Reducing Air Traffic Delay in a Space-Time Network

Dr. Marcia P. Helme
Center for Advanced Aviation System Development
The MITRE Corporation
McLean, VA 22102-3481

Abstract **The demand for air traffic has grown significantly during the last two decades, increasing the congestion and delay in the National Airspace System (NAS). When demand exceed capacity, present practice is to hold flights on the ground, prior to departure. The only capacity restriction driving these ground delay decisions is the arrival capacity of each flight's destination airport. A computerized system-wide decision tool would enhance this practice if it could also evaluate the impact of airway capacities upon traffic from multiple origins to multiple destinations. Such a tool would further enhance the process if it could also generate delay-reducing alternative schedules. Network flow optimization algorithms produce and evaluate alternative system-wide strategies very quickly. This paper describes a formulation of the NAS traffic management problem as a multicommodity minimum cost flow problem over a network in space-time.**

I. INTRODUCTION

The United States Federal Aviation Administration (FAA) must manage the daily traffic flow of approximately 60,000 flights among more than 400 towered airports. Two of the FAA's major air traffic flow management (TFM) objectives are to decrease congestion and to decrease delay. Congestion increases controller and pilot workloads and threatens human safety. Congested air traffic saturates portions of the National Airspace System (NAS) frequently, even in good weather. Inclement weather reduces airport and airspace capacities, exacerbating the traffic management problem.

Delay, on the other hand, is costly. The FAA has estimated the annual cost of delay at $1.4 billion [4]. System-wide delay in the NAS consists of both airborne and ground delay. For the purposes of this paper, we consider an aircraft to incur airborne delay if its travel time is greater than originally scheduled, and we consider an aircraft to incur ground delay if its departure time is later than originally scheduled. Airborne delay is usually more expensive than ground delay, since fuel usage is higher when airborne.

Note that delay and congestion can be competing conditions. The preventive measure to decrease airborne congestion is to hold flights on the ground prior to departure. Conversely, reducing the proposed ground delay at 7 a.m., for example, may increase airborne congestion later.

The contents of this paper reflect the views of the authors and not necessarily the official views or policy of The MITRE Corporation or the Federal Aviation Administration.

We consider managing traffic under normal capacities as a *planning problem* and the rescheduling of traffic under temporarily reduced capacities as an *operational problem.* This paper and presentation address the *operational problem.*

At the national level, the FAA uses several operational TFM strategies to diminish congestion and delay. One strategy, called the Ground Delay Program, is to delay at their departure airports those aircraft destined for airports suffering or anticipating arrival demand significantly in excess of expected arrival capacity. The intent of this practice is to reduce the arrival rates into those airports during the period when the airports are experiencing capacity insufficient for the demand. The only capacity restrictions driving the Ground Delay Program's decisions are the arrival capacities of each flight's destination airport. The program does not address the possible impact, on the total system, of simultaneous ground delays, airborne spacing restrictions, decreased airway capacities, and separation restrictions. The program ignores these factors not from unawareness of their importance but because the FAA has no tool to measure those factors nor any way to generate schedules in real time that reduce system-wide delay. This paper and presentation show how a network flow optimization formulation can generate and evaluate flight schedules that simultaneously minimize system-wide delay and adhere to capacity constraints at NAS facilities: departure, arrival and transit facilities., i.e., airports and airspace.

We consider that most models contain both *simplifying assumptions* and *nonsimplifying assumptions*. One employs simplifying assumptions, knowing that they do not accurately represent the system one is modeling, because they simplify either the model formulation or the model solution method, or both. One employs nonsimplifying assumptions because they represent system characteristics whose elimination would cause serious misrepresentation of the system.

An earlier MITRE TFM analysis showed the current ground delay practice yields a system-wide optimum, i.e., a delay-minimizing strategy, under a set of three *simplifying assumptions*: (1) the NAS is entirely deterministic, i.e., there is no uncertainty in the NAS capacities and travel times, (2) all airway capacities are infinite or at least nonrestrictive, and (3) an aircraft's flight plan arrival airport always represents its final destination. None of these three assumptions truly represents the NAS traffic. It follows that a delay-minimizing optimization model that adheres to all these assumptions can at best emulate present practice rather than

0-7803-0720-8/92 $3.00

improve it. Therefore, for an optimization model to generate strategies with less system-wide delay than present practice, the model's assumptions must negate at least one of the three simplifying assumptions, cited above, under which current practice is optimal. That is, *the model must reflect the effects of uncertainty, the effects of restrictive en route capacities, or the effects of hubbing*.

We also note three *nonsimplifying*, possibly complicating assumptions of present practice that we wish to retain in the model because they represent important characteristics of the NAS. (1) Contrary to automobile traffic or telecommunication traffic, for example, aircraft do not enter the system according to a Poisson process. In fact, the number, routes, and schedules of commercial flights are predetermined and controllable. (2) The capacities—of airports and airspaces—vary with time. The airspace over Kansas, for example, may have ample capacity under normal conditions but zero capacity during a thunderstorm. (3) If an operational tool is to help evaluate daily traffic management options, it must generate alternative strategies quickly. Optimization techniques are useful methods for generating and evaluating alternative strategies when there are competing objectives over many interacting variables and constraints. However, optimization methods that provide very good solutions sometimes require long execution times. An algorithm that requires a long execution time to generate a theoretical optimum is much less useful than a heuristic that quickly generates a solution that is not optimal but is a significant improvement over current strategies.

A literature search found no large-scale optimization models representing enough of the NAS's characteristics to be useful as an operational tool to manage traffic on the scale of the NAS. For example, several models consider en route congestion but only one destination airport. Authors who tried to extend these to examples of multiple destinations report that the solution never converged. Conversely, several models in the literature represent multiple destinations but assume infinite airspace capacity. MITRE project documentation describes the literature search and the models it evaluates.

II. MODEL FORMULATION

We now describe a network optimization formulation that negates one of the three simplifying assumptions of present practice, namely the assumption of nonrestrictive airway capacities. That is, this formulation assumes multi-destined traffic competing for a common set of finite, restrictive facilities. We describe in a later section how the model can accommodate negation of the other two simplifying assumptions. The model retains the three nonsimplifying assumptions stated above.

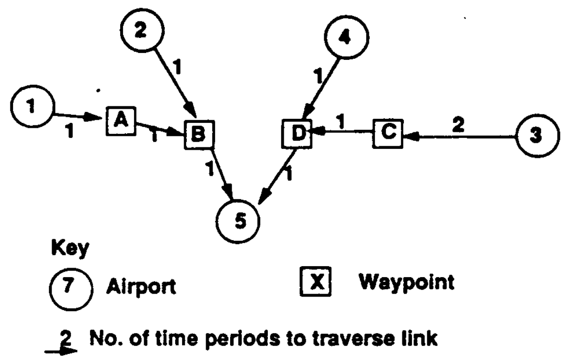
There are two networks of interest in the model: the spatial network and the time-space network.

A. The Spatial Network

A node in the NAS spatial network can represent an airport or a waypoint. Aircraft depart from and arrive at airports; aircraft fly through waypoints. Thus an airport is an origin for some flights and a destination for others. A waypoint is neither an origin nor a destination but is a trans-shipment node through which traffic flows; it may be a departure fix, an arrival fix, a point of intersecting airways, or any point in the airspace that one wishes to designate. A link represents an airway between two nodes. Inflows into the network at node i represent departures from airport i entering the airspace; outflows from node j represent arrivals at airport j exiting the airspace. The capacity of a link represents a never-to-be-exceeded limit that prevents excessive congestion. A link's cost parameter represents delay. The amount of flow on a link represents the number of aircraft on that link.

Classical network model formulations allow capacitated links but assume node capacity is unlimited. In the NAS, however, nodes (airports, fixes) are more likely to be capacitated than links. The formulation can resolve this dilemma and allow any node to be capacitated and/or cost-incurring with the following maneuver: one splits the node in two and calls it a link.

Consider an example of four origin airports, one destination airport, four waypoints, and four flights. Each flight originates at a different departure airport, but all flights are destined for the same airport, node 5, as shown in Figure 1.



Flight	Origin	Earliest Depart Time	Waypoints (location, time)			Destination
1	1	t=0	A, t=1	B, t=2		5, t=3
2	2	t=0		B, t=1		5, t=2
3	3	t=0		C, t=2	D, t=3	5, t=4
4	4	t=1			D, t=2	5, t=3

Figure 1. Spatial network with single destination.

Numbers designate airport nodes; letters designate waypoints. We assume that the scheduled departure times present no congestion problem in good weather, but that "today's" link and node capacities are reduced due to bad weather or any other phenomenon.

B. The Space-Time Network

To generate the space-time network equivalent of the spatial network in Figure 1, we first divide time into discrete periods of equal length. Time interval t, with t an integer, therefore refers to the set

$$\{ \tau \mid t-1 < \tau \leq t \}.$$

A time period may be of any size, probably on the order of 15 to 60 minutes. A shorter time period provides more accuracy, but it increases the size of the network.

In the space-time network, we use the terms *vertex* and *arc* to replace the terms node and link in the spatial network. Each vertex in the space-time network possesses a spatial index k and a time index t. There are three broad classes of arcs in the space-time network.

(1) An arc in space only, i.e., from one location to another with no change in time period. In Figure 2, the horizontal arcs a_1 and a_2 are arcs in space only. If each time period is 15 minutes or longer, landing is an example of an event in which an aircraft changes location within one time period.

(2) An arc in time only, i.e., from one time period to another with no change in location. In Figure 2, the vertical arcs a_3 and a_4 are arcs in time only. In the NAS's space-time network, these arcs represent potential delay.

(3) An arc from one location in one time period to another location in another time period. In Figure 2, the diagonal arcs—a_5, a_6, a_7, a_8, and a_9—are such arcs. In the NAS's space-time network, these arcs represent potential normal flight, i.e. without airborne delay.

With the information that Figure 1 provides, one may then transform Figure 1's spatial network into the space-time network shown in Figure 3, as described in the following paragraphs. Figure 3's horizontal axis represents two-dimensional space, collapsed into one dimension to simplify visualization in the plane. Points along the horizontal axis are not ordered. The vertical axis represents time. Points along the vertical axis represent discrete time intervals and are ordered.

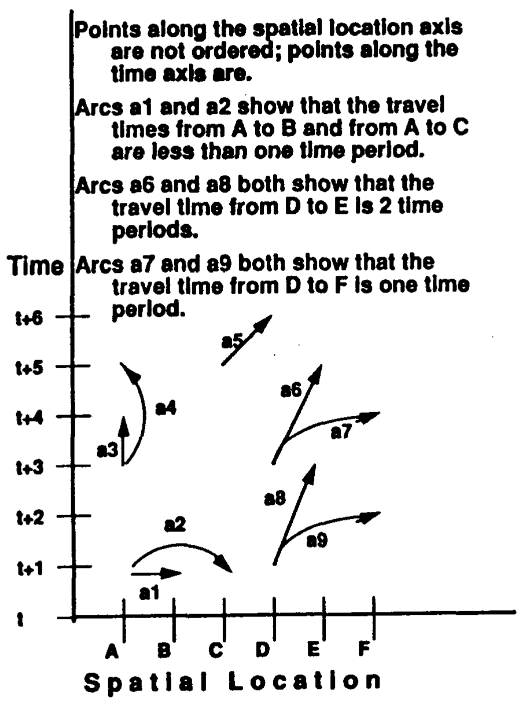


Figure 2. Arcs in space-time.

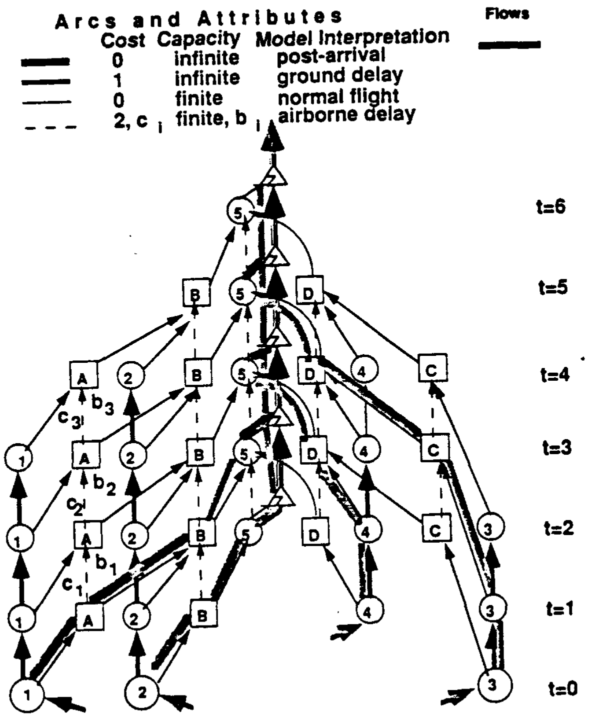


Figure 3. Space-time network with single destination.

It is important to differentiate between arcs and flows. *Arcs* in the space-time network represent *possible* events that the model's *constraints* allow. In Figure 3 all *arcs are black. Flows* occupying those arcs represent a *set of events that can actually occur*, i.e., *a feasible solution set*. In Figure 3 *flows are gray.*

C. Arcs in the Space-Time Network

Different kinds of lines represent different categories of arcs, i.e. potential events. Vertical arcs represent delay and therefore have positive cost. A heavy vertical arc at an origin represents ground delay at that airport. A dashed vertical arc at a waypoint or a destination represents airborne delay. In the example of Figure 3, a ground delay arc of one time period has a cost value of one and infinite capacity; an airborne delay arc of one time period has a cost value of two and finite capacity.

Diagonal arcs represent normal flight and reflect the travel times given in Figure 1. For example, the arcs from vertices (1,t) to (A,t+1) reflect the travel time of one time period from 1 to A. The arcs from vertices (3,t) to (C, t+2) reflects the travel time of two periods from 3 to C.

Vertices (5,t) represent the airspace near airport 5; vertices (Z,t) represent the ground of airport 5. The arcs from vertices (5,t) to (Z,t) represent landing. The triple-lined vertical arcs from vertices (Z,t) to (Z, t+1) represent post-arrival. These are dummy arcs of zero cost and infinite capacity. Their purpose is to allow a single exit vertex, (Z,T) for destination Z, where T is the final period of the time horizon.

D. Flows in the Space-Time Network

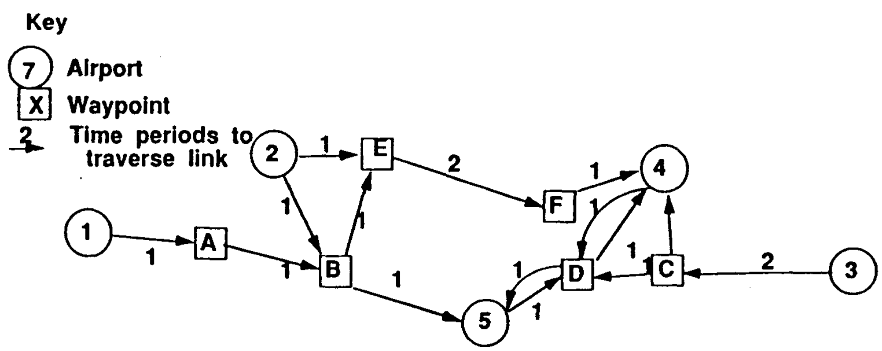
We consider the network of vertices and arcs as a given set of facilities through which flow must pass from origin vertices {(i,t)} to the single destination vertex (Z,T). Inflows represent originally scheduled departures. For example, in Figure 3, the inflow at vertex (4, t=1) represents a set of flights originally scheduled to depart from airport 4 during time period 1.

Figure 3's flows depict a feasible solution in which flights from airports 3 and 4 each experience one period of ground delay, and no flights experience airborne delay.

Flow on the arc from vertex (i,t) to vertex(j,u), denoted x_{it}^{ju}, represents the number of aircraft on that arc. The arc's delay cost parameter is given by c_{it}^{ju}, and its capacity by b_{it}^{ju}. The network cost of a feasible solution is the sum of all delay costs incurred on all arcs, i.e.,

$$\sum_{i}\sum_{t}\sum_{j}\sum_{u} c_{it}^{ju} x_{it}^{ju} .$$

Thus, one can formulate the single-destination minimum delay problem as a minimum cost flow problem with capacitated links, and one can solve it by methods well described in the literature [5], [6], [7], [9].



See Figure 1 for schedules of flights arriving at airport 5.

Flight	Origin	Earliest Depart Time	Waypoints						Destin-ation
5	1	t=0	A, t=1	B, t=2			E, t=3	F, t=5	4, t=6
6	2	t=2					E, t=3	F, t=5	4, t=6
7	3	t=0			C, t=2				4, t=3
8	5	t=1				D, t=2			4, t=3

Figure 4. Spatial network with two destinations.

E. Multiple Destinations in the Space-Time Network

Now suppose we expand this example to include flights to arrive at a second destination, airport 4, as shown in the spatial network of Figure 4. With multiple arrival airports, we define the parameters and variables as follows.

Known parameters

$c_{it}^{ju}(k)$ = delay cost parameter for one aircraft, destined for airport k, on the space-time arc from vertex (i,t) to (j,u);

$r_i^t(k)$ = net inflow at vertex (i,t) of flights destined for k. Thus,

if i is an origin, then $r_i^t(k) > 0$, and $r_i^t(k)$ represents the number of flights, destined for airport k, originally scheduled to depart airport i in time period t;

if i is a waypoint, then $r_i^t(k) = 0$;

if i is a destination airport, then k=i, and $r_i^t(i) < 0$.

b_{it}^{ju} = capacity of the arc from vertex (i,t) to (j,u).

Decision variables

$x_{it}^{ju}(k)$ = flow on the arc from vertex (i,t) to vertex (j,u), of traffic destined for k.

The problem now has the following formulation.

$$\text{Min} \sum_i \sum_t \sum_j \sum_u \sum_k c_{it}^{ju}(k)\, x_{it}^{ju}(k) \qquad (1)$$

(Minimize cost of system-wide delay.)

subject to

$$\sum_j \sum_u x_{it}^{ju}(k) - \sum_j \sum_u x_{ju}^{it}(k) = r_i^t(k) \qquad (2)$$

for all i,t,k;

(Conserve destination-specific flows at each vertex.)

$$\sum_k x_{it}^{ju}(k) \le b_{it}^{ju} \quad \text{for all } i,j,t,u; \qquad (3)$$

(Do not exceed arc capacities.)

$$x_{it}^{ju}(k) \ge 0 \quad \text{for all } i,j,t,u,k. \qquad (4)$$

Because of the summed capacity constraints (3), the multi-destination problem does not conform to the pure minimum cost flow structure. To formulate it as such would mean that the solution algorithm would ignore "who goes where." It might send all of Baltimore's departures to Baltimore.

If one decomposes the M-destinations problem into M independent pure minimum cost flow problems, then one must ignore airway capacities. This would merely repeat the assumptions of present practice, for which we already know the optimal solution.

An M-destinations problem can be formulated as an M-commodity minimum cost flow problem. In this type of problem formulation, the model denotes each commodity separately. For the NAS problem, one may treat all traffic destined for a given destination airport as one commodity. Earlier solution techniques for the multicommodity minimum cost flow problem generated long execution times for moderate-to-large problems [1], [8]; but more recent methods have shown significantly shortened computation time [2], [3].

III. CHARACTERISTICS OF THE SPACE-TIME NETWORK IN TERMS OF EARLIER STATED ASSUMPTIONS

A. Uncertainty

The space-time network model can address uncertainty in several ways. The model generates multiple arcs for the same spatial facility, one arc for each time period. Thus, one can assign multiple delay costs to the same spatial facility, one for each time period; and one can assign a heavier delay penalty to arcs whose capacity values are more certain. Suppose, for example, we wish to show that waypoint A (in Figure 1) has different airborne holding capacities at different times. Figure 3 shows this can be done by assigning capacity b_1 to arc (A,t=1), capacity b_2 to arc (A, t=2) - (A, t=3), and b_3 to arc (A, t=3) - (A, t=4). We can also assign different delay cost parameters, c_i, (i=1,2,3) for each time increment, assigning a larger cost to the earliest period, about which we are least uncertain and a smaller cost to the latest period, about which we are most uncertain.

A second method for handling uncertainty within the space-time network model is to allow the model to update frequently by reading the current state of the system (capacities, flights in progress) as part of the input and adjusting for such "givens" prior to executing the algorithm.

Data aggregation will also eliminate some of the uncertainty. Rather than treating an individual airport as a demand-generating entry point, one can treat a geographic collection of airports as a single demand-generating entry point. Rather than treating a single airway in space as a link or arc, one can consider a geographic collection of airways as a link or arc. A link or arc can be one mile wide or one hundred miles wide.

B. Restrictive Airway Capacities

The space-time network model, since it has a multicommodity flow formulation, specifically addresses the problem of multi-destined traffic occupying the same set of facilities, including airways, of finite capacity. The capacity constraints (3) demonstrate this.

C. Hubbing

To ignore hubbing is to assume that once an aircraft lands at an airport, that its work within the time horizon is done. However, frequently one aircraft's schedule requires a sequence of arrivals and departures at several airports within the time horizon. The space-time network model can accommodate hubbing merely by changing the input data. Thus far, we have considered airports and waypoints as distinct. Aircraft land at airports but not at waypoints. However, one can arrange the data so that the model thinks that a hub airport is really a spatial link between two waypoints. The two waypoints represent, respectively, airport A's arrival and departure runways. These "hub links" possess expected "travel" times. These travel times can in fact represent desired layover time.

D. Deterministic Attempted Entries

The space-time model does not assume that aircraft enter the system, or attempt to enter the system, by a Poisson process. It assumes that the location and time of every attempted entry is known. The model then determines the time of every actual entry, and the model forbids an entry prior to the attempted time. This is consistent with not allowing a flight to depart prior to its originally scheduled time.

E. Dynamic Capacities

This model allows dynamic capacities, i.e., capacity values that vary among time periods. In Figure 3, for example, the capacities of the arcs (1,t=0) - (A,t=1), (1,t=1) - (A,t=2), (1,t=2) - (A, t=3), (1,t=3) - (A, t=4) represent four different values, over time, of the capacity of spatial link 1-A. The model specifically allows these four capacities either to differ or to be the same. It does assume that the four values are known.

F. Execution Speed

Some authors report that multicommodity flow algorithms can speed execution by exploiting information about least cost paths for each origin-destination pair. The air traffic management problem always has known least cost paths in the space-time network, and in fact, known *n*th least cost paths, prior to execution. The proof is contained in MITRE project working documents.

IV. ADDITIONAL CHARACTERISTICS

A. Guaranteed Feasible Solution

We assume (1) that bad weather in any one place does not last forever, and (2) that the time horizon for a given set of flights, S, can be extended for as many periods as we like, with no additional flights attempting to enter the system. Then S always has a feasible flow through the space-time network, because one can add additional time periods, thus providing more options for the same finite set of inflows. The time-extension assumption is probably valid for most cases. Suppose, for example, we are planning the flights originally scheduled to leave between 7 a.m. and 10 p.m. today, and that there are severely reduced capacities in portions of the NAS. The time-extension practice assumes there are no flights scheduled to depart after 10 p.m. This assumption is reasonable, because graveyard shift traffic is light.

B. Multiple Routes

The examples presented so far show only one spatial route between an origin-destination pair. However, multiple routes from an origin to a destination require only minor modification. In this case, diagonal arcs in the space-time network receive positive cost, a function of normal traffic time with no delay. Vertical arcs continue to receive positive cost and represent delay. The objective function to be minimized then becomes total time, with airborne time considered more costly than ground delay time.

V. SUMMARY

We have developed a multicommodity minimum cost flow formulation of the multi-destination, delay minimization problem of managing traffic in the National Airspace System. We have described the model's potential for improving over present practice in terms of six assumptions of present practice. The space-time network model can address (1) uncertainty in capacity values, (2) restrictive airway capacities, and (3) hubbing. None of these first three capabilities are explicitly contained in the current decision tools for assisting in nation-wide ground delay decisions. The minimum cost flow formulation further assumes that (4) attempted entries into the system are deterministic, not Poisson, and that actual entries are controllable. The formulation (5) specifically allows dynamic capacities and (6) possesses good potential for short execution time.

REFERENCES

[1] A. A. Assad, "Multicommodity network flows—a survey," *Networks*, vol. 8, pp. 37-91, 1978.

[2] William J. Carolan, James E. Hill, Jeffery L. Kennington, Sandra Niemi, and Stephen J. Wichmann, "An empirical evaluation of the KORBX algorithms for military airlift operations," *Operations Research*, vol. 38, no. 2, pp. 240-248, 1990.

[3] Ron S. Dembo, and Ulrich Tulowitzki, "Computing equilibria on large multicommodity networks: an application of truncated quadratic programming algorithms," *Networks*, vol. 18, pp. 273-284, 1988.

[4] FAA System Capacity and Requirements Office, *1990-91 Aviation System Capacity Plan*, DOT/FAA/SC-90-1, pp. 1-11, 1988.

[5] L.R. Ford, and D. R. Fulkerson, *Flows in Networks*, Princeton: Princeton University Press, 1962.

[6] Michel Gondran, and Michel Minoux, *Graphs and Algorithms*, Chichester: John Wiley and Sons, 1984.

[7] Paul A. Jensen, and J. Wesley Barnes, *Network Flow Programming*, New York: John Wiley and Sons, 1980.

[8] Jeff L. Kennington, "A survey of linear cost multicommodity network flows," *Operations Research*, Vol. 26, No. 2, pp. 209-235, 1978.

[9] Jeff L. Kennington, and Richard V. Helgason, *Algorithms for Network Programming*, New York: John Wiley and Sons, 1980.