
Transportation Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

# Multimodal Express Package Delivery: A Service Network Design Application

Daeki Kim, Cynthia Barnhart, Keith Ware, Gregory Reinhardt,

# Multimodal Express Package Delivery: A Service Network Design Application

DAEKI KIM

*Optimization Center, Department of Mathematical Science, IBM Thomas J. Watson Research Center, Yorktown Heights, New York 10598*

CYNTHIA BARNHART

*Center for Transportation Studies, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139*

KEITH WARE and GREGORY REINHARDT

*United Parcel Service Airlines, Louisville, Kentucky 40223*



*The focus of this research is to model and solve a large-scale service network design problem involving express package delivery. The objective is to find the cost minimizing movement of packages from their origins to their destinations, given very tight service windows, limited package sort capacity, and a finite number of ground vehicles and aircraft. We have developed a model for large scale transportation service network design problems with time windows. With the use of route-based decision variables, we capture complex cost structures and operating regulations and policies. The poor linear programming bounds limit our ability to solve the problem, so we strengthen our linear programming relaxation by adding valid inequalities. By exploiting problem structure using a specialized network representation and applying a series of novel problem reduction methods, we achieve dramatic decreases in problem size without compromising optimality of the model. Our solution optimization approach synthesizes column and row generation optimization techniques and heuristics to generate solutions to an express package delivery application containing hundreds of thousands of constraints and billions of variables, using only a small fraction of the constraint matrix. The results are potential savings in annual operating costs of tens of millions of dollars, reductions in the fleet size required, dramatic decreases in the time required to develop operating plans, and scenario analysis capabilities for planners and analysts. Through this and additional computational experiments, we conclude that, although state-of-the-art integer programming methods can work well for relatively small, uncongested service network design problems, they must be used in concert with heuristics to be effective for large-scale, congested problems encountered in practice.*

Service network design problems arising at railroads, airlines, trucking firms, and intermodal partnerships require the determination of the cost minimizing or profit maximizing set of services and their schedules, given limited resources. Examples of service network design problems include determining the set of flights and their schedules for an airline; determining the routing and scheduling of tractors and trailers in a trucking operation; and jointly determining the aircraft flights, ground vehicle, and package routes and schedules for a provider of express package delivery service. In this paper, we focus our attention on a particular transportation service network design application involving express package delivery. The objective is to find the cost minimizing movement of packages from their origins to their destinations, given very tight service windows, limited package sort capacity and a finite



**Transportation Science, © 1999 INFORMS**
Vol. 33, No. 4, November 1999

0041-1655/99/3304-0391 $05.00
pp. 391–407, 1526-5447 electronic ISSN

number of ground vehicles and aircraft. Our goal is to develop models and solution procedures so that quality solutions can be generated and strategic planning issues can be analyzed rigorously.

Transportation service network design problems are a variation of the well-studied network design problem. Comprehensive surveys of network design research are presented in MINOUX (1989) and in MAGNANTI and WONG (1984), with relatively recent research on uncapacitated and capacitated network design presented in BALAKRISHNAN, MAGNANTI, and WONG, (1989, 1995), BALAKRISHNAN, MAGNANTI, and MIRCHANDANI (1994a,b), BIENSTOCK and GUNLUK (1994), CHANG and GAVISH (1995), CLARKE and GONG (1995), GENDRON and CRAINIC (1994), MAGNANTI, MIREAULT, and WONG (1986), MAGNANTI, MIRCHANDANI, and VACHANI, (1995), and MEDHI and TIPPER (1995).

There are two major difficulties in applying current network design models and approaches to transportation service network design problems, namely:

1. The interactions among design variables in transportation applications are more complicated than in some other application areas. For example, selecting service between two points implies that a vehicle of some type departs some location and arrives at another. This, in addition, implies that other services must be selected to ensure that conservation of flow is achieved for that vehicle. Another complexity resulting from these interactions involves fixed costs. Because multiple services may be performed by a single vehicle, fixed costs are associated with sets of vehicle movements and not a single movement.
2. A major issue associated with transportation applications is size. In transportation-related scheduling problems, because time and space are both essential ingredients, network size is often huge. State-of-the-art network design methods simply are not designed for problems of the immense size encountered in transportation.

BARNHART and SCHNEUR (1996), BILLHEIMER and GRAY (1973), CRAINIC and ROUSSEAU (1986), FARVOLDEN and POWELL (1994), KIM (1997), KUBY and GRAY (1993), LAMAR, SHEFFI, and POWELL (1990), LEUNG, MAGNANTI, and SINGHAL (1990), NEWTON (1996), POWELL (1986), and POWELL and SHEFFI (1989) focus specifically on transportation service network design. Kuby and Gray examine the effectiveness of hub-and-spoke networks with stopovers and feeders and compare their performance to direct flights into a hub. There are package flow decision variables and aircraft routing decision variables. They assume that one sorting hub exists and the network size is relatively small: they consider only the western United States to avoid computational difficulties.

Barnhart and Schneur develop models and algorithms to solve the particular class of problems we consider in this paper, namely, express package service network design problems. In their model, however, there are several operational restrictions that greatly simplify the problem. Namely, only one hub is allowed, transfer of shipments between aircraft at gateways is not allowed, only one aircraft is allowed to serve a gateway, and there is only ground vehicle feeder service. The result that shipment routings are determined completely by aircraft routes can be exploited to produce a much simpler, and smaller, model.

## CONTRIBUTIONS

IN THIS PAPER, we develop a general modeling and solution approach for large-scale multimodal express package service network design problems. Although our primary objective is to facilitate strategic long-range planning, we design our models and algorithms to be applicable in the contexts of near-term operation planning and market planning.

We present various models (both exact and approximate) and provide problem reduction methods, which reduce model size without compromizing optimality. Further, we dynamically generate decision variables and constraints to minimize memory requirements. The result is an enhanced branch-and-bound solution procedure that is specifically designed for large scale integer programs containing millions, or even billions, of decision variables. Finally, we apply and evaluate our approach using data from a carrier providing multimodal express package delivery.

In this paper, we summarize our contributions as:

- We have developed a representative, solvable model for large-scale multimodal express package delivery. With the use of route-based decision variables, we capture complex cost structures (i.e., non-linear and flow-dependent link or route costs) and satisfy implicitly very complicated rules. By exploiting special problem structure, designing specialized networks, and applying novel problem reduction methods, we achieve dramatic decreases in model size. These techniques transform the problem from intractable to tractable, and do so without losing exactness of the model.
- We have developed an approach that combines

approximate and exact models and methods to find solutions to a transportation service network design problem containing hundreds of thousands of constraints and billions of variables. Our approach synthesizes column and row generation techniques. Column generation overcomes the difficulties of a huge number of decision variables and row generation overcomes the difficulties associated with a weak LP relaxation, namely, the need to generate potentially many violated inequalities. By generating columns and rows on an as-needed basis, solutions can be generated using only a small portion of the constraint matrix.

- As a proof of concept, we applied our models and solution algorithms to a large-scale transportation service network design application. In the case of the company we consider, planning is still done primarily manually, with limited machine intervention. Our approach provides solutions that exhibit potential annual operating cost savings of tens of millions of dollars and a reduction in fleet size compared to those manually generated. Additionally, our approach achieves these results in reasonable run times, dramatically shortenening the current manual, lengthy planning processes and giving planners more time to develop scenarios, analyze the quality of solutions, and plan strategically.
- We conduct various experiments and show that standard integer programming approaches, even those designed for large-scale problems, are unsuccessful in solving the large-scale transportation service network design problems encountered in practice because of excessive run-time and memory requirements, and weak lower bounds. Our method improves upon these standard approaches but additional research is still warranted. Our approach generates high-quality solutions for the easier problems in which congestion is low, but the quality of the solutions generated for problems with higher levels of congestion is hard to assess because the gaps between the best feasible solutions and their lower bounds can be large.

## OUTLINE OF THE PAPER

IN SECTION 1, we describe our express package delivery application, and, in Section 2, we demonstrate how we model the underlying network. Because the network is extremely large, we develop and apply a series of network reduction methods that shrink network size, while maintaining optimality of the model. In Section 3, we present models and solution algorithms for a simplified problem in which the vehicle and aircraft routes and schedules are fixed, and the only decision variables involve the routing of packages. It is essential that this problem be solved quickly because it is a subproblem embedded within the original problem. In Sections 4 and 5, we consider the original problem, that is, the multimodal express package delivery problem, and develop, implement, and evaluate exact and approximate models and solution algorithms. In Section 6, we use our approach to perform various scenario analyses and gain insight about the effects of various parameters on operations and costs.

## 1. EXPRESS PACKAGE DELIVERY: PROBLEM DESCRIPTION

THIS SECTION DESCRIBES the service network design problem of a large provider of express package service. The objective is to find the cost-minimizing movement of packages from their origins to their destinations using a limited number of resources, such that service commitments are satisfied. There are multiple products or service types, defined by the speed of service required. Service occurring overnight, the most costly service, is referred to as Next-Day Service, whereas guaranteed service within 48 hours is called Second-Day Service, and service within 3–5 days is called Deferred Service. In this research, we consider only shipments requiring next day delivery and a single day planning horizon. We enforce conservation of flow of aircraft types by day to ensure that the solution can be repeated daily. We cannot capture day to day variations in volumes, so we use peak daily volumes as our estimated daily demand. This conservative approach to service design is arguably required due to the service-sensitive nature of, and rapid demand growth for, express delivery service.

This service network design problem, like others, requires two types of major decisions: the first is to determine the service network and the second is to determine the flows over the service network. The service network is defined by the movements in time and space of the transportation assets, in this case, jets, propeller aircraft, and ground vehicles. The specific package routings from origins to destinations determine the flows on each movement or link in the service network.

### 1.1 Service Network and Package Flows

Shipments originate and terminate at airport locations, called gateways, where packages are loaded, unloaded, and transferred between different jets, propeller aircraft and/or ground vehicles. From its initial gateway, a package may be flown to at most one additional gateway before it is flown to one of several hubs. Hubs are specialized gateways that perform the additional function of sorting packages. At the hub, a package is unloaded from an inbound (into the hub) jet, sorted, and loaded onto an outbound (out of the hub) jet. Again, it may be flown to at most one intermediate gateway before being flown to its final destination.

Packages are transported from origin to destination using one of the following modes:

1. *Exclusive feeder transportation.* Exclusive feeder transportation involves the transport of packages from their origins to their destinations using only ground vehicles and/or propeller aircraft. Because ground vehicle and propeller aircraft transport are cheap compared to jets, a package will always use exclusive feeder transportation if service requirements can be satisfied. As a result, any package that qualifies is assigned to exclusive feeder service and is removed from further consideration in our model.
2. *Combined transportation.* Typically, service requirements cannot be satisfied using exclusively the slower propeller aircraft and ground vehicle modes. Instead, transport of packages from origins to destinations is achieved using a combination of jet, propeller aircraft, and ground vehicle transport.

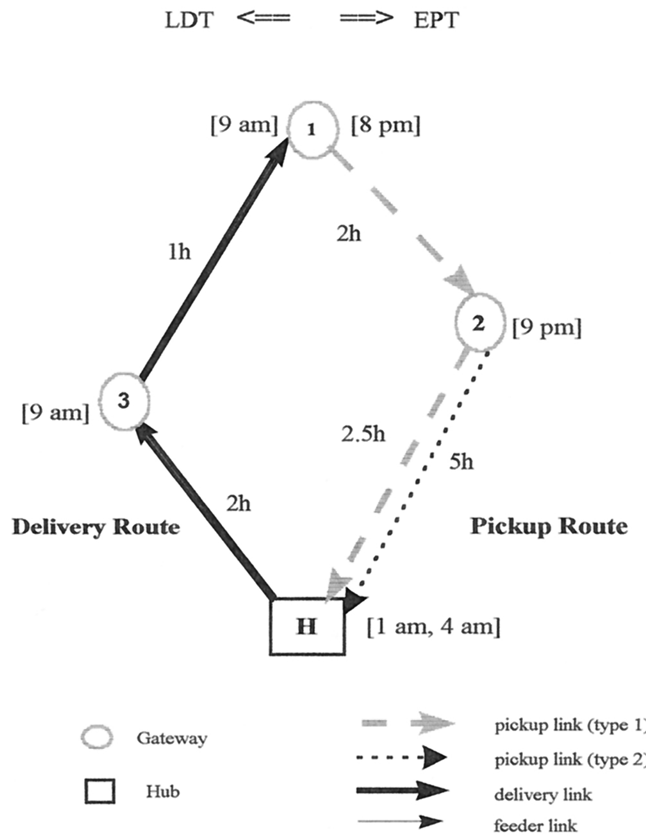


**Fig. 1.** Express package shipment operation.

Associated with gateways are earliest pickup times (EPTs) and latest delivery times (LDTs) that capture service considerations. An EPT denotes the time at which packages will be available for pickup at a gateway. Each gateway's EPT is scheduled as late as possible to allow customers time to prepare their shipments, but early enough so that delivery service standards can be met. A gateway's LDT denotes the time by which all packages must be delivered to the location to satisfy delivery standards. In setting EPTs and LDTs, we also consider time windows at hubs designating the start and end sort times.

An aircraft route can be decomposed into two distinct components, a pickup route and a delivery route, as depicted in Figure 1. A pickup route typically departs from some gateway in the early evening and is restricted to contain at most one intermediate stop before its final stop at a hub. A delivery route begins at a hub, typically departing in the early morning, and stops at most at two gateways.

The number of stops on a pickup or delivery route is restricted to three to limit the potential for schedule problems arising in this hub-and-spoke network. Fewer take-offs and landings result in a reduction in the expected schedule slippage. Given the time-sensitive nature of the express package delivery operation, robustness of operation is critical.

A pickup or delivery route is feasible only if all service requirements can be fully satisfied. For example, in Figure 1, we have two aircraft pickup routes, one is ($1 \rightarrow 2 \rightarrow H$) by aircraft type 1 and the other is ($2 \rightarrow H$) by aircraft type 2. Numbers beside aircraft legs denote the aircraft-specific transportation time in hours between two locations. Numbers in the brackets denote the earliest pickup time (in case of pickup routes) or latest delivery time (in case of delivery routes) at gateways. At hubs, sort start times and sort end times are specified. Pickup route ($1 \rightarrow 2 \rightarrow H$) by aircraft type 1 is feasible because the earliest arrival time at the hub [8:00 PM (EPT of gateway 1) + 4.5 hour (travel time) = 12:30 AM] is before the hub sort end time of 4:00 AM. Likewise, pickup route ($2 \rightarrow H$) by aircraft type 2 is feasible. If the sort end time were 1:00 AM instead, the pickup route ($2 \rightarrow H$) by aircraft type 2 would not be feasible because its earliest arrival time to the hub is too late. Pickup route ($1 \rightarrow 2 \rightarrow H$) by aircraft type 1, however, is still feasible. The delivery route ($H \rightarrow$

3 → 1) is feasible because the earliest arrival time at gateway 3 (sort start time of H at 1:00 AM + 3 hour travel time = 4:00 AM) is earlier than its latest delivery time (LDT) of 9:00 AM. Again, if the LDT of gateway 3 were 3:00 AM instead, then this delivery route would not be feasible.

Because shipments can be transferred between aircraft at gateways, the number of shipment routes often far exceeds the number of aircraft routes. Figure 1 illustrates this fact. There are three possible shipment routes and only two aircraft routes.

### 1.2 Costs

The costs of an express package operation can be expressed as the sum of jet aircraft costs, feeder route costs and package handling costs. For a given flight leg and aircraft type, jet aircraft operating costs are the sum of associated fuel cost, crew cost, cycle cost (that is, the cost of take-off and landing), and maintenance cost. Depending on the nature of the model, jet ownership cost may or may not be included. Ownership cost is not included for near-term planning because the company already owns the aircraft; however, a fixed cost per aircraft is included if the model is to be used in a more strategic manner. For example, if the model is to be used to determine future fleet composition. The costs for ground movements and feeder air movements are expressed for a package on a given leg as a total cost per package, for each type of movement (ground or propeller aircraft). Package handling costs are expressed per package for each gateway and hub location.

## 2. NETWORK REPRESENTATIONS

IN MANY TRANSPORTATION-related scheduling problems, time and space are essential elements. To evaluate properly the system, the conventional modeling practice is to build dynamic models. At their core, these models have a time–space structure, with nodes representing time and space and arcs representing movement in time and possibly, space. For our application specifically, each node in the time–space network corresponds to the origin or destination of a jet or feeder movement at some point in time, and each arc corresponds to a jet or feeder movement at a particular time. The drawback to this time–space representation is that it leads to an enormous network. So, we exploit the structure of our express package delivery problem, namely, the fact that at most three locations can be visited on a pickup or delivery route, and represent all movements and feasible package routings using a network of drastically reduced size. We refer to this mechanism for reducing network size as the derived schedule approach because it exploits the fact that we do not need to know the exact schedule for each move, we need only to know that there exists a feasible schedule.

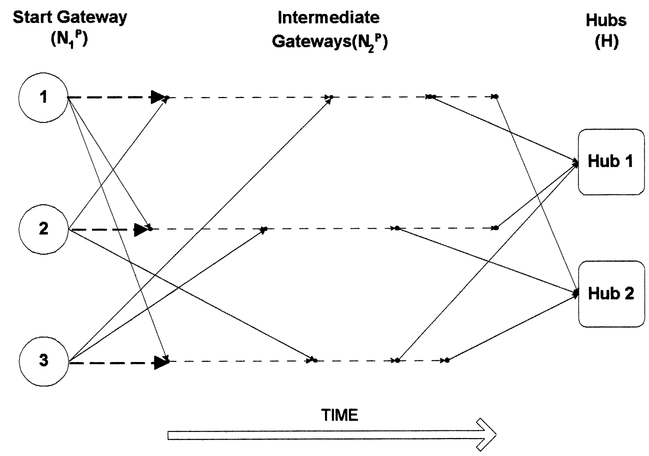


**Fig. 2.** Derived pickup network.

### 2.1 Derived Schedule Jet and Feeder Networks

For each jet type, we generate a pickup and a delivery derived schedule network, as shown in Figures 2 and 3. For now, consider one fleet type, i.e., a particular type of jet, propeller aircraft, or ground vehicle (we omit the notation designating fleet type for ease of exposition.) Each pickup route consists of at most two legs: the first from a starting gateway to either a hub or an intermediate gateway, and the second, if it exists, from an intermediate gateway to a hub. Similarly, each delivery route consists of at most two legs: the first from a hub to a gateway, and the second, if it exists, from this intermediate gate-

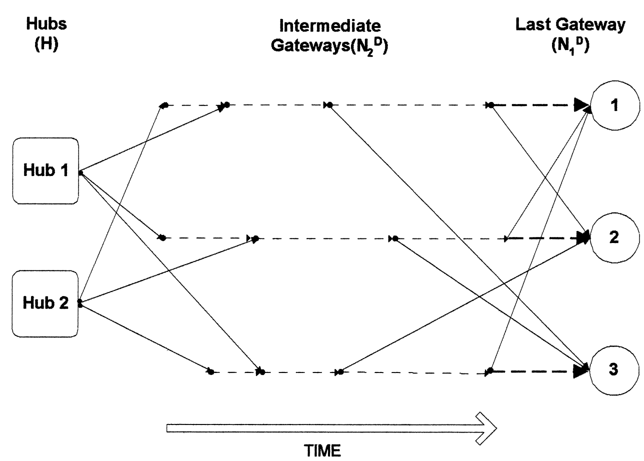


**Fig. 3.** Derived delivery network.

way to another gateway. In our derived schedule network for a single fleet, we represent routes using the following node and arc sets:

| | |
|---|---|
| $N$: | set of network nodes. |
| $A$: | set of network arcs. |
| $N_1^P$ ($N_1^D$) ($\subset N$): | gateway nodes representing first (last) stops on pickup (delivery) routes. |
| $N_2^P$ ($N_2^D$) ($\subset N$): | pair nodes representing intermediate stops on pickup (delivery) routes. |
| $H$: | hub nodes representing last (first) stops on pickup (delivery) routes. |
| $L^P$ ($L^D$) ($\subset A$): | leg arcs representing legs in pickup (delivery) routes. |
| $G^P$ ($G^D$) ($\subset A$): | ground arcs representing periods during which aircraft/vehicle location does not change in pickup (delivery) routes. |

The sets $N_1^P$ and $N_1^D$ each contain one node for each gateway. For ease of exposition, we let gateway node $i \in N_1^P \cup N_1^D$ correspond to gateway $i$. $H$ contains one node for each hub, and similarly, hub node $h$ corresponds to hub $h$. $N_2^P$ and $N_2^D$ each contain one node for each ordered pair of gateways and each ordered pair consisting first of a gateway and second, a hub, for a total of $|G|^2 + |G||H|$ nodes where $|G|$ is the number of gateways and $|H|$ is the number of hubs.

Each gateway node $i \in N_1^P$ has an associated time $t_i$ = EPT($i$) + time to load packages at $i$, representing the earliest time that an aircraft/vehicle may begin its pickup route at gateway $i$. Similarly, for each $i \in N_1^D$, $t_i$ = LDT($i$) − time to unload packages at $i$, reflects the latest time that an aircraft/vehicle may end its delivery route at gateway $i$ so that service requirements may be satisfied. Each hub node $h \in H$ for hub $h$ has an associated time $t_h$ = SET($h$), the sort end time at $h$. All pickup routes must arrive at $h$ before $t_h$ − the time to unload packages at $h$, and all delivery routes may not depart $h$ before $t_h$ + the time to load packages at $h$. The time $t_i$ of the pair node $i \in N_2^P$ for gateways $m$ and $n$, is the earliest time that an aircraft/vehicle initiating its pickup route at $m$ can depart its intermediate stop at $n$, i.e., $t_i$ = max[EPT($n$), $t_m$ + travel time($m \to n$)]. If $m = n$, then $t_i = t_m$. Similarly, the time $t_i$ of the pair node $i \in N_2^P$ for gateway $n$ and hub $h$, is the latest time that an aircraft/vehicle on its pickup route may depart $n$ to arrive at $h$ before its sort end time, i.e., $t_i$ = max[EPT($n$), $t_h$ − travel time($n \to h$) − time to unload packages at $h$]. With regard to delivery routes, the time $t_i$ of the pair node $i \in N_2^D$ for hub $h$ and gateway $n$, is the earliest time that an aircraft/vehicle initiating its delivery route at $h$ may arrive at the intermediate stop at $n$, i.e., $t_i$ = min[LDT($n$), $t_h$ + travel time($h \to n$) + time to load packages at $h$]. Finally, the time $t_i$ of the pair node $i \in N_2^D$ for gateway $n$ and gateway $m$, is the latest time that an aircraft/vehicle may leave $n$ on its delivery route to $m$ to satisfy service requirements at $m$, i.e., $t_i = t_m$ − travel time($n \to m$). Again, if $n = m$, then $t_i = t_m$.

We construct the set of leg arcs for pickup routes as follows:

- For each $i \in N_1^P$ and $l \in N_2^P$, we include an arc from gateway node $i$ to a pair node $l$ for the ordered pair $(i, j)$, for any gateway $j$ as long as the travel time $(i \to j)$ does not exceed the maximum time allowed.
- For each $l \in N_2^P$ and $h \in H$, we include an arc to hub node $h$ from pair node $l$ for the ordered pair $(i, h)$, for any gateway $i$, $i \neq h$, if the travel time $(i \to h)$ does not exceed the maximum time allowed and [$t_h$ − travel time($i \to h$) − time to unload packages at $h$] $\geqslant t_l$. (If this inequality is not satisfied, it is not feasible to service gateway $i$ through hub $h$.)

Similarly, for delivery routes, we build the set of leg arcs as follows:

- For each $l \in N_2^D$ and $h \in H$, we include an arc from hub node $h$ to pair node $l$ for the ordered pair $(h, j)$, for any gateway $j$, $h \neq j$, if the travel time $(h \to j)$ does not exceed the maximum time allowed and $t_l \geqslant t_h$ + travel time($h \to j$) + time to load packages at $h$. (If this inequality is not satisfied, it is not feasible to service gateway $j$ through hub $h$.)
- For each $i \in N_1^D$ and $l \in N_2^D$, we include an arc to gateway node $i$ from pair node $l$ for the ordered pair $(i, j)$, for all gateways $j$, if $t_l \geqslant t_j$ + travel time($j \to i$). (If this inequality is not satisfied, it is not feasible to service gateway $i$ through gateway $j$.)

Finally, we construct the set of ground arcs in the derived schedule network as follows:

- We assign gateway node $i \in N_1^P$ ($N_1^D$) to gateway $i$, pair node $l \in N_2^D$ for the ordered pair $(i, j)$ of gateways to gateway $j$, pair node $l \in N_2^P$ for the ordered pair $(j, h)$ for gateway $j$ and hub $h$ to gateway $j$, hub node $h \in H$ to hub $h$, pair node $l \in N_2^D$ for the ordered pair $(i, j)$ of gateways to gateway $i$, and pair node $l \in N_2^D$ for the ordered pair $(h, j)$ for hub $h$ and gateway $j$ to $j$.
- We sort all nodes assigned to gateway $j$ in increasing order of time and place a ground arc

between each pair of successive nodes, for each $j$. The from node of any ground arc has time not later than its to node.

The cost of each leg arc includes fleet and package routing costs. Fleet routing costs are composed of fixed and variable operating costs, and these costs vary by fleet. Package routing costs are the handling costs incurred by carrying along a package. The capacity of each leg arc corresponds to the capacity of fleet type used. All cost elements for each ground arc are set to zero, and the capacity of each ground arc is infinite.

*Attributes of the Derived Schedule Jet and Feeder Networks*

We construct one derived schedule network for propeller aircraft, one for ground vehicles, and one for each type of jet, because times associated with gateway and pair nodes vary for fleet type. By design, the derived schedule network for any fleet type $f$ has the following characteristics:

- Each network path beginning at a gateway node and ending at a hub node corresponds to a feasible pickup route for fleet type $f$.
- Each network path beginning at a hub node and ending at a gateway node corresponds to a feasible delivery route for fleet type $f$.
- Each feasible pickup or delivery route for fleet type $f$ is represented by a path in the network. Pickup (delivery) routes that contain only one leg, e.g., a pickup route from gateway $i$ to hub $h$, are represented by a path containing the leg arc from gateway node $i$ to pair node $l$ for the pair ($i$, $i$) of gateways, the ground arc from pair node $l$ to pair node $m$ for the pair ($i$, $h$) representing gateway $i$ and hub $h$, and the leg arc from pair node $m$ to hub node $h$.
- Each route has potentially many feasible schedules. To illustrate, consider a pickup route ($i - j - h$) represented by gateway node $i$, pair node $m$, pair node $n$, and hub node $h$. This route can begin as early as $t_i$ and as late as $t_i$ plus the amount of slack in the route, that is, $t_i + (t_n - t_m)$, where $(t_n - t_m)$ is nonnegative for each feasible route. If the route start time is $t_i + \partial$, for $\partial \leqslant (t_n - t_m)$, then the time of departure from intermediate gateway $j$ is $(t_m + \partial + \gamma)$, where $\gamma \leqslant (t_n - t_m - \partial)$. Finally, the arrival at $h$ will be $t_m + \partial + \gamma +$ travel time $(j - h)$.

## 2.2 Package Network

We represent the set of potential pickup (delivery) routes and schedules for packages using a package network, constructed by merging the pickup (delivery) derived schedule network for each fleet type. The package network is constructed as follows:

- Merge gateway and hub nodes from the derived schedule networks for each fleet:
  1. For each fleet $f \in F$, map its gateway node $i$, for all $i \in N_1^P$ ($i \in N_1^D$), into a single gateway node $i$ in the pickup (delivery) package network.
  2. For each fleet $f \in F$, map its hub node $h$, for all $h \in H$, into a single hub node $h$ in the package network.
- Add to the package network all leg arcs and pair nodes in the derived schedule network for fleet type $f$, for all $f \in F$.
- Sort all pair nodes at a single location in increasing order of time, and add ground arcs between successive pair nodes at a location, for each location.

Arc costs and capacities in the package network are as previously defined for the jet- and feeder-derived schedule networks.

*Attributes of the Package Network*

Like the jet and feeder networks, the package network has the following characteristics:

- Each network path beginning at any gateway node $i$, containing a hub node and ending at any gateway node $j$ is a feasible route for packages originating at $i$ and destined for $j$.
- Each feasible route for any package originating at $i$ and destined for $j$ is represented by a path in the package network.
- Each route has potentially many feasible schedules.

## 2.3 Network Reduction Methods

We can reduce dramatically the size of time-line networks using a network reduction method called node consolidation, previously applied by HANE, et al.(1995) in an application for the airline industry. Although our derived schedule service network avoids the massive explosion in the number of nodes and links of a conventional time–space network, our network is still very large. Specifically, one data set describing our particular application has 75,944 nodes and 151,002 links. When we apply node consolidation to our network, we achieve a reduced-size network containing 2,400 nodes and 72,144 links. This dramatic decrease in size can be explained by examining the particular structure of our network. Each pair node is associated with either the arrival or the departure of a leg arc. And, pair nodes for arrivals typically have associated time earlier than pair nodes for departures. The result is massive node consolidation and corresponding link elimination.

After node consolidation, there are many instances of parallel arcs between two nodes in the network. These arcs represent movements of different fleets between locations at roughly the same time. It is advantageous from the point of view of the formulation to represent these replicate arcs by a single arc, recording the set of relevant characteristics of the various equipment types that can be assigned to the corresponding leg. Applying link consolidation to our network of 72,144 links (after node consolidation), we reduced the number of links to 15,355.

## 3. THE PACKAGE FLOW PROBLEM

GIVEN A SERVICE NETWORK, the objective of the package flow problem is to find the minimum cost flow of packages from their origins to their destinations satisfying service commitments and network capacity. Alternatively, for a given service network, the objective may be to determine whether a set of package flows can be serviced and, if not, to determine where the network lacks sufficient capacity. Yet another objective may be to determine if service standards can still be met if they are changed, for some or all packages. Whichever objective, these problems all can be cast as linear multicommodity network flow problems. Multicommodity flow problems have been studied extensively, with AHUJA, MAGNANTI, and ORLIN (1993), ASSAD (1978), and KENNINGTON (1978) presenting comprehensive surveys.

JONES, et al. (1993) investigate the impact of problem formulation on Dantzig–Wolfe decomposition solution approaches for the multicommodity network flow problem. They compare a node–arc formulation, a path formulation, and a tree formulation. The formulations vary in their selection of decision variables: the node–arc formulation has one variable for each arc–commodity pair; the path formulation has one variable for each path–commodity pair and a commodity is defined by an origin–destination pair; and the tree formulation has one variable for each tree–commodity pair and a commodity is defined by a single origin and possibly, multiple destinations. In the interest of brevity, in the next section, we detail the tree formulation and solution, assuming that the node–arc and path formulations and solutions are both well known and detailed in several sources, including Ahuja, Magnanti, and Orlin (1993).

### 3.1 A Multicommodity Flow Tree Formulation

Even the path formulation with only a subset of columns may be too large to solve. The issue is that the number of constraints may be excessive. To remedy this, if arc costs do not vary by commodity (where here a commodity $k$ with demand $b^k$ is defined by an origin $\mathrm{O}(k)$ and destination $\mathrm{D}(k)$), we can reduce the number of constraints in the model using an alternative, yet equivalent, formulation called the tree formulation, as presented in Jones, et al. (1993). The idea is to aggregate commodities with the same origin into a single super commodity. (Alternatively, super commodities can be aggregated by destination.) Each super commodity $s_{\mathrm{O}(k)}$ in the set $S$ of super commodities corresponds to the set of commodities $k \in K$ originating at $\mathrm{O}(k)$. Because each commodity $k \in s_{\mathrm{O}(k)}$ may flow along several paths in the set of paths $P^k$ for commodity $k$, each super commodity $s_{\mathrm{O}(k)}$ can flow along several trees, denoted by the set of trees $\boldsymbol{Q}^{s_{\mathrm{O}(k)}}$. We let $\Gamma_{ij}^q$ equal 1 if tree $q$ contains arc $(i, j)$ and equal 0 otherwise, and $u_{ij}$ equal the capacity of arc $ij$. Each tree $q \in \boldsymbol{Q}^{s_{\mathrm{O}(k)}}$ is rooted at $\mathrm{O}(k)$ and contains one $\mathrm{O}(k)$ to $\mathrm{D}(k)$ path, denoted $p_q^k$, for only those $k \in s_{\mathrm{O}(k)}$. The flow on each path in a tree is a constant $w$, with $0 \leqslant w \leqslant 1$, of $b^k$ for every commodity $k$ in the tree.

Before presenting the formulation, we define these additional notations.

**Sets**

$s_i$ $= \{k | \mathrm{O}(k) = i, k \in K\}$, a super commodity composed of all commodities $k \in K$ originating at $i$

$S\ (\ni s_i)$ $= \{s_i | i \in \mathrm{O}(k),\ k \in K\}$, set of super commodities $s_i$, $i \in \mathrm{O}(k)$, for all $k \in K$

$q$ $= \{(i, j) | (i, j) \in p_q^k,\ p_q^k \in P^k,\ k \in s_{\mathrm{O}(k)}\}$, a tree composed of paths $p_q^k$, $p_q^k \in P^k$, $k \in s_{\mathrm{O}(k)}$

$p_q^k$: a path from $\mathrm{O}(k)$ to $\mathrm{D}(k)$ in tree $q$

$\boldsymbol{Q}^{s_{(\mathrm{O}(k)}}\ (\ni q)$: set of trees originating at $\mathrm{O}(k)$, for all super commodities $s_{\mathrm{O}(k)} \in S$

**Parameters**

$c_q^{s_{\mathrm{O}(k)}}$: total cost of sending $b^k$ units of $k \in s_{\mathrm{O}(k)}$ along the path $p_q^k$, for all $q \in \boldsymbol{Q}^{s_{\mathrm{O}(k)}}$ and all $s_{\mathrm{O}(k)} \in S$

**Decision Variables**

$w_q^{s_{\mathrm{O}(k)}}$: *fraction of* $b^k$ assigned to path $p_q^k$ for each $k \in s_{\mathrm{O}(k)}$, for all $q \in \boldsymbol{Q}^{s_{\mathrm{O}(k)}}$ and all $s_{\mathrm{O}(k)} \in S$

The tree formulation for the multicommodity flow problem is

$$\min \sum_{s_{\mathrm{O}(k)} \in S} \sum_{q \in \boldsymbol{Q}^{s_{\mathrm{O}(k)}}} c_q^{s_{\mathrm{O}(k)}} w_q^{s_{\mathrm{O}(k)}} \qquad (1)$$

$$\sum_{q \in Q^{s_{O(k)}}} w_q^{s_{O(k)}} = 1 \quad \text{for all } s_{O(k)} \in S \tag{2}$$

$$\sum_{s_{O(k)} \in S} \sum_{q \in Q^{s_{O(k)}}} \left( \sum_{k \in s_{O(k)}} \Gamma_{ij}^q b^k \right) w_q^{s_{O(k)}} \leq u_{ij} \quad \text{for all } (i, j) \in A \tag{3}$$

$$w_q^{s_{O(k)}} \geq 0 \quad \text{for all } q \in Q^{s_{O(k)}}, \quad \text{for all } s_{O(k)} \in S. \tag{4}$$

As before, the objective is to minimize the total cost of flowing commodities from their origins to their destinations. Constraints 2 ensure that exactly the total quantity of each commodity is assigned to the network. Constraints 3, the bundle constraints, ensure that the total arc flow does not exceed arc capacity, for each $(i, j) \in A$ and constraints 4 ensure that tree flows are all nonnegative.

The number of constraints in the tree formulation ($|S| + |A|$ with $|S|$ equal to the number of super commodities and $|A|$ the number of arcs) is reduced compared to the number in the path formulation ($|K| + |A|$ where $|K|$ is the number of commodities). The number of variables in the tree formulation, however, is increased exponentially relative to the number in the path formulation (the number of variables in the tree formulation is the number of ways to combine path variables for each commodity.)

### 3.2 Solution of the Tree Formulation

As with the path formulation, column generation can be used to solve the tree formulation. Column generation methods achieve optimal solutions to LPs containing a huge number of decision variables, without explicitly considering all decision variables. (Detailed descriptions are provided in several sources, including Ahuja, Magnanti, and Orlin 1993.) The master problem is the original problem containing all decision variables, and the restricted master problem (RMP) is the master problem with many of the variables eliminated. To check if an optimal solution to the RMP is also optimal for the master problem, a subproblem, called the pricing problem, is solved. The pricing problem for the tree formulation has the following form:

$$z^* = \underset{q \in Q^{s_{O(k)}}, s_{O(k)} \in S}{\text{Minimize}}\; c_q^{s_{O(k)}} - \sigma_{s_{O(k)}} - \sum_{(i,j) \in A} \left( \sum_{k \in s_{O(k)}} \Gamma_{ij}^q b^k \right) \pi_{ij},$$

where $\pi$ are non-positive and $(\sigma, \pi)$ are the dual variables associated with constraints 2 and 3, respectively. Again, because $z^*$ represents the minimum reduced cost value, if $z^* \geq 0$, then the tree formulation LP is solved. If $z^* < 0$, however, the solution to the pricing problem identifies a tree that may improve the current solution if added to the restricted LP.

Letting $\delta_{ij}^{p_q^k}$ equal 1 if arc $ij$ is on path $p$ and equal 0 otherwise, $c_p^k$ equal the cost of sending one unit of flow of $k \in K$ along path $p \in P^k$, and $c_{ij}^k$ equal the cost of sending one unit of flow of $k \in K$ along arc $ij \in A$, we rewrite the pricing problem as:

$$\begin{aligned}
z^* &= \underset{q \in Q^{s_{O(k)}}, s_{O(k)} \in S}{\text{Minimize}} \sum_{k \in s_{O(k)}} c_{p_q^k}^k b^k - \sum_{(i,j) \in A} \left( \sum_{k \in s_{O(k)}} \Gamma_{ij}^q b^k \right) \pi_{ij} - \sigma_{s_{O(k)}} \\
&= \underset{q \in Q^{s_{O(k)}}, s_{O(k)} \in S}{\text{Minimize}} \sum_{k \in s_{O(k)}} c_{p_q^k}^k b^k - \sum_{(i,j) \in A} \left( \sum_{k \in s_{O(k)}} \delta_{ij}^{p_q^k} b^k \right) \pi_{ij} - \sigma_{s_{O(k)}} \\
&= \underset{q \in Q^{s_{O(k)}}, s_{O(k)} \in S}{\text{Minimize}} \sum_{k \in s_{O(k)}} \sum_{(i,j) \in A} c_{ij}^k b^k \delta_{ij}^{p_q^k} - \sum_{(i,j) \in A} \left( \sum_{k \in s_{O(k)}} \delta_{ij}^{p_q^k} b^k \right) \pi_{ij} - \sigma_{s_{O(k)}} \\
&= \underset{q \in Q^{s_{O(k)}}, s_{O(k)} \in S}{\text{Minimize}} \sum_{(i,j) \in A} \sum_{k \in s_{O(k)}} (c_{ij}^k - \pi_{ij}) \delta_{ij}^{p_q^k} b^k - \sigma_{s_{O(k)}}.
\end{aligned}$$

If for every $s_{O(k)} \in S$,

$$\underset{q \in Q^{s_{O(k)}}}{\text{Minimize}} \sum_{(i,j) \in A} \sum_{k \in s_{O(k)}} (c_{ij}^k - \pi_{ij}) \delta_{ij}^{p_q^k} b^k \geq \sigma_{s_{O(k)}},$$

then optimality is achieved. This can be checked by solving a shortest path problem for each super commodity $s_{O(k)} \in S$ over the service network with modified link costs of $(c_{ij}^k - \pi_{ij})$, for all $(i, j) \in A$.

Solving the pricing problem for the tree formulation, like the path formulation, requires only the solution of one shortest path problem for each super commodity $sO(k) \in S$, because shortest path algorithms produce a shortest path tree from an origin to all destinations. So, even with the change in variables from paths to trees, the pricing problem remains the same.

### 3.3 Computational Results

Using data from a large express package delivery operation, we compare the solution of the node–arc, path and tree formulations, using an IBM RS/6000, 370 workstation with 256 MB RAM. The carrier's service network contains 807 nodes, 1,363 links, and 17,539 origin/destination-specific commodities. The number of super commodities, defined by the number of origins, is 136. Only 292 out of the 1363 links are capacitated because ground movement capacity is effectively unlimited (because it is relatively cheap), and only aircraft capacity is limited.

As a minor modeling enhancement, we deleted all uncapacitated links from the bundle constraints because their inclusion is not necessary. This reduces the number of constraints for the path and tree formulations from 18,902 to 17,832 and from 1,499 to 428, respectively. It is important to keep in mind that the number of constraints in the problem plays a critical role in determining the speed of the LP solver.

We were unable to solve the node–arc formulation due to limitations in the size of our random access memory. We solved, however, the path formulation in about 40 minutes and, contrary to the experiences reported by Jones, et al. (1993), we solved the tree-based model within 1 minute. We believe that the tree model was solved so quickly in this application because the network is relatively uncongested. Because the express package service problem has tight service standards, maximum consolidation and utilization of capacity is not possible. Instead, planes are forced to fly to satisfy service commitments. Lack of congestion together with a relatively small number of constraints (428 in the tree formulation versus 17,832 in the path formulation), allows the column generation algorithm to terminate quickly without tailing.

## 4. EXPRESS PACKAGE SERVICE NETWORK DESIGN: EXACT MODELS AND SOLUTIONS

IN THIS SECTION, we expand our package flow model and algorithm to create models and algorithms for designing the service network for a large express package delivery operation. The model inputs include package movement requirements, a network of potential feeder and jet movements, fleet composition and characteristics, and sort capabilities and characteristics. The fleet composition is given, with an option available to lease additional aircraft. All aircraft, gateway, and hub operating characteristics, such as range, speed, runway length, sort capacity, etc., are considered fixed.

In the context of multimodal express package service network design (SND), the design variables in the service network design models represent jet and feeder movements, and the flow variables describe package flows. In each case, these variables correspond to routes, and not single legs. We adopt this route-based variable definition for three major reasons:

1. We easily can capture fixed costs and other nonlinear costs because each aircraft/vehicle is assigned to at most one design variable.
2. Complex restrictions on aircraft, vehicle, or package routings are satisfied by each variable and need not be represented by constraints that are often difficult or even impossible to write.
3. The node–arc express package delivery formulation is prohibitively large for our application (with hundreds of millions of decision variables and more than 42 million constraints) and impossible to solve.

The disadvantage of route-based variables, however, is that the number of variables explodes, and so, specialized solution procedures, like column generation must be used.

Given our variable definitions, the objective of our SND problem is to find:

1. Jet and feeder routes that minimize fixed design and variable operating costs; and
2. Package flow routes that satisfy customer demands without violating service restrictions and capacity limits imposed on each design leg.

To model this, we augment traditional network design formulations to reflect particular express package delivery operational constraints, including fleet balance, fleet size, hub sort capacity, hub landing capacity, and connectivity. Additionally, as is commonly done, we enhance the solvability of our model by including cutset inequalities that strengthen the LP relaxation of our model. All of these constraints are detailed in the following sections.

### 4.1 Fleet Balance

Fleet balance constraints force the number of routes into a location for a particular fleet to be equal to the number out, for each fleet $f \in F$,

$$\sum_{r \in R_P^f \cup R_D^f} \beta_i^r y_r^f = 0 \quad \text{for all } i \in N, \quad f \in F, \qquad (5)$$

where $N$ is the set of network nodes, $F$ is the set of fleets, $R_p^f$ ($R_D^f$) is the set of pickup (delivery) routes assigned to fleet $f$, $\beta_i^r$ is equal to 1 if route $r$ ends at node $i$, is equal to $-1$ if route $r$ begins at $i$, and is equal to 0 otherwise. $y_r^f$, a decision variable, is equal to 1, 2, . . . , $n$ if route $r$ is flown 1, 2, . . . , $n$ times by fleet $f$, and is equal to 0 otherwise.

### 4.2 Fleet Size

Each fleet has a limited number of aircraft/vehicles, and so, the number of aircraft/vehicles of a particular type used may not exceed the number available. We model these fleet size constraints by restricting the number of pickup routes or the number of delivery routes for fleet $f$ to be less than $n^f$, the fleet size, in $f \in F$,

$$\sum_{r \in R_P^f} y_r^f \leq n^f \quad f \in F;$$

or

$$\sum_{r \in R_D^f} y_r^f \leq n^f \quad f \in F.$$

We do not need to include both sets of constraints because fleet balance is ensured by Eq. 5.

### 4.3 Hub Sort Capacity

Each hub is able to sort a limited number of packages per time period. To model these hub sort capacity constraints, for each hub, we divide the total time during which sorting is performed into equal-length intervals $t = \{1, 2, \ldots, T\}$. We let $P_i^t$ be the set of package routes with earliest arrival time at hub $i$ later than or equal to the start time of interval $t \in \{1, 2, \ldots, T\}$ and $e_i^m$ be the sort capacity of hub $i$ during interval $t \in T$. Then, the hub sort capacity constraints are

$$\sum_{s \in S} \sum_{q \in Q^s} \left\{ \sum_{k \in K} \sum_{p \in P^k \cap P_i^t} (\delta_{ij}^p b^k) \right\} w_q^s \leq \sum_{m=t}^{T} e_i^m \quad i \in H, \quad t \in \{1, 2, \ldots, T\}.$$

To illustrate, assume that there are five packages arriving at a hub with 3 hours of sort time and a sort capacity of 2 packages per hour. The corresponding hub sort capacity constraints are:

time interval 1: $w_1 + w_2 + w_3 + w_4 + w_5 \leq 6$
time interval 2: $w_3 + w_4 + w_5 \leq 4$
time interval 3: $w_4 + w_5 \leq 2$.

### 4.4 Hub Landing Capacity

Each hub has a landing capacity that limits the number of aircraft that can land in an interval of time. To model these hub landing capacity constraints, for each hub, we divide the total time during which aircraft are arriving into equal-length intervals $t = \{1, 2, \ldots, T\}$. We let $L_i^t$ denote the set of pickup routes with earliest arrival time at hub $i$ no earlier than the start time of interval $t$, and $a_i^t$ be the maximum number of aircraft that can land (i.e., the landing capacity) at hub $i$ during interval $t$, for all $t \in T$. Then, the hub landing capacity constraints can be represented as

$$\sum_{f \in F} \sum_{r \in R_P^f \cap L_i^t} y_r^f \leq \sum_{m=t}^{T} a_i^m \quad i \in H, \quad t \in \{1, 2, \ldots, T\}.$$

### 4.5 Connectivity

Many package delivery networks contain a major hub (for example, Memphis, Tennessee for Federal Express, Louisville, Kentucky for United Parcel Service, and Wilmington, Ohio for Airborne Express) and one or more regional hubs. For operational reasons, some carriers require all-point service to and from their major hub. This means that, for every origin location, there should be at least one route from that location to the major hub and similarly, for every destination location, there should be at least one route to it from the major hub. The idea is that, with all the aircraft/vehicles arriving and departing the major hub, achieving timely service should still be possible, even with service disruptions.

We let $V_P^i$ represent the set of pickup routes including gateway $i$ that end at the major hub and let $V_D^i$ represent the set of delivery routes including gateway $i$ that begin at the major hub. We model these all-point service connectivity constraints for each gateway $i$ in the set of gateways $\mathcal{G}$:

$$\sum_{f \in F} \sum_{r \in R_P^f \cap V_P^i} y_r^f \geq 1 \quad i \in \mathcal{G}$$

$$\sum_{f \in F} \sum_{r \in R_D^f \cap V_D^i} y_r^f \geq 1 \quad i \in \mathcal{G}.$$

### 4.6 The SND Formulation

The resulting formulation, denoted SND, for the express package service network design problem is

(SND)

$$\min \sum_{f \in F} \sum_{r \in R_P^f \cup R_D^f} h_r^f y_r^f + \sum_{s \in S} \sum_{q \in Q^s} \left( \sum_{k \in q} \sum_{p \in P^k} c_p^k b^k \right) w_q^s \quad (6)$$

$$\sum_{s \in S} \sum_{q \in Q^s} \left( \sum_{k \in q} \sum_{p \in P^k} \Gamma_{ij}^q b^k \right) w_q^s \leq \sum_{f \in F} \sum_{r \in R_P^f \cup R_D^f} \alpha_{ij}^r u^f y_r^f \quad \text{for all } (i, j) \in A \quad (7)$$

$$\sum_{r \in R_P^f \cup R_D^f} \beta_i^r y_r^f = 0 \quad \text{for all } i \in N, \quad f \in F \quad (8)$$

$$\sum_{r \in R_P^f} y_r^f \leq n^f \quad f \in F \quad (9)$$

$$\sum_{f \in F} \sum_{r \in R_P^f \cap L_i^t} y_r^f \leq \sum_{m=t}^{T} a_i^m \quad i \in H, \quad t \in \{1, 2, \ldots, T\} \quad (10)$$

$$\sum_{f \in F} \sum_{r \in R_P^f \cap V_P^i} y_r^f \geq 1 \quad i \in \mathcal{G} \quad (11)$$

$$\sum_{f\in F}\sum_{r\in R_D^f\cap V_D^i} y_r^f \geq 1 \quad i \in \mathcal{G} \tag{12}$$

$$\sum_{q\in Q^s} w_q^s = 1 \quad \text{for all } s \in S \tag{13}$$

$$\sum_{s\in S}\sum_{q\in Q^s}\left\{\sum_{k\in K}\sum_{p\in P^k\cap P_i^t}(\delta_{ij}^p b^k)\right\} w_q^s \leq \sum_{m=t}^{T} e_i^m \quad i\in H,\quad t\in\{1, 2, \ldots, T\} \tag{14}$$

$$w_q^s \geq 0 \quad \text{for all } q \in Q^s, \quad s\in S \tag{15}$$

$$y_r^f \geq 0 \text{ and integer} \quad \text{for all } f \in F, \quad r \in R_P^f \cup R_D^f, \tag{16}$$

where $\alpha_{ij}^r$ equals 1 if arc $ij$ is contained in route $r$, and equals 0 otherwise.

The objective function 6 is to find the cost minimizing set of aircraft, vehicle, and package routes. Constraints 7 are forcing constraints that assign packages to a leg only if that leg is part of a selected aircraft/vehicle route and the leg's capacity is not exceeded by the assignment. With respect to the route design variables, constraints 8 are balance constraints, constraints 9 are fleet size constraints, constraints 10 are the hub landing capacity constraints, and constraints 11 and 12 are the connectivity constraints. With respect to the package flow variables, constraints 13 are the demand constraints that ensure that all shipments are serviced, and constraints 14 are the hub sort capacity constraints. Constraints 15 ensure nonnegativity of package flows, and constraints 16 limit the design variables to nonnegative integer values.

## 4.7 Cutset Inequalities

Because the LP relaxations of network design models are notoriously weak, we add aggregate capacity demand inequalities to our model. These inequalities, an application of the cutset inequalities of Magnanti, Mirchandani, and Vachani (1995), require that the total capacity provided by the design variables be larger than or equal to the total demand, for any O–D cutset $\{\mathcal{S}, \mathcal{T}\}$. An O–D cutset $\{\mathcal{S}, \mathcal{T}\}$ is defined by a partitioning of the node set $N$ into two nonempty disjoint sets $\mathcal{S} \subset N$ and $\mathcal{T} = N\backslash\mathcal{S}$. An arc $(i, j)$ belongs to cutset $\{\mathcal{S}, \mathcal{T}\}$ if nodes $i$ and $j$ belong to different sets $\mathcal{S}$ and $\mathcal{T}$. Aggregate demand, $D_{\mathcal{S},\mathcal{T}}$; denotes the total demand of each commodity with its origin and destination in different subsets, i.e., with $O(k) \in \mathcal{S}$ and $D(k) \in \mathcal{T}$ or $O(k) \in \mathcal{T}$ and $D(k) \in \mathcal{S}$. Letting $u_f$ represent the capacity of fleet $f$, $Y_f^{\mathcal{S},\mathcal{T}} = \sum_{r\in R^f}\sum_{(i,j)\in\{\mathcal{S},\mathcal{T}\}\cap r} y_r^f$ and $D_{\mathcal{S},\mathcal{T}}$ denote the total demand from the set $\mathcal{S}$ to the set $\mathcal{T}$, the aggregate capacity demand inequalities are

$$\sum_{f\in F} u_f Y_f^{\mathcal{S},\mathcal{T}} \geq D_{\mathcal{S},\mathcal{T}} \quad \text{for all O–D cutsets } \{\mathcal{S},\mathcal{T}\}. \tag{17}$$

We strengthen Eq. 17 using the Chvatal–Gomory integer rounding procedure (detailed in NEMHAUSER and WOLSEY 1988) to achieve improved LP bounds. For ease of exposition, we let inequality 17 represent both the original aggregate capacity demand inequalities and the lifted inequalities, and we refer to them as cutset inequalities. We refer to the SND model with cutset inequalities 17 as SND-Cut.

TABLE I
*Data Set Size*

| | DS1 | DS2 | DS3 |
|---|---|---|---|
| Locations | 31 | 91 | 141 |
| Hubs | 4 | 7 | 9 |
| O–D (tree) Commodities | 863 (30) | 6,157 (80) | 17,651 (136) |
| Volume | 70,156 | 404,241 | 852,362 |
| Number of Fleet Types | | | |
| Jets | 3 | 5 | 5 |
| Feeder | 2 | 2 | 2 |

## 4.8 Problem Size

The express package delivery company provided three different data sets, denoted DS1, DS2 and DS3, representing their express package delivery operation (Table I). DS3 represents their entire operation, with DS1 and DS2 representing portions of their operation. We use DS1 and DS2 primarily for computational testing, and for gaining insight about the larger DS3 problem.

A time-space network for DS3 is huge and impractical to use. We use instead our new derived schedule network representation in building our route-based model and solution approach. Even using the derived schedule approach, the SND model for DS3 is still very large, containing 151,002 constraints, 1.0 billion package flow variables, and 0.3 million jet and feeder route variables. This size problem (even the LP relaxation) is too large for state-of-the-art LP/IP solvers (such as, CPLEX 1995, MINTO 1994, OSL 1992, etc.). After applying the network reduction methods of node and link consolidation, we are able to reduce the number of forcing constraints of SND from 151,002 to 15,355 while maintaining exactness of the formulation.

## 4.9 SND and SND-Cut Solution

Even using our problem reduction methods, SND-Cut, with its huge number of constraints and its huge number of variables, is impossible to solve directly. Instead, we use both row and column generation. Our solution procedure for the SND-Cut LP relaxation begins with a very small RMP containing only dummy variables and cutset inequalities with $|\mathcal{S}| = 1$ and $|\mathcal{T}| = 1$.

Then, we explicitly price out jet and feeder route design variables and add them to RMP if necessary. By explicit, we mean that we compute the reduced cost of each route for each fleet and add some number of variables with negative reduced cost. Explicit pricing is quite efficient because there are relatively few design variables (about 0.3 million). In this case, implicit pricing, which involves solving a pricing problem to determine the variable with the minimum reduced cost, is very inefficient because the pricing problem cannot be formulated as a computationally easy problem. The difficulty is that dual prices correspond to routes and not arcs.

Next, we implicitly price out package flow variables by solving an underlying shortest path problem which implicitly considers the large number of paths, and add them to RMP if necessary. We repeat this process until no negative reduced cost columns are found. It is important to note that only implicit column generation is practical for the package flow variables because there are about 1.0 billion of them. The hub sort capacity constraints, however, make implicit pricing difficult. Without them, the pricing problem is for each commodity $k$, a shortest path problem over the derived schedule network with modified link costs of $(c_{ij}^k - \pi_{ij})$, for all $(i, j) \in A$, where $\pi$ is the vector of dual variables associated with the forcing constraints. When the hub sort capacity constraints are included, their associated dual variables cannot be assigned to arcs in the derived schedule network because they correspond to routes in the network. The pricing problem, then, is greatly complicated and can no longer be solved using shortest path algorithms. The result is likely to be a tremendous increase in solution time because the pricing problem is solved so many times. Consequently, we ignore the sort capacity constraints and capture their effects by enforcing the landing capacity constraints at hubs. By limiting the aircraft arrival rate at the hubs, we are able to spread out the shipment arrivals and, hence, satisfy sort capacity constraints.

Finally, we generate additional cutset inequalities by explicitly checking whether there are any violated constraints. (An efficient algorithm for solving the separation problem has not been found, Balakrishnan, Magnanti, and Mirchandani, 1994a). Violated constraints found are added to the RMP and the RMP is solved again.

We repeat this entire process until no violated constraints and no negative reduced cost columns are found. We reduce memory requirements for our SND-Cut algorithm by considering only cutsets with $|\mathcal{S}| \leqslant 3$ and $|\mathcal{T}| \leqslant 3$ (i.e., CUTSET_3).

Implementing row and column generation procedures is a nontrivial task. One challenge is in adding constraints that do not change the solution procedure for the pricing problems and adding variables that do not change the solution procedure for generating violated inequalities. In the latter case, this is a non-issue because we generate cuts with explicit testing of each inequality. For the case of generating columns, we again have no problem with the design variables because they are generated explicitly. The flow variables, however, are generated implicitly, and, therefore, we must be able to apply the dual variables corresponding to cuts with non-zero elements in package flow columns to the arcs in the derived schedule network. Otherwise, our shortest path algorithm will not solve the pricing problem. Fortunately, because our cuts involve only design variables, the package flow variables have only zero coefficients in the cutset inequality rows and so, the solution procedure for the package flow pricing problem is unaffected by the addition of cuts.

After obtaining a root node LP optimal solution, we use branch and bound to obtain integer solutions. This is a heuristic solution approach because an optimal solution may include columns and/or cuts that are not generated in solving the root node LP.

We solve SND using the SND-Cut algorithm with all the steps involving constraint generation eliminated.

### 4.10 SND and SND-Cut Computational Results

We are able to solve the LP relaxation of SND within 40 minutes on a SGI Power Challenge 2-processor workstation with 256 MB RAM using CPLEX version 4.0 (1995). The LP bound generated, however, is very loose and ineffective. In fact, the best IP solution obtained after more than two days of CPU time using branch and bound was more than three times the value of the LP solution.

Unfortunately, the SND-Cut model and algorithm does not improve these results. The issue is computer memory: we run out of memory before we can add enough cuts to improve significantly the LP relaxation. We resolve this issue by pursuing a heuristic approach that utilizes a variant of the SND-Cut Model.

## 5. HEURISTIC SOLUTION APPROACH

BECAUSE SND-CUT is too large to solve, we adopt a heuristic solution approach for our express package service network design problem. Our approach is to first select a subset of variables for consideration. Second, using our SND-Cut algorithm, we solve a restricted SND-Cut model, denoted RSND-Cut, including only the selected variables.

Clearly, the manner in which we select the variables to be included in RSND-Cut has a dramatic effect on the quality of the solution generated. Our selection process involves solving an approximate SND-Cut model, one in which capacity in the system is modeled inexactly. Our idea is that the solution to this approximate model will provide an indication of which variables should be included in our selected subset. In fact, legs used in the solution to our approximate SND-Cut model are considered in constructing the set of potential design variables for our restricted SND-Cut model.

We construct our approximate SND-Cut model so that its optimal solution will fall into one of two categories: either the solution will be optimal for the SND-Cut model, or it will be infeasible because of insufficient capacity to service one or more packages.

Even when the result is infeasibility, we believe that much of the approximate SND-Cut solution is likely to be contained in an optimal solution for the SND-Cut model.

## 5.1 Approximate SND-Cut Model

Our approximate SND-Cut model, denoted ASND-Cut, is SND-Cut with the package flow variables and associated constraints eliminated.

$$\text{(ASND-Cut)} \qquad \min \sum_{f \in F} \sum_{r \in R_P^f \cup R_D^f} h_r^f y_r^f \tag{18}$$

$$\sum_{f \in F} u_f Y_f^{\mathcal{S},\mathcal{T}} \geq D_{\mathcal{S},\mathcal{T}} \quad \text{for all O–D cutsets } \{\mathcal{S},\ \mathcal{T}\} \tag{19}$$

$$\sum_{r \in R_P^f \cup R_D^f} \beta_i^r y_r^f = 0 \quad \text{for all } i \in N, \quad f \in F \tag{20}$$

$$\sum_{r \in R_P^f} y_r^f \leq n^f \quad f \in F \tag{21}$$

$$\sum_{f \in F} \sum_{r \in R_P^f \cap L_i^t} y_r^f \leq \sum_{m=t}^{T} a_i^m \quad i \in H, \quad t \in \{1, 2, \ldots, T\} \tag{22}$$

$$\sum_{f \in F} \sum_{r \in R_P^f \cap V_P^i} y_r^f \geq 1 \quad i \in \mathcal{G} \tag{23}$$

$$\sum_{f \in F} \sum_{r \in R_D^f \cap V_D^i} y_r^f \geq 1 \quad i \in \mathcal{G} \tag{24}$$

$$y_r^f \geq 0 \text{ and integer} \quad \text{for all } f \in F, \quad r \in R_P^f \cup R_D^f \tag{25}$$

With only fleet design variables included, the objective function 18 is to find the cost-minimizing set of routes satisfying constraints 19–25, as defined for SND-Cut.

## 5.2 ASND-Cut Solution

The algorithm to solve ASND-Cut is summarized as:

Step 0. Solve ASND-Cut LP using column and row generation, restricting to $|\mathcal{S}| \leq 3$ and $|\mathcal{T}| \leq 3$ to avoid memory problems.

Step 1. Achieve a feasible IP solution to ASND-Cut, given the current set of columns and constraints, using branch and bound.

Step 2. Explicitly generate violated inequalities with $|\mathcal{S}| \leq 3$ and $|\mathcal{T}| \leq 3$, given the current IP solution. If there are no violated inequalities found, go to Step 4; otherwise add the violated inequalities to the model.

Step 3. Solve ASND-Cut LP, given the current set of constraints, using column generation and go to Step 1.

Step 4. Generate all route variables that contain at least one link used in the ASND-Cut IP solution.

Step 4 of the ASND-Cut IP solution procedure generates the only variables considered in the RSND-Cut model.

## 5.3 Computational Experiences

We implemented our solution approach on a SGI Power Challenge workstation, using CPLEX 4.0 (1995) with two processors and 256 MBs RAM.

To evaluate the performance of our approach, we compare our solution cost for the three test problems, DS1, DS2, and DS3, with the corresponding costs of the solutions generated primarily by hand by planners of the express package delivery company. For the full-scale operation, our results show that our approach has planned costs that are nearly 10% less than those of the planners. This translates to potential cost savings in the tens of millions of dollars annually. In addition, compared to the planners' solution, our solution reduces the number of aircraft miles flown by about 12% and uses about 10% fewer aircraft. The reductions are achieved by increasing the number of routes with two stops, taking advantage of excess aircraft capacity and slack time in schedules. Additional gains are achieved by building pickup and delivery routes that are not mirror images and by ferrying (repositioning empty) aircraft to improve aircraft utilization.

In our calculations, we do not include the savings from reducing the number of aircraft needed because we assume that the aircraft used are owned by the company. In fact, reducing the fleet size by 10% could result in tremendous additional savings, potentially measuring in the billions of dollars.

The run times for DS1, DS2, and DS3 are about 20 seconds, 4 hours, and 11 hours, respectively. Because these are planning tools, run times of even 11 hours are acceptable, particularly because they could result in significant savings.

## 6. SCENARIO ANALYSIS

TO GAIN INSIGHT regarding the effects of various problem parameters on the quality of our heuristic solution and on run time, we generate and solve several representative, yet small, test problems that mirror the structure of our express package service network design problem. In these test problems, we vary the number of packages, the number of fleets, and the number of hubs, and evaluate the effects of these variations.

Our test case network is a subset of DS1 and contains 24 locations, two fleet types with the capacity of the second fleet twice that of the first, three hubs, and 552 commodities (each specified by an origin–destination pair) totaling 70,156 packages. This problem instance is small enough to test all of our solution algorithms and yet, complex enough to include all components of our multimodal express package delivery application.

### 6.1 Three Hubs, Two Fleet Types, Varying Demand

Using our test case with three hubs and two fleet types, we evaluate the effects of different levels of demand, namely 50, 100, 150, 200, 250, and 300% of the original demand level in our test problem. We observe that overall run times increase with demand, from about 15 seconds for our approach with demand volumes at 50% of the original to about 476 seconds for demand volumes at three times the original. This increase in run time is caused in large part by the increased difficulty of assigning flows to routes, requiring additional iterations of the column generation procedure.

Further, we observe that, for lower levels of demand, that is one-half of the original demand and even the original demand, the cutset inequalities are inconsequential (the gap between the best integer solution found and the lower bound produced by the LP relaxation of SND without cuts is between 0 and 2%). As demand increases, however, the cuts are essential. To illustrate, when the original demand is tripled, the gap between the solution to SND with and without cuts is over 21%.

We observe from these experiments that, for all levels of demand, our ASND-Cut LP is a tight relaxation of the SND-Cut LP model. That is, their solutions were equal for demands of 50, 100, and 150%; and for demands of 200, 250, and 300%, the gap between their solutions never exceeded 0.25%.

Our approach is particularly fast to execute compared to the LP solution time for SND-Cut. In the case when the original demand level is halved, our approach takes 15 seconds to generate a feasible solution, and the time to solve the SND LP relaxation is 60 seconds without cuts and 987 seconds with cuts. This difference is exacerbated for high demand levels. When the original demand is tripled, the run time for our approach is 476 seconds, compared to 206 seconds for the LP relaxation of SND without cuts and 10,347 seconds with cuts. (The gap between the solution without cuts and the solution with is 21%, so it is important to include cuts.)

A drawback to our approach (just as in solving SND or SND-Cut) is that, as demand levels become very high and the network is very congested, the quality of the solution is hard to evaluate. For example, when the original demand is increased threefold, our solution has an optimality gap between our best solution and the SND-Cut LP of about 30%. Although our ASND LP solution is about the same as the SND-Cut LP solution, we lose about 13% of the solution quality in getting an IP solution to ASND-Cut. Further, we lose about 11% in moving from our RSND LP solution to a feasible IP solution. The rest is lost in moving from the ASND solution to a feasible LP solution to RESPND. The issue is that good LP solutions do not necessarily yield good IP solutions. At this point, it is hard to know if improved IP solutions exist and, if they do, if they exist in our restricted set of feasible solutions. Further research and experiments are necessary to answer these questions.

### 6.2 One Hub, Varying Numbers of Fleet Types, Varying Demand

We consider the test case with a single hub and a single fleet type, again with varying levels of demand. These problems are much easier to solve, with run times from 4 to 192 seconds and gaps between the best solutions found and the lower bounds of 0.5–11%. They are easier to solve, in part, because they are much smaller, with far fewer routing and fleeting choices.

Next, we evaluate the case of a single hub and two fleet types. Compared to the single hub and single fleet scenario, the solution time for SND-Cut LP is reduced surprisingly. For example, the solution time for demand levels increased by 300% is reduced from 10,347 seconds to only 1191 seconds. Also, the objective function reduces from 300,667 to 294,055. It seems that the greater flexibility in assigning route capacity helps the column generation procedure to converge more quickly compared to the single fleet scenario.

Interestingly, for demands exceeding the original level, the optimality gap is increased compared to the single hub and single fleet scenario, from 7.96 to 15.96% for the case when the original demand is tripled. This likely results because the LP relaxation takes combinations of the two fleet types to make an idealized aircraft with minimum cost and exactly the capacity needed.

### 6.3 Three Hubs, One Fleet Type, Varying Demand

Finally, we consider the case of a single fleet and three hubs with varying levels of demand. The run time of SND-Cut is increased greatly and the quality of the lower bound is degraded compared to the single hub–single fleet case, illustrating the large computational impact of moving from a single hub to multiple hubs. To solve the most congested network (that is, three times the original demand) it takes 2482 seconds, compared to 1008 seconds when only one hub is considered. Further the gap between the best solution found and the lower bound is 15.65%, compared to 7.96% for the single hub case. These run time increases result because an increased number of hubs yields a larger problem size with an increased number of shipment routings. The result is more iterations of column generation and longer run times. The degradation in LP solution quality as the number of hubs increases is related to the connectivity constraints and cutset inequalities. With only one hub, connectivity constraints for that hub ensure exact coverage of each leg. For additional hubs, cutset inequalities apply but connectivity constraints do not. Because we do not include all cutset inequalities, we cannot guarantee that each leg is covered once. Hence, more fractional route variables occur in the SND LP solution for multiple hubs than for a single hub.

### 6.4 Summary

The results of our scenario analyses are summarized as:

- Run time and quality of solution deteriorates as demand increases. Excluding cutset inequalities has an increasingly negative impact as demand increases.
- Near optimal solutions are found when the demand level is low, largely because connectivity constraints are the only tight constraints.
- An increase in the number of fleets results in improved solution quality and surprisingly comparable run times, but larger optimality gaps.
- The minimum number of aircraft used occurs for the single hub networks, because maximum consolidation of demand is possible. In the multiple hub networks, although the number of aircraft used increases, the variable operating costs decrease and service reliability increases (reduced total distance traveled and more capacity in the system).

Through these experiments and our original application, we assess the capabilities and limitations of current state-of-the-art integer programming approaches for large-scale service design problems. We show that, although these approaches are powerful, there is still work to be done before the problems encountered in practice can be solved routinely. The major drawbacks to current IP approaches include run time and memory requirements to achieve tight lower bounds and quality feasible solutions.

## ACKNOWLEDGMENT

This research has been supported by a grant to the Massachusetts Institute of Technology from the United Parcel Service.

## REFERENCES

R. K. AHUJA, T. L. MAGNANTI, AND J. B. ORLIN, *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall, Englewood Cliffs, N.J., 1993.

A. A. ASSAD, "Multicommodity Network Flows—A Survey," *Networks* **8,** 37–91 (1978).

A. BALARISHNAN, T. L. MAGNANTI AND R. WONG, "A Dual-Ascent Procedure for Large-Scale Uncapacitated Network Design," *Opns. Res.* **37,** 716–740 (1989).

A. BALAKRISHNAN, T. L. MAGNANTI AND P. MIRCHANDANI, "A Dual-based Algorithm for Multilevel Network Design," *Management Sci.* **40,** 567–581 (1994a).

A. BALAKRISHNAN, T. L. MAGNANTI AND P. MIRCHANDANI, "Modeling and Heuristic Worst-Case Performance Analysis of the Two-Level Network Design Problem," *Management Sci.* **40,** 846–867 (1994b).

A. BALAKRISHNAN, T. L. MANGANTI, AND R. T. WONG, "A Decomposition Algorithm for Local Access Telecommunications Network Expansion Planning," *Opns. Res.* **43,** 58–76 (1995).

C. BARNHART AND R. R. SCHNEUR, "Air Network Design for Express Shipment Service," *Opns. Res.* **44,** 852–863 (1996).

D. BIENSTOCK AND O. GUNLUK, Capacitated Network Design—Polyhedral Structure and Computation," Working paper, Columbia University, New York, 1994.

J. W. BILLHEIMER AND P. GRAY, "Network Design with

Fixed and Variable Cost Elements," *Transp. Sci.* **7,** 49–74 (1973).

S. CHANG AND B. GAVISH, "Lower Bounding Procedures for Multiperiod Telecommunications Network Expansion Problems," *Opns. Res.* **43,** 43–57 (1995).

L. W. CLARKE AND P. GONG, Capacitated Network Design with Column Generation, Working Paper LEC-96-06, The Logistics Institute, School of Industrial and System Engineering, Georgia Institute of Technology, Atlanta, GA, 1995.

CPLEX OPTIMIZATION, INC., "Using the CPLEX Callable Library, version 4.0," CPLEX Optimization, Inc., Incline Village, NV, 1995.

T. G. CRAINIC AND J. M. ROUSSEAU, "Multicommodity, Multimode Freight Transportation: A General Modeling and Algorithmic Framework for the Service Network Design Problem," *Transp. Res.* **20B,** 225–242 (1986).

J. M. FARVOLDEN AND W. B. POWELL, "Subgradient Methods for the Service Network Design Problem," *Transp. Sci.* **28,** 256–272 (1994).

B. GENDRON AND T. G. CRAINIC, Relaxations for Multicommodity Capacitated Network Design Problems, Working paper, Centre de recherche sur les transports, Publication #965, Université de Montréal, Montréal, Canada, 1994.

C. A. HANE, C. BARNHART, E. L. JOHNSON, ET AL., "The Fleet Assignment Problem: Solving a Large-Scale Integer Program," *Math. Prog.* **70,** 211–232 (1995).

INTERNATIONAL BUSINESS MACHINE CORP., "IBM Optimization Subroutine Library (OSL) Guide and Reference," *IBM Systems Journal,* SC23-0519, Vol. 31, 1992.

K. L. JONES, I. J. LUSTIG, J. M. FARVOLDEN, AND W. B. POWELL, "Multicommodity Network Flows: The Impact of Formulation on Decomposition," *Math. Prog.* **62,** 95–117 (1993).

J. L. KENNINGTON, "A Survey of Linear Cost Multicommodity Network Flows," *Opns. Res.* **26,** 209–236 (1978).

D. KIM, Large Scale Transportation Service Network Design: Models, Algorithms and Applications, Ph.D. thesis, Center for Transportation Studies, Massachusetts Institute of Technology, Cambridge, MA, 1997.

M. J. KUBY AND R. G. GRAY, "The Hub Network Design Problem with Stopovers and Feeders: The Case of Federal Express," *Trans. Res. A,* **27A,** 1–12 (1993).

B. W. LAMAR, Y. SHEFFI AND W. B. POWELL, "A Capacity Improvement Lower Bound for Fixed Charge Network Design Problem," *Opns. Res.* **38,** 704–710 (1990).

J. M. Y. LEUNG, T. L. MAGNANTI, AND V. SINGHAL, "Routing Point-to-Point Delivery Systems: Formulation and Heuristics," *Transp. Sci.* **24,** 245–260 (1990).

T. L. MAGNANTI AND R. T. WONG, "Network Design and Transportation Planning: Models and Algorithms," *Transp. Sci.* **18,** 1–55 (1984).

T. L. MAGNANTI, P. MIREAULT, AND R. WONG, "Tailoring Benders Decomposition for Uncapacitated Network Design Problem," *Math. Prog.* **26,** 112–154 (1986).

T. L. MAGNANTI, P. MIRCHANDANI, AND R. VACHANI, "Modeling and Solving the Two-Facility Capacitated Network Loading Problem," *Opns. Res.* **43,** 142–157 (1995).

D. MEDHI AND D. TIPPER, On Some Approaches to Solving a Multi-Hour Communications Network Design Problem, Working paper, Computer Science Telecommunications, University of Missouri–Kansas City, Kansas City, KS, 1995.

M. MINOUX, "Network Synthesis and Optimum Network Design Problems: Models, Solution Methods and Applications," *Networks* **19,** 313–360 (1989).

G. L. NEMHAUSER, M. W. P. SALVELSBERGH, AND G. C. SIGISMONDI, "MINTO, a Mixed INTeger Optimizer," *Opns. Res. Lett.* **15,** 47–58 (1994).

G. L. NEMHAUSER AND L. A. WOLSEY, *Integer and Combinatorial Optimization*, John Wiley and Sons, Inc., Chichester, U.K., 1988.

H. N. NEWTON, "Network Design Under Budget Constraints with Application to the Railroad Blocking Problem, Ph.D. thesis, Industrial and Systems Engineering, Auburn University, Auburn, AL, 1996.

W. B. POWELL, "A Local Improvement Heuristic for the Design of Less-than-Truckload Motor Carrier Networks," *Transp. Sci.* **20,** 246–257 (1986).

W. B. POWELL AND Y. SHEFFI, "Design and Implementation of an Interactive Optimization System for Network Design in the Motor Carrier Industry," *Opns. Res.* **37,** 12–29 (1989).

(Received, July 1997; revisions received, July 1998; accepted, August 1998)