CHAPTER 4

COMBINED ROUTING AND SCHEDULING

4.1. INTRODUCTION

Most combined routing and scheduling problems occur as applications and are characterized by task precedence and time window constraints. Task precedence relationships force the pickup activity for a task to precede the delivery activity for the task and the pickup and delivery tasks must be on the same vehicle. The difficulties due to precedence constraints are illustrated in the following example.

In Fig. 4.1, the optimal tour for the traveling salesman problem is *D*–1–2–3–4–*D* with a length of 12 hours. If the precedence relationship forces node 1 to be serviced before node 3 and node 4 to be serviced before node 2 (e.g. deliveries are required from node 1 to node 3 and from node 4 to node 2), then the optimal tour of length 13 hours is *D*–1–4–3–2–*D*. If a driver cannot work more than 12 hours a day, then two tours *D*–1–3–*D* and *D*–4–2–*D* each 9 hours in duration result and, hence, two vehicles are needed.

A second set of constraints involves the servicing of tasks within specified time windows. A time window on a service task requires that the task be serviced within the specified time interval. For example, in a delivery problem, a particular delivery may be constrained to occur between 10:00–11:30 AM. Thus, any route which involves this particular task must ensure that the delivery time fall within these time bounds. In this discussion, the time interval is a contiguous period of time and this type of window is called a *simple time window*.

The example in Fig. 4.2 shows how time windows may complicate a routing problem. Node *D*, the depot, provides service to the three points 1, 2, and 3 each requiring 1/3 of a truck-load. The vehicle cannot leave the depot until 8:00 AM and must return by 5:00 PM. The travel times (in hours) between nodes are shown in Fig. 4.2. An optimal tour of length 8 is *D*–1–2–3–*D*. If nodes 1 and 3 must be serviced between 8–12 AM and node 2 after 12 AM, then the above tour is no longer feasible and the optimal solution is to have two tours *D*–1–2–*D* and *D*–3–*D* with lengths equal to 7 and 4 hours, respectively. Hence, in this case two vehicles are required.

With no time windows, the set of tasks that may follow a particular task can be specified *a priori* and one can construct a network (directed or undirected depending upon the application) including all the tasks. With time windows, the complete set of tasks that can feasibly follow a given task cannot, in general, be specified beforehand since the exact time of service for a given task cannot be ascertained in advance. It is possible for a task to follow a given task, say task *t*, on a route if task *t* is serviced at the beginning of its time window but not in the case where task *t* is serviced at the end of the time window. Hence, with time windows, it becomes very difficult to construct the complete network of possible connections *a priori*. The works of Swersey and Ballard[639], Orloff[526], Wren and Holliday[695], Christofides *et al.*[144], and Baker[21] comprise the only literature on the routing problem with simple time windows that we are aware of.

A more complicated version of a time interval is when the time interval involves a noncontiguous period of time. Under such a requirement, a task may require service a specified number of times over a time duration of say, a week, and there may be constraints on the pattern of days for servicing these tasks. For example, there may be a separation of at least two days required between consecutive services of the task or it may be that a task can only be serviced on Monday, Tuesday, or Friday. Once the days of the week are assigned, there may be additional simple time window and task precedence constraints on the times for servicing the tasks during the days scheduled. The assignment-routing problem of Russell and Igo[573] is one example of a problem with these kinds of constraints. Having assigned the tasks to days of the week, the problem for a given day becomes a daily routing problem which may be solved by one of the routing procedures presented in Chapter 2. However, the complicated interaction of the routing and assignment decisions is a major source of difficulty in this problem and can

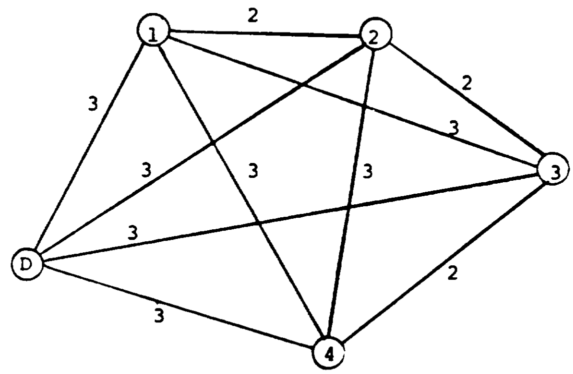


Fig. 4.1. A routing problem with precedence relations.

seriously effect the solution to the problem. To our knowledge, there has been almost no work done on solving the combined assignment-routing problem when the time windows involve noncontiguous periods of time and there exist precedence constraints on the servicing of the tasks on the days assigned.

The pickup and delivery problems considered in this chapter fall into the class of routing and scheduling problems known as "many to many". The "many to many" means that each item to be serviced can have a different origin (or pickup) location x_i and a different destination (or delivery) location y_i, although these origins and destinations need not be unique. In general, however, these problems can have "many" origins and "many" destinations. In the single depot vehicle routing problems considered in Chapter 2, all items at each pickup point are being delivered to a central depot so that all the destinations for all the items are the same. We call this problem, the "many to one" routing problem since these problems have many origins and one destination (the depot). Similarly, the "one to many" problem has one origin and many destinations. In this problem, the items are loaded on the vehicles at a central depot and delivered to the "many" customers. The "many to few" and "few to many" routing problems correspond to the multiple depot routing problems considered in Chapter 2. We should remark that in "many to many" problems, "many" refers to the pickup and delivery locations of the tasks being serviced and the trips to and from the depots or garages are additional. In "many to one", "one to many", "many to few" and "few to many" problems, the "many" refers to the servicing of the required tasks (either pickup or delivery) and the "one" (or "few") refers to the beginning or ending locations of the routes at the depot (garage) or depots (garages).

In numerous combined routing and scheduling problems, we talk about the pickup and delivery locations of the items being serviced and do not explicitly worry about the garages where the vehicles are housed. The deadhead times to go from the depots or the garages to the first stop on the routes and from the last stop on the routes to the garages or depots are generally added to the length of the routes after the routes are formed, and they are not considered part of the optimization. If the distance to and from the garage (or garages) were to

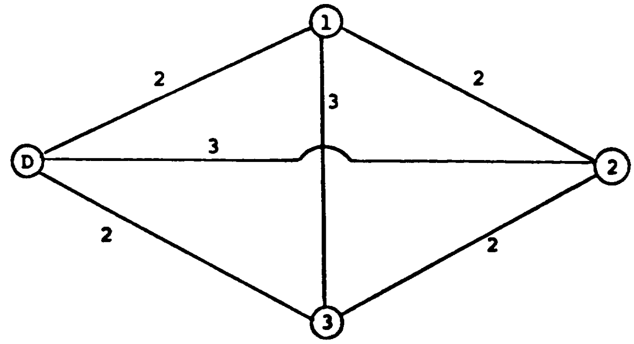


Fig. 4.2. A routing problem with time windows.

be taken into account in the optimization, then the routes might be altered somewhat. This consideration may complicate the models and affect the solution. In many cases, we have found that the total deadheading time to and from the garages is marginally small when compared to the total deadheading or travel times on the routes themselves.

In this chapter, the following routing and scheduling problems are discussed:

(1) school bus routing and scheduling
 *no mixed loads
 *mixed loads
(2) tractor-trailer routing and scheduling with full loads
(3) tractor-trailer routing and scheduling with partial loads
(4) routing and scheduling of street sweepers and household refuse collection vehicles
 *no constraints
 *parking regulation constraints
(5) airplane scheduling
(6) dial-a-ride routing and scheduling
 *without delivery time constraints
 *with delivery time constraints.

In the next section, a description of each of the above problems is presented. Most of these problems involve precedence relationships or time windows, or both. In each application, the situation and the constraints are described. In the remaining sections of this chapter, algorithms for solving these problems are presented. Practical considerations regarding implementation are brought out in this chapter and are developed further in Chapter 5. All of these problems are *NP*-hard. In most cases, the complications in these problems are such that exact algorithmic approaches based on mathematical programming formulations have not proven successful.

4.2 DESCRIPTION OF VARIOUS ROUTING AND SCHEDULING PROBLEMS

In this section, a description of a variety of routing and scheduling problems is given. In Sections 4.3–4.10, algorithms for solving these problems are presented.

4.2.1 *School bus routing and scheduling*

In the school bus routing and scheduling problem, there are a number of schools. Each school has a set of bus stops with a given number of students assigned to each stop and a fixed starting and ending time with corresponding time windows for the pickup and delivery of the students. The time window before the starting time of the school involves the allowable time interval for the delivery of students to the school in the morning and the time window after the ending time of the school is the allowable time interval associated with the pick-up of the students at the end of the school day. The primary objective of this problem is to minimize the "transportation costs" for the district. For a district with a leased fleet of vehicles, a convenient surrogate objective to transportation costs is to minimize total number of vehicles required. For a district-owned fleet of vehicles, a convenient surrogate objective is to minimize a combination of the fleet's operating cost and the number of vehicles used.

The term "transportation costs" merits some discussion. In many states, some of the student transportation costs are borne by the school district while other student transportation costs are subsidized by the state (up to 90% in New York State). However, the aid formula under which the subsidy is disbursed may involve factors other than the number of students being transported. As such, these aid formulas can further complicate the problem of minimizing the total student transportation costs for a district. For example, the formula used by New York State only subsidizes a district for those riders who live more than 1.5 miles from the school and this formula takes into account the length of the route as well as the number of students who live over 1.5 miles from the school. Since the local school districts set up the bus routes for their transportation programs, it may be more economical for a local school district to utilize more buses than absolutely necessary and have less efficient routes. As a further complication, the transportation cost for a school district is not necessarily a multiple of the number of buses used by the district. For example, in 1974, the Brentwood School District on Long Island, New York, paid about $11,000 per year for a bus and driver leased for 6 hours a day, $14,000 per year for a bus and driver leased for 11 hours a day, and $15,000 per year for a bus and driver leased for 12 hours a day. This cost has almost doubled since then.

For many districts, the starting and ending times of the schools fall into relatively distinct time periods. For example, all high schools might start about 7:30, middle schools about 8:15 and elementary schools about 9:00. In this type of situation, the problem of minimizing the capital cost for the district becomes one of minimizing the maximum number of buses needed in any time period. If the starting and ending times of the schools do not fall into distinct time periods, then the problem of minimizing capital costs becomes the problem of minimizing the maximum number of buses needed at any point in time.

The first problem one must solve in scheduling school buses is to select the proper starting and ending times of the schools. If the school starting and ending times can be changed so as to substantially reduce the number of students requiring transportation in the peak time period, then a reduction in the number of buses needed for peak time operation can be realized. As an example, if time period 1 requires 20 buses and time period 2 calls for 40 buses, then a minimum of 40 buses will be used in the district. If, however, the starting and ending times of the schools are changed so that 32 buses are needed in time period 1 and 28 in time period 2, then the minimum number of buses required is only 32—a savings of 8 buses. Of course, the scheduling component may find a set of schedules for the buses requiring more than 32 buses. However, if capital cost minimization is a prime objective, it is better to begin the routing and scheduling of the school buses with a requirement of 32 buses rather than a requirement of 40 buses. If the starting and ending times of the schools do not fall into distinct time periods, then this problem becomes the problem of determining the appropriate starting and ending times in order to minimize the number of vehicles on the street at any point in time. This problem has been analyzed in detail by Desrosiers[180] and is discussed in Section 4.3.1.

Having determined the starting and ending times for the schools, the routing and scheduling algorithms must go to great lengths to ensure that no bus is idle in a peak time period; that is to say, every bus utilized in a time period immediately preceding or immediately following the peak time period must also service a route during the peak time period. Consequently, the routes for the buses in the off-peak time periods need not be as efficient as the routes in the peak time periods. As long as each bus utilized in an off-peak time period can be assigned a route during the peak time period, the capital requirements for the system are unchanged. Therefore, routes in the off-peak period can be shortened in duration to pick up fewer students than would ordinarily be prescribed by an "optimal seeking" routing procedure such as those described in Chapter 2. The scheduling component must assure that the routes in the off-peak fit together with the routes in the peak time periods to form bus schedules which minimize capital requirements.

In a suburban area, a bus is generally filled to capacity before the time to service the route is exhausted so that the routing problem for each school becomes a clustering problem which is relatively easy to solve. The clusters are the bus stops that form a route and it is required that the sum of all students to be picked up at the stops in the cluster does not exceed the bus capacity. In a suburban area, a bus can generally service many routes in a day (6–14 for the Brentwood, New York school district). Under these conditions, therefore, the scheduling problem is extremely important with regard to the effective utilization of the buses. In a rural area, on the other hand, a bus typically handles very few routes in a day. Each of these routes tends to be long and bus capacity may not become saturated on any of these routes. Hence, in scheduling school buses in a rural area, the routing problem becomes very important.

Most papers on the routing and scheduling of school buses focus primarily on the routing component (Bennett and Gazis[66], Angel *et al.*[8], and Tracz and Norman[653]). Newton and Thomas[508], and Bodin and Berman[82] discuss procedures for forming daily bus schedules with distinct time periods for the starting and ending time of the schools, as well as methods for routing the buses. In Swersey and Ballard[639], two integer programming formulations and a solution procedure based on linear programming are described for solving the bus scheduling problem when the starting and ending times of the schools do not fall into distinct time periods. In [639], the routes for each of the schools and the starting and ending times of the schools are assumed given.

Most papers in this area are concerned with the *single load problem*, which is the problem in which the routes either pick up or deliver students to a single school. In rural areas, the *mixed loading problem* can occur. In the mixed loading problem, students from more than one school are picked up by a single bus, some students are delivered to one school while others remain on the bus and are transported to the next school along with additional students who are picked up in between the two schools. This problem resembles problems described in other sections of this chapter (see Sections 4.5 and 4.9). Much of the work on the mixed load routing and scheduling problem that we are aware of has been carried out by Orloff, Lockfeld, and members of the Center for Urban Analysis of Santa Clara County in California[116]. Our discussion in Section 4.3 will focus on the single load problem.

4.2.2 *Tractor-trailer routing and scheduling with full loads*

A common commercial distribution problem entails the routing and scheduling of the tractor portion of tractor trailer trucks where the trailers are assumed to have a full load. The term *full load* means that a trailer is attached to the tractor and is transported directly from a pickup point (the origin) to a delivery point (the destination). The load on a trailer has a unique and specified destination and is not to be split among different destination locations. The capacity of a tractor is one trailer. Since each trailer is transported from its origin to its destination, this problem obviously involves precedence constraints. A typical route for a tractor may be as follows (see Fig. 4.3)

> "leave domicile, pick up a trailer, deliver a trailer, deadhead, pick up a trailer, deliver a trailer, pick up a trailer, deliver a trailer, . . . , deliver a trailer, return to domicile."

The demands are specified in terms of the number of trailer trips between origin/destination pairs. Given this demand data, one may address the following two decision problems:

Problem 1. Minimize the total distribution cost for handling *all* origin–destination demands.

Problem 2. Determine the optimal fleet size required to service a subset of the origin–destination demands given that the remaining demand is to be serviced by common carrier.

Thus, Problem 2 involves an additional decision as to which subset of required demands is to be serviced. This subset is chosen using a profit-maximizing criterion with the unprofitable demand requirements being turned over to a common carrier. The paper by Ball *et al.*[38] addresses this problem in the context of the distribution activities of a large chemical firm. It is

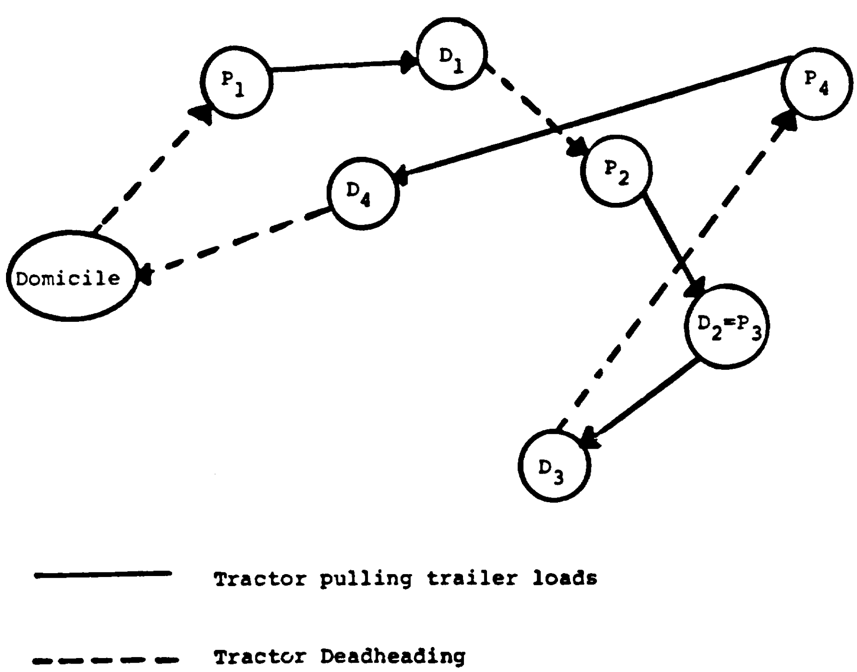


Fig. 4.3. Example of route for tractor trailer with full load problem.

also possible for a common carrier to have this problem in the sense that a common carrier may turn over undesirable or unprofitable trailer trips to a second common carrier (see Goren[312]).

In the application studied by Ball *et al.*[38], severely restrictive window constraints were not present. Nevertheless, the following timing constraints had to be observed: (a) Certain pickups and deliveries had to be carried out during specific hours of the day; (b) Each route served by two-driver crews could not exceed 120 hours in total driving time and each route served by one driver could not exceed 60 hours in total driving time; (c) multiple trips between the same origin-destination pair had to be spread out over the week. For example, if there were 7 trips between points A and B over the course of a week, constraint (c) requires that not all of these be scheduled for the same day, but rather that they be approximately evenly distributed over the days of the week (for example, two trips on Monday and on Thursday and one trip on Tuesday, Wednesday, and Friday). In Section 4.4, three algorithms for solving this problem are given.

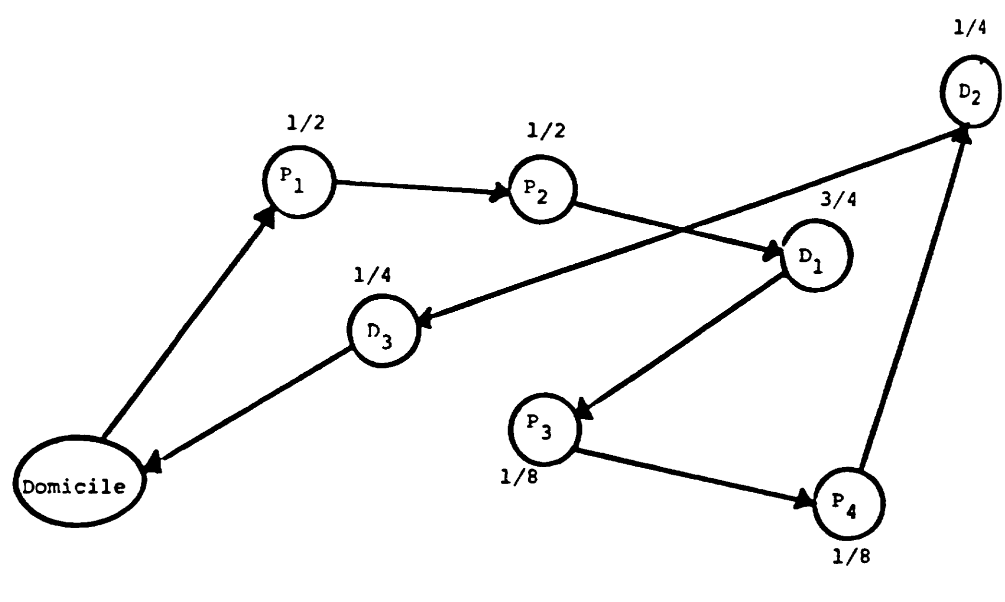
4.2.3 *Tractor trailer routing and scheduling with partial loads*

This problem is similar to the full load problem of Section 4.2.2 except that the demand for each origin–destination pair need not be a full trailer load. Consequently, the load on a trailer may be split among different origins and destinations. In this case, a typical route may look like the following:

> "leave domicile, pick up part of a load, deadhead, pick up part of a load, deadhead, deliver part of a load, pick up part of a load, deadhead, . . . , deliver part of a load, return to domicile."

Such a route is illustrated in Fig. 4.4.

The constraints on this problem are the precedence relationships for each origin-destination pair, time windows (most pickup and delivery locations have specific hours of operation), and crew workrules such as maximum number of hours/week that a crew can drive and break-time restrictions. Although many of the crew workrule constraints were considered explicitly in the algorithm that we devised, the firm assumed responsibility for forming the final set of crew



Note: The numbers outside of the nodes indicate the % of trailer either picked up or delivered at the node. Thus, at P_2 we pick up 1/2 a trailer load.

Fig. 4.4. Example of route for the tractor trailer with partial route problem.

schedules associated with the generated vehicle routes. The crew workrules incorporated by the algorithm into vehicle schedules were meant to facilitate this task for the firm.

In Assad *et al.*[18], this tractor trailer routing problem is discussed in greater detail. Problems of a similar structure can occur for the routing and scheduling of messenger services, package delivery within a local area, deliveries between banks, etc. This problem is very general and is deserving of further analysis.

It is interesting to draw an analogy between this problem and the school bus routing and scheduling problem with mixed loads (see Section 4.2.1). For the school delivery problem (that is, students being delivered to the schools), the number of destination locations (number of schools in the district) is very small compared with the number of origin locations (number of bus stops where students are to be picked up). The time window on the pickup time for a student is an interval [start time of the school—T, start time of the school] where T is a constant like 15 minutes. The loads at a bus stop are equivalent to the partial loads at the pickup or delivery points. The capacity of the vehicle is equal to the number of students that can be carried on it, and there are virtually no driver workrule constraints since school bus scheduling has virtually no mid-day required service. To our knowledge, nobody has utilized an algorithm such as described in [18] for solving the school bus routing and scheduling problem with mixed loads.

4.2.4 *Street sweeper and household refuse collection routing and scheduling*

The problems of scheduling street sweepers and household refuse collection vehicles are applications of the rural postman problem (see Chapter 2). For both of these problems, a set of street segments is specified as needing service. The basic problem is to cover every street segment by a vehicle in such a way that the minimum number of vehicles is used. A highly correlated objective is to minimize the total deadhead time of the vehicles for either a fixed or variable fleet size. There are no precedence relationships on the entities to be serviced. The time windows for the street sweeper routing and scheduling problem correspond to the parking regulations on the streets. In most cases, there are no time windows on the demands for the household refuse collection problem. If the network of streets to be serviced in a time period is connected, then these problems reduce to the Chinese postman problem of Chapter 2 which is very simple to solve.

The basic approaches for solving these problems are variants of the "route first-cluster second" or "cluster first-route second" procedures described in Chapter 2 as applied to the Chinese postman problem. The "route first" approach for routing and scheduling problems will also be encountered for the tractor trailer routing and scheduling problem with full loads (see Section 4.4). The algorithm for street sweepers will utilize the procedure for solving the directed rural postman problem whereas the algorithm for the tractor trailer routing and scheduling problem with full loads will use the algorithm for the undirected rural postman problem.

The key confounding constraint for street sweeper routing and scheduling is the constraint that sweeping can only take place when parking is banned from the street since only during this time can a sweeper come close enough to the curb to be effective. In New York City, for example, there are about seventy different parking regulations ranging from "no parking during peak traffic hours" to "parking on alternate sides of the street". Also, many New York City street segments† have more than one parking code. For example, a street segment can have the following two parking regulations: no parking 8–9 AM on part of the segment and no parking any time on the remainder of the segment. The "no parking 8–9" is to restrict parking during the AM rush hour while "no parking any time" is to restrict parking in front of some special building (for example, a school or church). The only time that the entire segment can be swept is 8–9 AM since we can only sweep the entire segment when parking is banned from the street. Therefore, in the network representation for this region, we set the parking regulation for this street segment to "no parking 8:00–9:00 AM".

A basic problem in household refuse collection is the statistical estimation of demand for

†A street segment is the street between two intersections. A street segment will become a branch in the corresponding network constructed for this problem.

service on a street. Based on the characteristics of the street (type of housing, number of houses, etc.) the only estimation procedure that we know of that works is based on a count of the number of houses on the street (Schuster and Schur[587]). As long as the neighborhood is homogeneous (like single family housing), this procedure is fine. However, in locations with many different types of buildings (such as Manhattan), the procedure breaks down. To our knowledge, no successful procedure for estimating demand other than the one cited above has been developed. The discussion in Section 4.6 concentrates on the street sweeper routing and scheduling problem. The household refuse collection routing and scheduling problem is quite simple once the demand on each street segment is known.

4.2.5 *Airplane scheduling*†

Airlines coordinate the scheduling of their airplanes with the process of generating their timetables. The generation of a timetable takes into account such factors as the expected number of passengers traveling between cities, frequency of service desired, nonstop versus multiple stop service, etc. The scheduling of the airplanes also gives an estimate of the capital and operating costs attributable to that timetable. This process takes into account the problems of generating pairings and bid lines for the crews (see Chapter 3) in that the airlines might change their timetables and vehicle schedules if a savings in crew costs can be realized. However, the operating costs attributable to running and maintaining the airplanes are significantly greater than the crew operating costs. In spite of this, most scheduling of airplanes for commercial airlines is carried out on a manual basis or in an interactive computing mode with little algorithmic sophistication.

Federal Express Corporation has the only computerized procedure that we are aware of in the area of plane scheduling. The timetable at Federal Express is changed every four to five weeks to take into account changes in demand, new cities, seasonality factors, etc. Input to the process is a matrix which gives the estimated package count between each pair of cities. The package count at a city is converted into a percent of capacity of the plane. Every city is to be visited once (we are talking about Federal Express Corporation's Priority 1 packages only). In general, each package leaves its origin city on a plane, travels to Memphis, Tennessee where it get sorted by its destination location and is then delivered to its destination location on a second plane. Each city is regarded as both an origin and a destination and each city can have a one-sided time window for pickups and deliveries. One time window specifies the earliest time that a plane can leave the city with its pickups. The second time window gives the latest time that a plane can get to the city with its deliveries. There are also constraints on the total length of an incoming and outgoing route and the capacity of a plane.

Two routing problems are solved. One routing problem is for the pickups and the second routing problem is for the deliveries. A sample set of delivery routes is given in Fig. 4.5. These routes look incomplete in the sense of the routes displayed in Fig. 2.1 in that the routes do not return to Memphis, the depot. The reason for this is that the planes layover at the end of the outgoing or delivery routes and start their pickups from the layover cities. The routes, when formed, give the time for delivery and pickup; thus, the timetable is generated as the routes are created. An analogous algorithm is available for the pickup problem. In Section 4.7, a description of the algorithm for solving the delivery problem is given.

4.2.6 *Dial-a-ride routing and scheduling problems*

In recent years, the area of dial-a-ride routing and scheduling has received considerable attention. In the dial-a-ride problem, customers call a dispatcher or scheduler requesting service. Each customer specifies a distinct pickup and delivery point and, perhaps, a desired time for pickup or delivery. If all customers demand immediate service, then routing and scheduling is done in real time and the problem is referred to as the *dynamic* or *real time dial-a-ride problem*. If all customers call in advance, so that a complete database of customer demand is known before any routing or scheduling is carried out, then this problem is referred to as the *subscriber or static dial-a-ride problem*. Both the dynamic and static dial-a-ride

†Virtually all that is discussed in this section is due to Joe Hinson of Federal Express Corporation. We are indebted to him for his assistance in providing us with the background for this section.

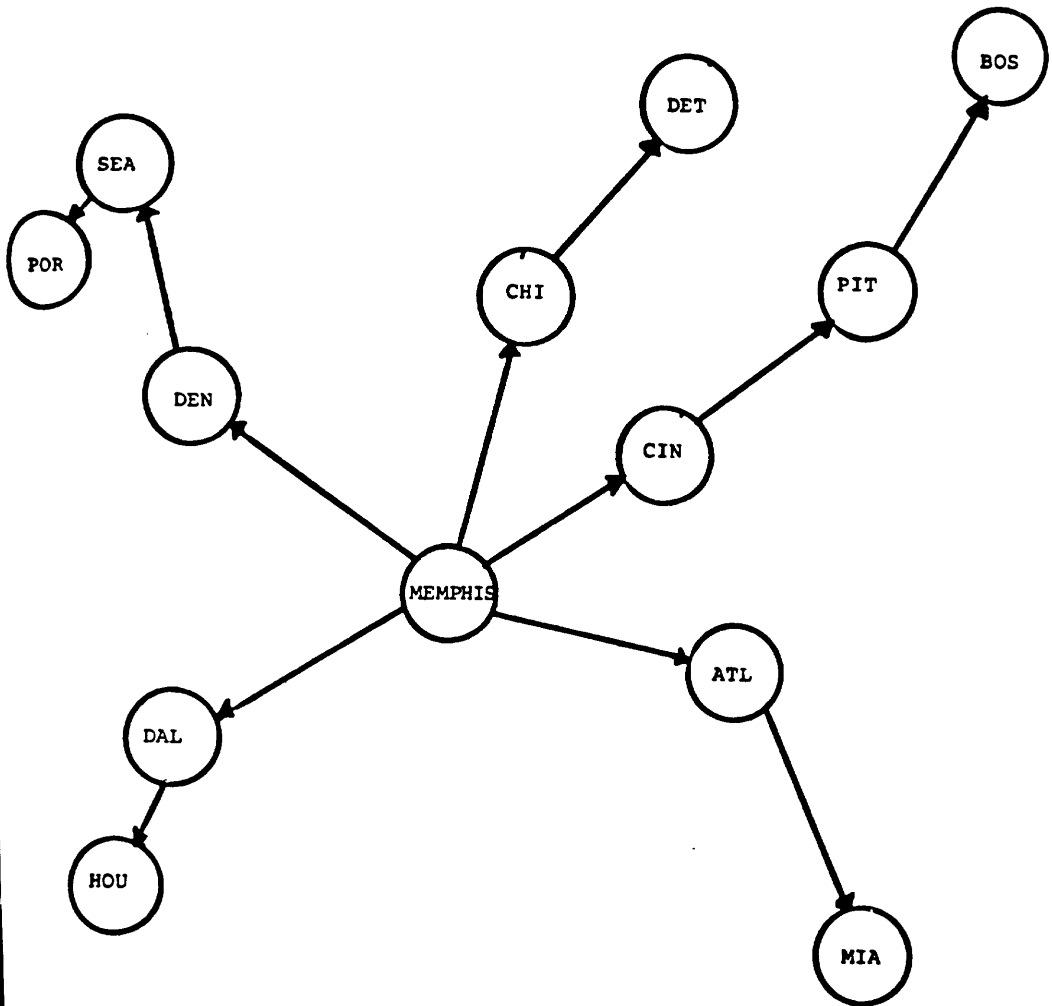


Fig. 4.5. Delivery routes for airplanes for an airplane routing and scheduling problem. (Memphis is the domicile for the planes.)

problems have precedence relationships since a customer must be picked up before he is delivered. In some situations, a desired time of pickup or delivery is specified in advance and the "other service" (either delivery or pickup) must be carried out within a given number of minutes from either the desired or the actual time of delivery or pickup. This situation introduces in a certain sense, a two-sided time window on the "other service".

A two-sided window introduces feasibility problems since it may be impossible (or very difficult) to handle the "other service" within its two-sided window. In [94] and [591], the two-sided window on the "other service" is disregarded and if a desired delivery time is specified, the delivery can take place any time before the desired delivery time and the pickup can be made at any time preceding the delivery time. Similarly, if a desired pickup time is stated, the pickup can be made at any time following the desired pickup time and the delivery can be made at any time following the pickup time. In essence, therefore, two-sided windows are replaced by a one-sided window on the activity with a desired pickup or delivery time and no time window is placed on the other activity. With one-sided windows, the potential exists, for customers to be on a vehicle for an excessive amount of time. However, experimentation with the algorithms (see [94], [591]) has shown that in this case most customers are still delivered within a reasonable amount of time.

Dial-a-ride problems and their extensions occur in many applications such as shared cab rides, package delivery, bank deliveries, etc. Perhaps the application that has received the most attention is the routing and scheduling of vehicles for social services such as for the elderly and