traffic flow model is implemented and validated against real air traffic data. It is shown that the aggregate model can be adapted to fit various spatial and temporal resolutions, and generate a similar traffic situation compared with historical air traffic data. Using the aggregate flow model, a linear program optimization model is proposed for traffic flow management at a national level. Solving the optimization problem results in optimal traffic flows. Based on the aggregate model and the optimal traffic flows, a disaggregation algorithm is developed to convert the flow-based optimal solution to flight-specific control actions. The output from the modeling-optimization-disaggregation framework includes the optimal route, the optimal entry time to each center, and the optimal arrival time at the destination for each flight. The computational efficiencies for the optimization model and the disaggregation algorithm are evaluated. The optimization model and the disaggregation algorithm are implemented and integrated into FACET, a NASA software for a flexible air traffic management simulation environment, and applied to nationwide traffic flow management problems.

Appendix A: Minimum Dwell Time Constraint

Because every aircraft flies at a reasonable speed (not too fast) in a center, the optimization method enforces that each flight has to spend a minimum amount of time in each center it passes.

In the constraint (14),

$$\sum_{j=1}^{N} \beta_{ij}(k)x_i(k) = 0, \qquad k \in \{0, \ldots, T_i - 1\}$$

means that the outflows from center i (the left-hand side of the equation) in the first T_i units of time is zero, because any flight in center i has to stay in the center for at least T_i units of time and cannot exit from the center before time T_i. Combining the nonnegativity constraints on β and x, this constraint is identical to the following constraint:

$$\beta_{ij}(k)x_i(k) = 0, \qquad k \in \{0, \ldots, T_i - 1\}, \qquad j = \{1, \ldots, N\}$$

The second part of Eq. (14), $a_i(k) = 0$, $k \in \{0, \ldots, T_i - 1\}$, means that there are no arrivals in center i during the first T_i units of time because any flight that arrives in the destination center i has to spend a minimum of T_i units of time flying in center i before landing.

Constraint (15) means that there are no incoming flows or departures in center i during the last T_i units of time, because otherwise these flights will still be in air at the end of time horizon (T) in the study, due to the minimum flight time requirement.

In constraint (16),

$$\sum_{j=1}^{N} \beta_{ij}(k)x_i(k) + a_i(k)$$

represents the number of "loss" flights, which equal the number of outgoing flights ($\sum_{j=1}^{N} \beta_{ij}(k)x_i(k)$) and landing flights $a_i(k)$ in center i at time k. Consequently, the term

$$\sum_{k=T_i}^{t+T_i} \left(\sum_{j=1}^{N} \beta_{ij}(k)x_i(k) + a_i(k) \right)$$

represents the cumulative loss flights from time T_i to $t + T_i$ in center i. This is also the cumulative loss flights from time 0 to $t + T_i$ because during the first T_i units of time there are no loss flights [by constraint (14)]. As a counter part, the term

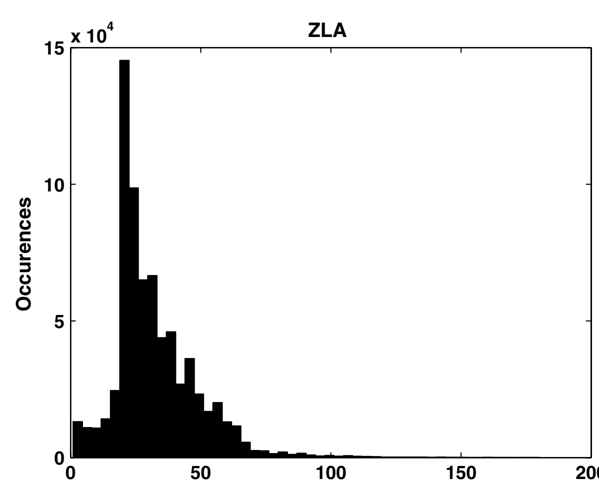

Fig. 16 Dwell times for all flights in ZLA. Data collected in March, May, and September of 2005.

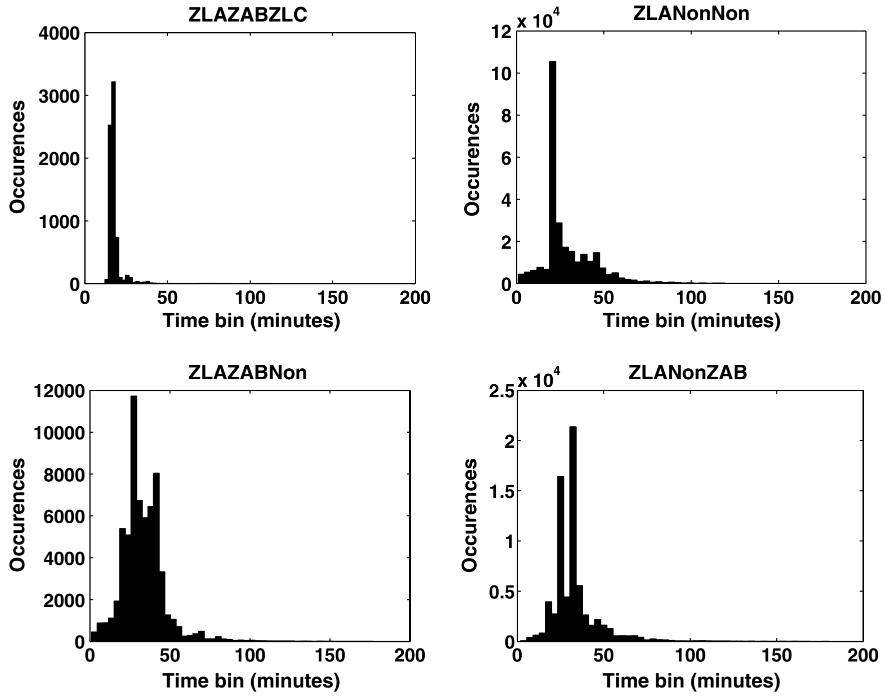

Fig. 17 Examples of dwell times: a) flights from ZAB through ZLA to ZLC, b) flights taking off and landing in ZLA (flights did not exit ZLA), c) flights from ZAB landing in ZLA, and d) flights taking off in ZLA and entering ZAB. Data collected in March, May, and September of 2005.

$$\sum_{k=0}^{t}\left(\sum_{j=1}^{N}\beta_{ji}(k)x_j(k)+d_i(k)\right)$$

represents the cumulative "gain" flights from time 0 to time t. Altogether, the constraint (16) can be interpreted as follows. In center i, the cumulative number of loss flights cannot exceed the cumulative number of gain flights from T_i units of time earlier. The gain flights have to spend at least T_i time in center i before getting lost.

The minimum dwell time is computed using historical air traffic data. Figure 16 shows an example of the dwell times in the ZLA center. In this example, the 25th percentile dwell time (20 min) is chosen as the minimum dwell time for all flights through ZLA. Figure 17 shows examples of dwell times for four different routes inside the ZLA center.

Appendix B: State and Travel Time

One of the characteristics of the LDSM is that the states of the dynamical system are directly related to flight time. State $x_i(k)$ represents the number of aircraft in the center at time k. Assuming the flights do not transfer to a neighbor center or land, $x_i(k)$ is also the number of aircraft in center i during a left-closed and right-open time interval $[k\Delta T, (k+1)\Delta T)$. Therefore, $x_i(k)\Delta T$ is the flight time for all aircraft in center i in the kth time interval. Clearly, the total flight time spent in center i from time T_i^0 to T_i^1 is

$$\Delta T\sum_{k=T_i^0}^{T_i^1}x_i(k)$$

For all the centers, the total flight time (during the entire time horizon) is

$$\Delta T\sum_{i=1}^{N}\sum_{k=0}^{T}x_i(k)$$

Therefore, minimizing the following cost function

$$\mathcal{J}(x)=\sum_{i=1}^{N}\sum_{k=0}^{T}x_i(k)$$

is equivalent to minimizing the total flight time.

References

[1] Erzberger, H., "CTAS: Computer Intelligence for Air Traffic Control in the Terminal Area," NASA Ames Research Center TM 103959, July 1992.
[2] Bilimoria, K., Sridhar, B., Chatterji, G., Sheth, K., and Grabbe, S., "FACET: Future ATM Concepts Evaluation Tool," *Air Traffic Control Quarterly*, Vol. 9, No. 1, 2001, pp. 1–20.
[3] Rhodes, L. S., Rhodes, L. R., and Beaton, E. K., "CRCT Capabilities Detailed Functional Description," MITRE Tech. Rept. 00W0000302, March 2001.
[4] Krozel, J., Rosman, D., and Grabbe, S., "Analysis of En Route Sector Demand Error Sources," AIAA Paper 2002-5016, Aug. 2002.
[5] Mueller, E., and Chatterji, G., "Analysis of Aircraft Arrival and Departure Delay Characteristics," AIAA Paper 2002-5866, Oct. 2002.
[6] Menon, P. K., Sweriduk, G. D., and Bilimoria, K., "A New Approach for Modeling, Analysis and Control of Air Traffic Flow," AIAA Paper 2002-5012, Aug. 2002.
[7] Sridhar, B., Soni, T., Sheth, K., and Chatterji, G., "Aggregate Flow Model for Air-Traffic Management," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 4, 2006, pp. 992–997. doi:10.2514/1.10989
[8] Sridhar, B., and Menon, P. K., "Comparison Of Linear Dynamic Models for Air Traffic Flow Management," *16th IFAC World Congress*, Paper 20050703-08, July 2005.
[9] Sun, D., Yang, S., Strub, I., Bayen, A., Sheth, K., and Sridhar, B., "Eulerian Trilogy," AIAA Paper 2006-6227, Aug. 2006.
[10] Chatterji, G., and Sridhar, B., "Some Properties of the Aggregate Flow Model of Air Traffic," AIAA Paper 2005-7456, Sept. 2005.
[11] Sun, D., Sridhar, B., and Grabbe, S., "Traffic Flow Management Using Aggregate Flow Models and the Development of Disaggregation Methods," AIAA Paper 2009-6007, Aug. 2009.
[12] Sridhar, B., Grabbe, S., and Mukherjee, A., "Modeling and Optimization in Traffic Flow Management," *Proceedings of the IEEE*, Vol. 96, No. 12, Dec. 2008, pp. 2060–2080. doi:10.1109/JPROC.2008.2006141
[13] Rios, J., and Ross, K., "Solving High Fidelity, Large-Scale Traffic Flow Management Problems in Reduced Time," AIAA Paper 2008-8910, Sept. 2008.

[14] Grabbe, S., Sridhar, B., and Mukherjee, A., "Sequential Traffic Flow Optimization with Tactical Flight Control Heuristics," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 3, 2009, pp. 810–820. doi:10.2514/1.40300

[15] Wanke, C., "Continual, Probabilistic Airspace Congestion Management," AIAA Paper 2009-5864, Aug. 2009.

[16] Sun, D., "Large scale modeling and optimization of en route air traffic," Ph.D. Dissertation, Univ. of California at Berkeley, Berkeley, CA, Dec. 2008.

[17] Bloem, M., and Sridhar, B., "Optimally and Equitably Distributing Delays with the Aggregate Flow Model," *IEEE/AIAA 27th Digital Avionics Systems Conference*, Inst. of Electrical and Electronics Engineers, New York, Oct. 2008, pp. 3.D.4-1–3.D.4-14.

[18] Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C., *Introduction to Algorithms*, 2nd ed., Prentice–Hall, Upper Saddle River, NJ, 2002, pp. 788–789.

[19] Cai, X., Kloks, T., and Wong, C., "Time-Varying Shortest Path Problems with Constraints," *Networks*, Vol. 29, No. 3, 1997, pp. 141–150. doi:10.1002/(SICI)1097-0037(199705)29:3<141::AID-NET2>3.0.-CO;2-H

[20] Klein, A., Cook, L., and Wood, B., "Airspace Availability Estimation for Traffic Flow Management Using the Scanning Method," *IEEE/AIAA 27th Digital Avionics Systems Conference*, Inst. of Electrical and Electronics Engineers, New York, Oct. 2008, pp. 3.B.5-1–3.B.5-10.