OPERATIONS RESEARCH CENTER

working paper

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

COMBINATORIAL OPTIMIZATION AND VEHICLE FLEET PLANNING:

PERSPECTIVES AND PROSPECTS

by

Thomas L. Magnanti

OR 106-81 January 1981

This research was supported in part by the National Science Foundation under Grant 79-26225-ECS.

ABSTRACT

As a well-structured and costly activity that pervades industries in both the public and private sector, vehicle fleet management would appear to be a splendid candidate for model-based planning and optimization. And yet, until recently the combinatorial intricacies of vehicle routing and of vehicle scheduling have precluded the widespread use of optimization (exact) methods for this problem class. Our discussion in this paper identifies the extent and nature of these problem complexities and draws contrasts with other applications of combinatorial optimization. It also summarizes a number of successful uses of optimization for vehicle fleet planning and highlights potentially fruitful avenues for algorithmic development.

In particular, we describe several alternative models and novel algorithms for the vehicle routing problem, show how various modeling approaches for this problem are intimately related, and illustrate the interplay between model formulations and the algorithms that they suggest. This discussion shows that prospects for applying exact methods, possibly in conjunction with heuristics, are far from fully realized and points to vehicle fleet planning as a tempting target of opportunity for further investigation.

1. INTRODUCTION

Vehicle fleet planning is a generic and broad class of practical decision making problems. Its applications encompass such diverse activities as retail distribution (e.g. [64]), school bus routing ([15], [99]), mail and newspaper delivery ([60]), municipal waste collection ([14]), fuel oil delivery ([49]), and airline and railway fleet routing and scheduling ([3], [4], and [116]). In addition, seemingly unrelated applications such as machine scheduling or communication systems management often lead to planning issues and models very similar to those that arise in vehicle fleet planning (identify machines or information packets with vehicles). As might be expected, the underlying assumptions associated with vehicle fleet planning are almost as diverse as the applications themselves. Bodin and Golden [16] and Schrage [112] summarize many of these assumptions and indicate how they affect model-based approaches to resource planning.

Despite this variety, almost any audit of comtemporary distribution and transportation systems would identify two prevailing planning issues; namely, the routing of (capacitated) vehicles through a collection of demand points to pick-up or deliver goods, the *vehicle routing problem,* and the scheduling of vehicles to meet timing or precedence restrictions imposed upon the vehicles' routes, the *vehicle scheduling problem.* The intrinsic combinatorial nature of routing and scheduling problems suggests, at least in principle, that almost every vehicle planning problem can be formulated and solved as an integer programming problem. And yet, very few studies document the use of optimization algorithms of an integer programming variety for these problems. Instead, historically, when confronted with vehicle fleet planning problems, transportation analysts and operations research practitioners have usually resorted to heuristic algorithms.

What, then, is the role of optimization in the context of vehicle routing and scheduling? In what limited circumstances has optimization proved to be successful? Is the notion of an optimal solution an outdated ideal that should be abandoned in most instances in favor of a less demanding planning objective? Do vehicle routing and scheduling problems differ from other combinatorial optimization problems in their use and potential use of optimization? And, what are prospects for future breakthroughs in optimization based methods for vehicle routing and scheduling?

In discussing research related to these issues, this paper briefly surveys the use of optimization methods for vehicle fleet planning, contrasts optimization based procedures with heuristic methods, and comments on recent developments that suggest future research directions. The discussion is intended to delineate some of the uses, advantages, and limitations of exact optimization, rather than to describe particular optimization studies or techniques in detail. To help provide some historical perspective, we begin by contrasting the evolution of analytic modeling for vehicle fleet planning with other areas of combinatorial optimization.

Our coverage of model-based approaches to vehicle fleet planning is not intended to be comprehensive, particularly with respect to the vast literature devoted to the traveling salesman problem. Rather, we cite representative results. Bellmore and Nemhauser [13] summarize research on the traveling salesman problem through the late 1960's and Golden and Magnanti [59] cite many references concerning the traveling salesman, vehicle routing, and vehicle scheduling problems through the mid 1970's in their extensive bibliography on network optimization. Several earlier books and surveys ([19], [36], [86], [94], and [104]) are additional sources that discuss optimization-based methods for the vehicle routing and related topics.

2. TRAVELING SALESMAN PROBLEM AND COMBINATORIAL OPTIMIZATION: TRENDS AND INFLUENCES

The traveling salesman problem is the most basic, and easily the most intensely studied, version of the vehicle routing problem. The last two generations of combinatorial mathematicians and operations research analysts cumulatively have devoted literally hundreds of man years to its study.

Since the routing of a single vehicle through the collection of demand points assigned to it is an essential component in most vehicle routing applications, one might expect that algorithms for solving vehicle routing problems would evolve and progress as dramatically as algorithms for the traveling salesman problem. And yet, the traveling salesman problem is more representative of combinatorial optimization in general than of vehicle fleet planning. In fact, it is remarkable how often new methods for solving the traveling salesman problem have precipitated general techniques in combinatorial optimization, and yet how infrequently these techniques have been applied successfully to vehicle fleet planning.

2.1 Early Developments

From even the earliest studies of discrete models, the traveling salesman problem has been a major stimulant to combinatorial optimization. Early studies of the traveling salesman problem pioneered the use of cutting plane techniques in integer programming ([31]) and were responsible for several important ideas associated with tree enumeration methods including coining the term branch and bound ([84]). They also introduced problem partitioning and decomposition techniques in the context of dynamic programming ([65]) that have later proved to be fruitful in other applications of dynamic programming and in assessing heuristic methods for combinatorial optimization. An isolated probabilistic study of the traveling salesman problem in the plane ([12]) has recently become widely recognized as a seminal contribution to the probabilistic evaluation of heuristic methods for combinatorial optimization problems.

Many contributions to combinatorial optimization throughout the 1950's and 1960's, for such problem classes as machine scheduling and production planning with set-up costs, crew scheduling, set covering, and facility location problems were extensions and generalizations on these basic themes Research focused on the design of optimization algorithms, usually based upon dynamic programming recursions or somewhat tailored versions of general-purpose integer programming methods, often for special cases in the problem class. Studies of scheduling theory as summarized by Conway, Maxwell and Miller [24] and of uncapacitated inventory and production lot size planning by dynamic programming ([125]) are prototypes of this period as are branch and bound methods ([33], [38]) for plant location problems.

At the same time that these integer and dynamic programming methods were evolving, combinatorial optimization was emerging and flourishing as a discipline in applied mathematics, based, in large part, on the widespread practical and combinatorial applications of network flow theory ([45]) and its generalizations such as nonbipartite matching and matroid optimization ([78]). Indeed, it is hard to understate the importance of these landmark contributions in defining combinatorial optimization as we know it today. In a recent survey, Klee [75] summarizes much of the latest research devoted to these topics.

Although researchers were designing and applying heuristic algorithms during the 1950's and 1960's (for example, exchange heuristics for both the traveling salesman and facility location problems ([84], [90]) and a "greedy-like" heuristic for warehouse location ([76])), optimization-based methods remained at the forefront of academic activity. The heuristic algorithms developed at this time may have been progenitors of algorithms studied later, but their analysis was often of such a rudimentary nature that heuristics did not capture the imagination and full acceptance of the academic community. Limited empirical verification of heuristics, rather than statistical assessment or error bound analysis, ruled the day.

2.2 New Directions

Two new developments in the early and mid 1970's, namely the emergence of computational complexity theory and the evolution of enhanced capabilities in mathematical programming, revitalized combinatorial optimization and precipitated a new focus in its research. The now familiar computational complexity theory ([25], [73]) shows that the traveling salesman problem and nearly every other "difficult" combinatorial problem, the so called NP-complete class of problems, are all computationally equivalent; namely, each of these problems has eluded any algorithmic design guaranteed to be more efficient than tree enumeration, and if one could be solved by an algorithm that is polynomial in its problem size, then they all could. This revelation suggested that algorithmic possibilities for optimizaiton methods were limited and motivated renewed interest to design and *analyze* effective heuristics. Again, the traveling salesman problem was at the forefront. Worst case (i.e., performance guarantee) analysis ([20], [109]), statistical analysis ([56]), and probabilistic analysis ([71], [72]) of various heuristics for the traveling salesman problem typified this period of research and were among the first steps in the evolution of new analytic approaches for evaluating heuristic methods. Indeed, the mere fact that computational complexity theory embraced the "infamous" traveling salesman problem undoubtedly was instrumental in the theory's acceptance as a new paradigm in computer science and operations research.

Computational complexity theory has now become pervasive, so much so that Garey and Johnson's recent and comprehensive monograph [48] discusses more than 300 combinatorial applications. Lenstra and Rinnooy Kan's [81] summary of computational complexity as applied specifically to vehicle routing and scheduling shows that most of these problems are NP-complete. As a consequence, vehicle fleet planning would appear to be a prime candidate for analysis by heuristic methods.

At approximately the same time that the first results in computational complexity were emerging, so too were new advances in mathematical programming, particularly effective new methods for decomposing integer programming models and improved implementations of basic network algorithms. The new implementations of network flow algorithms, shortest path algorithms, and other fundamental network procedures based on enhanced data manipulation techniques (see [48], [74] and [86] for summaries and for citations to the literature) gave further impetus to the development of problem decomposition, since the iterative solution of these frequently appearing modeling components became more and more attractive.

Again, the traveling salesman problem was a major source of inspiration. One of the first triumphs of integer programming decomposition involved the use of Lagrange multiplier techniques, used previously with so much success in non-linear programming, for the traveling salesman problem (Held and Karp [66], [67]). Essentially, attaching a vector of multipliers u with all but the subtour breaking constraints of a traveling salesman problem (see problem formulation (5) stated later) decomposes the model into an easily solved variant of the minimal spanning tree problem whose optimal objective value $C^*(u)$ is a lower bound (see section 3.2.1) on the minimum cost C^* of the traveling salesman problem. Since $C^*(u) \leq C^*$ for any u, the value $d \equiv \max_u C^*(u)$ of the *Lagrangian dual* to the traveling salesman problem is also a lower bound on C^*. This bounding technique is a natural candidate to be embedded within a branch and bound procedure. Computational experience ([11], [63]) has borne out this promise.

This type of Lagrangian decomposition, or relaxation as it is often called, has now become a staple of combinatorial optimization. Fisher and Shapiro [43] and Geoffrion [52] describe uses and properties of the method as applied to combinatorial optimization in general. Magnanti, Shapiro and Wagner [87] point out the relationship with generalized linear programming and convexification of the model being studied, and Magnanti [85] emphasizes the close ties between Lagrangian relaxation and other iterative decomposition methods such as Benders decomposition.

Decomposition has now been applied successfully in many combinatorial problems including distribution systems design ([53]), power generation systems scheduling ([96]), facility location selection ([26]) and personnel scheduling ([115]). Surveys by Fisher [39] and Shapiro [114] summarize a number of other applications as well.

Cumulatively, this fertile decade of research has yielded much improved capabilities for applying optimization methods to combinatorial problems, capabilities that tend to counterbalance the trend, stimulated by computational complexity theory, toward heuristic methods. As a consequence, both exact algorithms and heuristic methods provide exciting, and as yet not fully realized, opportunities for improved model-based approaches to vehicle fleet planning.

3. OPTIMIZATION AND VEHICLE ROUTING

To this point, our discussion has portrayed a research environment in the 1950's and 1960's conducive to the application and tailoring of general purpose optimization techniques and in the 1970's conducive to the design both of special purpose decomposition methods and of heuristic algorithms supported by error bound analysis. Despite the prominence of the traveling salesman problem in forging and sustaining this research momentum, the analysis of more general vehicle routing problems has, over the years, been decidedly distinct in flavor.

Since 1959, when Dantzig and Ramser [32] first introduced the vehicle routing problem and proposed a linear programming based heuristic for its solution, the overwhelming majority of attempts at solving the problem have focused on heuristic methods (e.g. [23], [55] and [60]). As Christofides [19], [22], who has been a leading proponent of exact algorithms, has acknowledged, the largest vehicle routing problem of any complexity solved to date by exact methods and reported in the open literature contains only 31 demand points. This capability contrasts sharply with achievements in other areas of combinatorial optimization: several studies now demonstrate that exact algorithms are capable of solving traveling salesman problems ten times as large ([27], [108]), facility design problems with hundreds of customer zones and with dozens of potential locations for distribution centers ([53]), and facility location models with over 150 potential site selections ([26]). The success of optimization-based procedures for vehicle routing has been limited, in part, by the fact that, until most recently (see section 3.2), algorithms for this problem have relied on specialized versions of branch and bound methods ([21], [104]) and of cutting plane techniques ([7], [46]); they have not exploited recent advances in integer programming.

Computational complexity theory has also had little impact on vehicle routing. With the exception of one study of the worst case behavior of a sequential vehicle routing heuristic ([57]) and one statistical study of another vehicle routing heuristic ([58]), limited empirical experience has been the sole mechanism used to assess heuristic algorithms for this problem class.

Consequently, from almost the beginning, research on this problem has been somewhat out of phase with general attitudes and trends in combinatorial optimization. Other aspects of vehicle fleet planning, particularly special versions of vehicle fleet planning that are amenable to dynamic programming analysis, have, however, been more representative of combinatorial optimization. We consider these problems in section 4.

The vehicle routing problem is so prevalent in practice that it is often viewed as synonomous with vehicle fleet planning. For this reason, and since the problem has attracted so much research attention, we have chosen to emphasize vehicle routing in our discussion. Moreover, vehicle routing is a setting that will permit us to illustrate some of the obstacles that have limited the use of exact methods for vehicle fleeting planning in general, and to illustrate the interplay between model formulations and the solution procedures that they suggest.

3.1 Model Formulations

Integer programming models of the vehicle routing problem are quite varied. In fact, by considering various problem features it is possible to identify a wide spectrum of overlapping and interrelated models. Nevertheless, most of these models are elaborations and twists on three basic approaches to be discussed in this section - namely a set covering formulation, a commodity flow based formulation, and a vehicle flow based formulation. Although these basic formulations have been valuable in the past in identifying and highlighting planning issues for the vehicle routing problem, for the most part they have been ignored in algorithm development. Only recently have researchers tapped the algorithmic potential of these formulations. In the next subsection we describe some of these novel methods.

Throughout our discussion, we assume that a fleet with NV capacitated vehicles, domiciled at a common depot, must be routed to deliver goods to n demand points with specified delivery requirements $d_1, d_2, \ldots, d_n$ and must subsequently return to the depot. Each of the models[†] that we discuss can be extended to incorporate multiple depots, maximum route time restrictions, time windows imposed upon deliveries at each point, and many other practical issues.

We shall suppose that the cost (distance) c_{ij} of any vehicle's traveling from point i to point j is known and that our objective is to minimize total routing costs. We let index i=0 or j=0 denote the depot.

SET COVERING FORMULATION ([7])

This formulation is representative of a cluster-first route-second approach to vehicle routing in which demand points are first assigned to

[†] But not necessarily the solution strategies.

vehicles, the "vehicle clusters," and then each vehicle is routed over the demand points assigned to it to determine a delivery sequence. In principle, the formulation requires the enumeration of every possible assignment of the demand points into vehicle clusters.

The decision variables z_j are binary and specify whether or not cluster j is used, $z_j = 1$, or not, $z_j = 0$. Let $a_{ij} = 1$ if demand point i is assigned to cluster j and $a_{ij} = 0$ otherwise. These coefficients are fixed and defined for each cluster j.

The formulation is:

$$\left.\begin{array}{lll} \text{Minimize} & \sum_{j=1}^{J} \hat{c}_j z_j & \\ \text{subject to:} & \sum_{j=1}^{J} a_{ij} z_j = 1 & (i = 1,2,\ldots,n) \\ & z_j = 0 \text{ or } 1 & (j = 1,2,\ldots,J). \end{array}\right\} \quad (1)$$

The constraints state that each demand point i must be assigned to exactly one of the possible clusters $j = 1,2,\ldots,J$. The objective coefficient $\hat{c}_j$ is the minimum cost of any vehicle route passing through the demand points i assigned to the j^{th} cluster (i.e. the demand points i with $a_{ij} = 1$). This routing cost is the solution to a traveling salesman problem!

Note that the coefficients a_{ij} are required to satisfy any restrictions imposed upon the cluster assignments. For example, if K is the capacity of any vehicle in the fleet, which we assume for now to be homogeneous, then the 0 - 1 coefficients a_{ij} for any cluster j must satisfy the constraint

$$\sum_{i=1}^{n} a_{ij} d_i \leq K. \quad (2)$$

In adopting this formulation, Balinski and Quandt ([7]) overcome the difficulty in enumerating all clusters by preselecting a small "attractive" subset. They then use a cutting plane algorithm to solve the resulting set covering problem (1) with computational experience limited to 15 delivery points. Foster and Ryan ([46]) suggest a refinement that incorporates a richer collection of route restrictions and automates the selection of attractive clusters. They choose a small number J of "petal" clusters defined by grouping points that are adjacent in radial regions along rays emanating from the depot. This combined heuristic-optimization procedure permits consolidating the benefits of sweep type heuristic methods ([55]) with the strengths of an integer programming formulation. It has been applied to problems containing as many as 100 delivery points.

Observe that a slightly altered version of formulation (1) will accommodate nonhomogeneous vehicle fleets. Let $\mathcal{C}_v$ for v = 1,2,...,NV be the set of candidate clusters for vehicle v. These would be the same for all vehicles if the fleet were homogeneous. In any case, we add to (1), the multiple choice constraints

$$\sum_{j \in \mathcal{C}_v} z_j = 1 \qquad (v = 1,2,\ldots,NV) \tag{3}$$

stating that each vehicle is assigned exactly one group of customers, possibly the null group. The restrictions (2) imposed upon the assignment of demand points to clusters then become vehicle dependent; for any cluster j in the candidate set $\mathcal{C}_v$, the capacity K_v of vehicle v, replaces the constant K in (2) to give:

$$\sum_{i=1}^{n} a_{ij} d_i \leq K_v \qquad \text{for all } j \in \mathcal{C}_v. \tag{4}$$

VEHICLE FLOW BASED FORMULATION ([60])

This formulation is the most direct extension of the assignment based formulation of the traveling salesman problem. The decision variables x_{ij}^{v} which are binary indicate whether vehicle v travels from point i to point j, $x_{ij}^{v} = 1$, or not, $x_{ij}^{v} = 0$. The formulation is:

$$\text{minimize} \quad \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{v=1}^{NV} c_{ij} x_{ij}^{v}$$

$$\text{subject to:} \quad \sum_{v=1}^{NV} \sum_{i=0}^{n} x_{ij}^{v} = 1 \qquad (j = 0,1,\ldots,n) \qquad (5.1)$$

$$\sum_{v=1}^{NV} \sum_{j=0}^{n} x_{ij}^{v} = 1 \qquad (i = 0,1,\ldots,n) \qquad (5.2)$$

$$\sum_{i=0}^{n} x_{ip}^{v} - \sum_{j=0}^{n} x_{pj}^{v} = 0 \qquad (p = 1,2,\ldots,n;\ v = 1,2,\ldots,NV) \qquad (5.3)$$

$$\sum_{j=1}^{n} x_{0j}^{v} \leq 1 \qquad (v = 1,2,\ldots,NV) \qquad (5.4)$$

$$\sum_{i=1}^{n} d_i \left(\sum_{j=0}^{n} x_{ij}^{v} \right) \leq K_v \qquad (v = 1,2,\ldots,NV) \qquad (5.5)$$

$$x_{ij}^{v} = 0 \text{ or } 1 \qquad \text{all } i,j,v \qquad (5.6)$$

$$X \in S. \qquad (5.7)$$

Constraints (5.1) - (5.3) insure that one, and the same, vehicle enters and leaves each delivery site. Inequality (5.5) models the demand limitations imposed by the capacity K_v of each vehicle. The last condition which is imposed on the matrix X with entries x_{ij}^{v} prohibits subtours not containing the depot. There are several possible ways to fulfill this condition. For example, S might be composed of circuit breaking constraints imposed upon each vehicle type, that is S might be the union of sets S_v defined by:

$$S_v = \left\{x_{ij}^v : \sum_{i\varepsilon Q} \sum_{j\varepsilon Q} x_{ij}^v \leq |Q| - 1 \text{ for every nonempty subset } Q \text{ of } \{1,2,\ldots n\}\right\}. \quad (5.8)$$

Alternatively, the subtour breaking set S can be written with fewer circuit breaking inequalities by imposing constraints on the aggregate variables $x_{ij} \equiv \sum_v x_{ij}^v$ instead of on each individual x_{ij}^v. The form (5.8) is more useful in the algorithms that we will discuss subsequently since, in this form, the constraints separate by vehicle type.

Note that when NV = 1, constraint (5.3) is redundant as is constraint (5.5) unless $\sum_{i=1}^{n} d_i$ exceeds $K \equiv K_1$ in which case the problem is inconsistent. Consequently, for applications with a single vehicle this formulation reduces to the usual assignment-based model of the traveling salesman problems.

<u>COMMODITY FLOW BASED FORMULATION</u> ([49])

This formulation combines assignment constraints, like those used in the standard formulation of the traveling salesman problem, for modeling vehicle movements together with multicommodity flow constraints modeling movements of goods. The goods destined for any demand point are viewed as a separate commodity.

The decision variables x_{ij} are binary and indicate whether a vehicle moves from demand point i to j, $x_{ij} = 1$, or not, $x_{ij} = 0$. The decision, or flow, variables y_{ij}^k specify how much of the demand destined for point k is transported from point i to point j. The formulation is:

$$\text{Minimize} \quad \sum_{i=0}^{n} \sum_{j=0}^{n} c_{ij} x_{ij}$$

$$\text{subject to:} \quad \sum_{i=0}^{n} x_{ij} = 1 \qquad (j = 1,2,\ldots,n) \qquad (6.1)$$

$$\sum_{j=0}^{n} x_{ij} = 1 \qquad (i = 1,2,\ldots,n) \qquad (6.2)$$

$$\sum_{j=1}^{n} x_{0j} \leq NV \qquad (6.3)$$

$$\sum_{i=0}^{n} y_{ij}^{k} - \sum_{\ell=0}^{n} y_{j\ell}^{k} = \begin{cases} d_k & \text{if } j=k \\ 0 & \text{if } j \neq 0 \text{ or } j \neq k \end{cases} \qquad \text{all } j,k \qquad (6.4)$$

$$\sum_{k=1}^{n} y_{ij}^{k} \leq Kx_{ij} \qquad (i \neq j = 0,1,\ldots,n) \qquad (6.5)$$

$$y_{ij}^{k} \geq 0 \qquad \text{all } i,j,k \qquad (6.6)$$

$$x_{ij} = 0 \text{ or } 1 \qquad \text{all } i,j. \qquad (6.7)$$

The first two sets of constraints insure that exactly one vehicle enters and leaves each demand site. Inequality (6.3) restricts the number of available vehicles and (6.4) are mass balance constraints modeling the movement of goods. The "forcing constraint" (6.5) insures that good movement from i to j does not exceed vehicle capacity, and, if no vehicle travels from i to j, i.e. $x_{ij} = 0$, then no goods are shipped on this transportation link.

To extend this formulation in order to model nonhomogeneous fleets, we can replace (6.1) - (6.3) by (5.1) - (5.4) and replicate constraints (6.5) NV times, with $K_v x_{ij}^v$ replacing Kx_{ij} in the v^{th} replication.

3.2 Algorithms

When viewed as general purpose integer or mixed integer programming models, the set partitioning, vehicle flow based, and commodity flow based formulations of the vehicle routing problem all appear to be severely limited. The size of each of these models far exceeds the current capabilities of even the most sophisticated general-purpose integer programming codes. For example, an application with 100 demand points and only 5 vehicles gives rise to a 100 constraint, 2^{100} variable set partitioning model; a 710 constraint, 50,000 discrete variable vehicle flow based model (even without subtour breaking constraints); and a 20,301 constraint, 10,100 discrete variable, and about 1 million continuous variable commodity flow based model. It is no wonder that these formulations have not stimulated widespread development of exact algorithms, particularly in the 1950's and 1960's during the embryonic stages in the evolution of integer programming techniques.

Nevertheless, the underlying network components of these formulations are so prominent that each is a promising candidate for specialized problem decomposition that exploits embedded structure. In this section we summarize several possibilities of this nature. We begin by discussing two rather general algorithmic strategies - price directive and resource directive decomposition. These solution strategies not only suggest a host of algorithmic possibilities, but also reveal intimate connections between the apparently disparate modeling approaches derived from set partitioning, vehicle flow, or commodity flow formulations. Finally, we discuss two optimization-based heuristic methods that have proved to be very successful in solving a variety of vehicle routing problems.

3.2.1 Lagrangian Relaxation

Vehicle Flow Formulation

We begin by considering the vehicle flow based formulation (5.1) - (5.8). Since constraints (5.2) and (5.3) imply (5.1), we can eliminate (5.1) from the formulation. Then, except for the constraints (5.2), the variables x_{ij}^{v} for different vehicles do not appear in any common constraints. Attaching prices, or Lagrange multipliers, $u_1, u_2, \ldots, u_n$ with these constraints and bringing them into the objective function as $\sum_{i=1}^{n} u_i(1 - \sum_{v=1}^{NV} \sum_{j=0}^{n} x_{ij}^{v})$ gives the modified problem:

$$\text{Minimize} \quad \sum_{i=0}^{n} \sum_{j=0}^{n} \sum_{v=1}^{NV} \bar{c}_{ij}^{v} x_{ij}^{v} + \sum_{i=0}^{n} u_i$$

$$\text{subject to:} \quad (5.3) - (5.8). \tag{7}$$

Here, in contrast to our subsequent discussion, the modified cost coefficients $\bar{c}_{ij}^{v} = c_{ij} - u_i$ are independent of the vehicle type v. That is, here $c_{ij}^{v} = c_{ij}^{v'}$ if $v \neq v'$.

This problem decomposes into a separate subproblem for each vehicle. Its solution, which is the minimum cost vehicle route through all subsets of points whose demand does not exceed vehicle capacity, is reminiscent of the coefficient and cost structure associated with any column in the set covering formulation (1). Let us pursue this connection further.

First, recall the familiar bounding, or weak duality, argument of Lagrangian relaxation: since any feasible solution to (5.1) - (5.8) is feasible in (7) and has exactly the same objective value in both problems and since (7) might have other more cost effective solutions that are not feasible in (5.1) - (5.8), the minimum cost C*(u) to (7) is no larger than the minimum cost C* to formulation (5). The sharpest of these lower bounds, which generally will be strictly less than C*, is determined by solving the *Lagrangian dual* *problem* of maximizing C*(u) over all n-vectors u. There are several ways of performing this maximization including dual ascent or subgradient optimization ([85]). Generalized linear programming, which is still another method for solving the Lagrangian dual problem ([87]), can be viewed as a solution strategy that reformulates the vehicle flow based model in a form very similar to the set partitioning model (1). To see this, recall that the generalized linear programming algorithm expresses any solution x_{ij}^{v} to the constraints (5.2) as a convex combination $x_{ij}^{v} = \Sigma\theta_k x_{ij}^{v(k)}$ of a finite number of feasible solutions $x_{ij}^{v(k)}$, indexed by k in some index set $\mathcal{C}_v$, to the remaining constraints (5.3) - (5.8). Making this substitution in (5.2) gives the revised formulation:

$$\text{Minimize} \quad \sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{v=1}^{NV} c_{ij} \sum_{k\varepsilon\mathcal{C}_v} \theta_k x_{ij}^{v(k)} = \sum_{v=1}^{NV}\sum_{k\varepsilon\mathcal{C}_v} \hat{c}_k \theta_k$$

$$\text{subject to} \quad \sum_{v=1}^{NV}\sum_{j=0}^{n}\sum_{k\varepsilon\mathcal{C}_v} \theta_k x_{ij}^{v(k)} \equiv \sum_{v=1}^{NV}\sum_{k\varepsilon\mathcal{C}_v} a_{ik}\theta_k = 1 \quad \text{all } i \qquad (8.1)$$

$$\sum_{k\varepsilon\mathcal{C}_v} \theta_k = 1 \qquad (v = 1,2,\ldots,NV) \qquad (8.2)$$

$$\theta_k \geq 0. \qquad \text{all } v \text{ and all } k_v \qquad (8.3)$$

In this formulation, the 0 - 1 coefficients $a_{ik} \equiv \sum_{j=0}^{n} x_{ij}^{v(k)}$ for any vehicle v and any $k\varepsilon\mathcal{C}_v$ defines the set of demand points covered by the vehicle solution $x_{ij}^{v(k)}$; $\hat{c}_k$ is the associated routing cost. Note that with θ_k identified with z_k this generalized programming formulation (8) is just the linear programming relaxation to the set covering problem (1) and (2). The optimal linear programming dual variables to the constraints (8.1) solve the Lagrangian dual of maximizing C*(u) over u. Consequently, the linear programming relaxation of the set covering problem is identical to this Lagrangian dual. This observation not only delineates an intimate connection between the set covering and vehicle flow formulation of the vehicle routing problem, but also suggests a number of algorithmic possibilities for the set covering formulation. Generalized linear programming, which generates coefficients a_{ik} and $\hat{c}_k$ for the variables θ_k when needed, is a column generation procedure for solving the linear programming relaxation of the set covering formulation. Any other scheme for solving the Lagrangian dual problem of maximizing $C^*(u)$ over u is an alternative solution strategy for solving this linear programming relaxation.

There are a number of other ways to apply Lagrangian relaxation to the vehicle flow based formulation. For example, we could associate Lagrange multipliers γ_v with the vehicle capacity constraints (5.5) and Lagrange multipliers w_j with constraints (5.1).† The Lagrangian relaxation (7) with cost coefficients $\bar{c}_{ij}^v = c_{ij} - u_i - w_j - \gamma_v d_i$ and constraints (5.3), (5.4), (5.6) and (5.7) then becomes an unconstrained routing problem in which only a subset of the demand nodes need be covered on the vehicle routes. Or, if we assume, by adding fictitious demand points if necessary, that every vehicle must be dispatched to at least one demand point, we could add the redundant constraints $\sum_{j=1}^{n} x_{0j} = NV$, $\sum_{i=1}^{n} x_{i0} = NV$ and $\sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij} = n - NV$, where $x_{ij} = \sum_{v=1}^{NV} x_{ij}^v$, and apply Lagrange multipliers to all but these constraints and the subtour breaking constraints defined in terms of the aggregate variables x_{ij} (see the comment following (5.8)). Then the solution to the Lagrangian relaxation (7) is a minimal cost tree with n - NV arcs on the demand nodes 1,2,...,n together with the NV least cost arcs emanating from the depot plus the NV least cost arcs directed into the depot, all with respect to the modified costs $\bar{c}_{ij}^v$. When $NV = 1$ and $K \geq \sum_{i=1}^{n} d_i$, this relaxation reduces to the Held and Karp relaxation that we have mentioned previously for solving the traveling salesman problem.

†Although (5.1) is redundant in the integer programming formulation, it need not be redundant in a Lagrangian relaxation.

Christofides, Mingozzi and Toth [22] have recently proposed and tested an algorithm that is a slight modification of this last approach.† We will interpret their results as arising directly from a Lagrangian relaxation.

They note, again assuming that every vehicle in the fleet must be dispatched to at least one demand point, that every feasible solution $x_{ij} = \sum_{v=1}^{NV} x_{ij}^{v}$ to the vehicle flow based formulation decomposes into a spanning tree on the nodes 0,1,2,...,n having degree 2(NV) at the depot, node 0, together with NV additional arcs. That is, $x_{ij} = x'_{ij} + x''_{ij}$ where x'_{ij} defines a 0 - 1 incidence vector for the arcs i - j in the spanning tree; that is, it satisfies $\sum_{j=1}^{n} x'_{0j} + \sum_{i=1}^{n} x'_{i0} = 2(NV)$ as well as subtour breaking constraints on the nodes 0,1,2,..., n. The variables x''_{ij} define a 0 - 1 incidence vector on the arcs i - j with $i \neq 0$ and $j \neq 0$ and with exactly NV components equal to 1. Consequently, dualizing all but the subtour breaking constraints on x_{ij} and the constraints $\sum_j x'_{0j} + \sum_i x'_{i0} = 2(NV)$, $x_{ij} = x'_{ij} + x''_{ij}$, $x''_{0j} = x''_{i0} = 0$, and $\sum_{i=1} \sum_{j=1} x''_{ij} = NV$ gives a Lagrangian relaxation whose solution is a constrained minimum spanning tree with degree 2(NV) at the depot together with the NV additional least cost arcs not incident to the depot. The authors' algorithm is actually somewhat more complicated since they permit the computation of minimal spanning trees with degree k, $NV \leq k \leq 2(NV)$, at the depot together with the cheapest 2(NV) - k additional arcs incident to the depot, and cheapest k - NV arcs not incident to the depot. Again, the algorithm can be veiwed in terms of Lagrangian relaxation, now by replacing the constraints on x'_{ij} and x''_{ij} with $\sum_{j=1}^{n} x'_{0j} + \sum_{i=1}^{n} x'_{i0} = k$, $\sum_{j=1}^{n} x''_{0j} + \sum_{i=1}^{n} x''_{i0} = 2(NV) - k$ and $\sum_{i=1}^{n} \sum_{j=1}^{n} x''_{ij} = k - NV$. The implementation of this algorithm, which uses a subgradient optimization algorithm to maximize the Lagrangian dual and applies the Lagrangian relaxation within the context of a branch and bound procedure, has been successful in solving problems with from 10 to 20 demand points and 3 to 6 vehicles.

---

†They also discuss another algorithm that we do not describe.

As our final example of Lagrangian relaxation for the vehicle flow based formulation, we note that associating Lagrange multipliers γ_v to dualize the capacity constraints (5.5) results in a multiple traveling salesman problem. That is, in the Lagrangian relaxation exactly one of the "uncapacitated" vehicles must pass through each demand point and in a way that minimizes total routing costs, where routing costs $\bar{c}_{ij}^{v} = c_{ij} - \gamma_v d_i$ now include a loading charge $\gamma_v d_i$ for using vehicle capacity. Stewart and Golden [119] have proposed a modification of this procedure for routing problems with a homogeneous fleet. They rewrite the demand constraints as

$$\sum_{i=1}^{n} d_i \left(\sum_{j=0}^{n} x_{ij}^{1} \right) \leq K$$

$$\text{and} \quad \sum_{i=1}^{n} d_i \left(\sum_{j=0}^{n} x_{ij}^{v} \right) \leq \sum_{i-1}^{n} d_i \left(\sum_{j=0}^{n} x_{ij}^{v-1} \right) \quad (v = 2,3,\ldots,NV)$$

and then associate a Lagrange multiplier λ with only the first of these constraints; the Lagrangian relaxation is a multiple traveling salesman problem with penalty of λ units imposed upon the largest demand route (i.e., $\bar{c}_{ij}^{1} = c_{ij}^{1} - \lambda$). To solve this problem, they use a modified exchange heuristic. (See [119] for details).

Commodity Flow Based Formulation

Lagrangian relaxation can be applied to the commodity flow based formulation in much the same way as it is applied to the vehicle flow based model. Since the concepts and methods are much the same in both instances, we will be more concise in this section.

First note that applying Lagrange multipliers to dualize the forcing constraints (6.5) of the commodity flow based formulation and transfer them into the objective function produces a constraint set with assignment constraints in terms of the variables x_{ij} and flow constraints in terms of the now uncapacitated variables y_{ij}^k. Consequently, the Lagrangian relaxation decomposes into an assignment problem and one shortest path problem for each flow commodity k. One disadvantage of this approach is that it requires a large number, n^2, of Lagrange multipliers. An alternative would be to dualize the assignment constraints (6.1) through (6.3) with multipliers v_j, u_i and u_0 respectively. The Lagrangian relaxation is then solved as follows. First we use the fact that each variable x_{ij} only appears in one forcing constraint of the relaxation to eliminate these variables; that is, if $\bar{c}_{ij} = c_{ij} - u_i - v_j \leq 0$ in the relaxation then it is optimal to set $x_{ij} = 1$; if $\bar{c}_{ij} > 0$, it is optimal to set x_{ij} as small as possible; that, is $x_{ij} = \frac{1}{K} \sum_{k=1}^{n} y_{ij}^k$. In the second case, we substitute for x_{ij} in the objective function in terms of the y_{ij}^k's and replace the constraint $x_{ij} \leq 1$ with $\sum_{k=1}^{n} y_{ij}^k \leq K$. These manipulations show that the Lagrangian subproblem is a multicommodity flow problem in the variables y_{ij}^k.

These two different relaxations of the commodity flow based formulation illustrate a common trade-off in Lagrangian approaches to discrete optimization problems -- one method requires more multipliers, which generally seems best to avoid for efficiency in maximizing C*(u) over u, whereas the second method requires a much more complicated Lagrangian subproblem to be solved in order to evaluate C*(u) for any given value of u. Neither of these methods has been tested, so any judgement at this point concerning their relative merits would be speculative.

Gavish and Graves [51] have proposed a series of models for a variety of vehicle routing type problems that suggest algorithms to overcome some of the limitations of the last two approaches. Their formulation of the vehicle routing problem can be viewed as an aggregation of the commodity flow based formulation. To obtain their model, we let $y_{ij} = \sum_{k=1}^{n} y_{ij}^{k}$ denote the total flow of goods from i to j and then sum the constraints (6.4) over k. The resulting model has constraints (6.1) - (6.3) and (6.7) as in the original formulation, but the following constraints replace (6.4) - (6.6):

$$\sum_{i=0}^{n} y_{ij} - \sum_{\ell=0}^{n} y_{j\ell} = d_j \qquad (j = 1,2,\ldots,n) \qquad (6.4')$$

$$y_{ij} \leq Kx_{ij} \qquad \text{for all i and j} \qquad (6.5')$$

$$y_{ij} \geq 0 \qquad \text{for all i and j.} \qquad (6.6')$$

Introducing Lagrange multipliers u_i and w_j to dualize the assignment constraints (6.1) - (6.3) in the new model, we note, as before, that either $\bar{c}_{ij}^{v} = c_{ij} - u_i - w_j \leq 0$ and $x_{ij} = 1$ in the Lagrangian relaxation or $\bar{c}_{ij}^{v} > 0$ and $x_{ij} = \frac{1}{k} y_{ij}$. Consequently, since the Lagrangian relaxation can be written solely in terms of the variables y_{ij}, it has become a single, rather than multi, commodity flow problem. Alternately, we can associate multipliers π_j with the commodity flow constraints (6.4'). For any set values of these multipliers, the Lagrangian relaxation requires the minimization of $\sum_{ij}\sum c_{ij}x_{ij} + \sum_{ij}\sum(\pi_i - \pi_j)y_{ij} + \sum_j u_j d_j$ subject to the constraints (6.1) - (6.3), (6.5), (6.6') and (6.7); since the variables y_{ij} appear only in the forcing constraints (6.5') in this relaxation, they are made as small or large as possible depending upon the sign of $(\pi_i - \pi_j)$. Thus, for any given values for the variables x_{ij}, optimal values for the variables y_{ij} are $y_{ij} = 0$ if $\pi_i - \pi_j > 0$ and $y_{ij} = Kx_{ij}$ if $\pi_i - \pi_j \leq 0$. Consequently, we can substitute for each y_{ij} in terms of x_{ij} to eliminate the forcing constraints and the Lagrangian relaxation becomes an easily solved assignment problem in the variables x_{ij}.

The last two Lagrangian approaches illustrate the advantages (fewer multipliers and/or more easily solved relaxation) of the aggregate formulation with constraints (6.4') - (6.6') and are representative of algorithmic possibilities that this formulation suggests. See [62] for more details and [50] where Gavish uses a similar formulation to devise Lagrangian algorithms to solve some degree-constrained minimal spanning tree problems that arise in computer network design. His experience, which indicates that solving computer design problems with up to 200 nodes requires only a few seconds of computer time, underscores the potential of Lagrangian methods as applied to the aggregate commodity flow formulation of vehicle routing problems.

The computational advantages of the aggregate commodity flow based formulation are not without costs. First of all, the linear programming relaxation of the original commodity flow based formulation (6.1) - (6.7) provides a tighter lower bound on the optimal objective value C* of the vehicle routing problem than does the linear programming relaxation of the aggregate commodity flow based formulation. In fact, examples (see [128]) show that lower bounds obtained from the linear programming relaxation of the aggregate formulation can be significantly inferior to those obtained from the detailed commodity flow based formulation. Moreover, Wong [128] has shown that when specialized to the traveling salesman problem (i.e., NV = 1, K = n and all $d_i = 1$), the linear programming relaxation of a modified version of the original formulation (6.1) - (6.7) is equivalent to the linear programming relaxation of the vehicle flow based formulation (5.1) - (5.8), even though this second formulation contains many more constraints (the subtour breaking constraints). In the modified formulation, $y_{ij}^k \leq nx_{ij}$ replaces $\sum_k y_{ij}^k \leq nx_{ij}$ for each i, j, and k and one unit must be shipped not only from the depot to each demand point, but from each demand point to the depot.

Since the effectiveness of integer programming decomposition techniques is intimately related (see [88]) to linear programming relaxations, these last results not only demonstrate the strength of different linear programming relaxations, but also of different Lagrangian relaxations. Indeed, whenever the solution to the Lagrangian relaxation of any integer programming problem can be obtained, for all values of the Lagrange multipliers, by solving *its* linear programming relaxation (this is the so called *integrality property* [52], then the Lagrangian dual problem is equivalent to the linear programming relaxation of the original problem. In these instances, the Lagrangian relaxation might be viewed simply as a mechanism for identifying new solution schemes other than the simplex method for the linear programming relaxation, but that exploit underlying problem structure. Since most of the relaxations that we have been considering satisfy the integrality property, in particular whenever the relaxation is either a minimal spanning tree problem or an assignment problem in the variables x_{ij}, these observations on linear programming relaxations show that the detailed commodity flow based formulation usually provides tighter Lagrangian relaxation generated lower bounds on C* than does the aggregate model. In a branch and bound setting, this characteristic of the aggregate model may mitigate its advantages. The decision as to which is preferred, stronger bounds leading to smaller branch and bound enumeration trees or more readily solvable Lagrangian relaxations, is an issue whose resolution must await further, and most likely empirical, investigation.

3.2.2 Benders Decomposition

Benders decomposition is a resource directive procedure for decoupling interrelated decisions and exploiting structure in models that contain "complicating variables." The algorithm iteratively fixes values for the complicating variables, solves the relatively easy problem that remains, and then uses its solution to redefine values for the complicating variables and, therefore, to reinitiate the algorithm's iterative steps. In principle, the algorithm is very attractive for vehicle routing applications. In cluster-first route-second procedures, like those that arise in the set covering formulation (1), once the "complicating" cluster decisions have been made, the problem reduces to a relatively simple traveling salesman problem for each vehicle. This is especially true when a small number, say 20 - 30, of demand points are assigned to any vehicle as is typical in practice. Similarly, in commodity flow based approaches like the formulation (6.1) - (6.7), once the route choice variables x_{ij} have been set, the remaining network flow constraints become relatively easy to solve.

Unfortunately, as even the most ardent advocates of mathematical programming might acknowledge, the practicality of applying Benders method to vehicle routing applications remains questionable. Although the method results in conceptually attractive planning procedures that often reflect hierarchical decision making practices of operating transportation planners, studies have yet to demonstrate its computational viability. Nevertheless, because the algorithm is appealing in many ways, because it does identify connections between different problem formulations, and because it has suggested heuristic methods for vehicle routing applications, we will outline some of its uses.

<u>Commodity Flow Based Formulation</u>

We first show that, when applied to the commodity flow based formulation, Benders decomposition leads to a vehicle flow formulation like (5.1) - (5.8). Our discussion rests upon a slightly modified version of arguments used by

Gavish and Graves [51] in the context of the traveling salesman problem. Suppose that we have ignored the flow constraints (6.4) - (6.6) in the commodity flow based formulation and have solved the resulting assignment problem in the variables x_{ij} obtaining an optimal solution x^*_{ij}. If the constraints (6.4) - (6.6), which might be viewed as consistency checks in the vehicle assignment variables, have a feasible solution y^*_{ij}, then x^*_{ij} and y^*_{ij} solve the routing problem. On the other hand, (6.4) - (6.6) might not have a feasible solution in which case the solution x^*_{ij} defines (i) some subtour T on the nodes 1,2,...n, and/or (ii) a subtour T containing the depot, but whose total demand exceeds vehicle capacity K. In either instance, Farkas' Theorem of the alternative characterizes the inconsistency in the system (6.4) - (6.6). That is, the system has no solution when $x_{ij} = x^*_{ij}$ if and only if there are constants $\pi^k_0 \equiv 0$, $\pi^k_1,\ldots,\pi^k_n$, for all k, and nonnegative constants γ_{ij}, for all i and j, satisfying the inequalities:

$$\pi^k_i - \pi^k_j + \gamma_{ij} \geq 0 \qquad \text{for all } i,j,k \tag{9.1}$$

and

$$\sum_{k=1}^{n} \pi^k_k d_k + K \sum_{i=0}^{n} \sum_{j=0}^{n} \gamma_{ij} x^*_{ij} < 0. \tag{9.2}$$

We can define the constants in these inequalities as follows: let $T_1 = \{j \varepsilon T : j \neq 0\}$, let T_2 equal the nodes 0,1,...,n not in T_1, and define $\pi^k_i = -1$ for all $i \varepsilon T_1$ and $\pi^k_i = 0$ otherwise; define $\gamma_{ij} = 1$ whenever $i \varepsilon T_1$ and $j \varepsilon T_2$; $\gamma_{ij} = 0$ otherwise. Making these substitutions, we see that (9.1) is satisfied and that (9.2) becomes

$$-\sum_{k \varepsilon T_1} d_k + K \sum_{i \varepsilon T_1} \sum_{j \varepsilon T_2} x^*_{ij} < 0$$

or, with $D(T) \equiv \sum_{k\varepsilon T} d_k$ denoting the demand on subtour T,

$$\sum_{i\varepsilon T_1} \sum_{j\varepsilon T_2} x^*_{ij} < D(T)/K. \qquad (9.3)$$

Here we have defined $d_0 = 0$. Note that inequality (9.3) is satisfied by x^*_{ij} since either (i) $0 \notin T$ and the left-hand side is zero or (ii) $0 \varepsilon T$, $D(T) > K$, and the left-hand side is one. Any feasible solution x_{ij} and y_{ij} to the formulation (6.1) - (6.7) will violate (9.3); in fact

$$\sum_{i\varepsilon T_1} \sum_{j\varepsilon T_2} x_{ij} \geq \lceil D(T)/K \rceil, \qquad (9.4)$$

where $\lceil z \rceil$ denotes least integer no smaller than z, since the right-hand side is the smallest number of vehicles with capacity K that must be routed through the demand points on T and subsequently on to either other demand points or the depot. When combined with the assignment constraints (6.1) - (6.2), the inequality (9.4) is equivalent to

$$\sum_{i\varepsilon T_1} \sum_{j\varepsilon T_1} x_{ij} \leq |T_1| - \lceil D(T)/K \rceil. \qquad (9.5)$$

In summary, the application of Benders decomposition to the commodity flow based formulation is conceptually quite simple. It first solves the assignment problem in variables x_{ij} and then generates additional constraints (9.4), or equivalently (9.5), that are added to modify the assignment formulation. The modified assignment problem is solved and the procedure is repeated with more constraints of the form (9.5) being added until the solution to the modified assignment problem satisfies the vehicle capacity constraints and route integrity (every subtour contains the depot) conditions.

Note that when the vehicle fleet is uncapacitated, that is $K \geq \sum_{k=1}^{n} d_k$, constraints (9.5) can only arise from the route integrity conditions. That is, $0 \notin T$ so that $|T_1| = |T|$ and $D(T) \leq K$ so that $\lceil D(T)/K \rceil = 1$. Consequently, the added constraints (9.5) are exactly subtour breaking constraints of the form (5.8) in terms of the aggregate variables $x_{ij} = \sum_v x_{ij}^v$ of the vehicle flow formulation. This observation shows that Benders decomposition is a cutting plane technique for the vehicle flow based formulation (5.1) - (5.8) that starts with the assignment problem as a relaxation and adds subtour breaking constraints on an as needed basis. When vehicles are capacitated, Benders decomposition is also adding capacity generated subtour breaking cuts (9.4) instead of the more compact constraints (5.5). In partial compensation, we can now write the vehicle flow constraints more compactly with aggregate variables x_{ij} rather than the detailed vehicle flow variables x_{ij}^v. On the other hand, if we initiate the algorithm with constraints (5.1) - (5.6) and use constraints (6.4) - (6.6) with $K \geq \sum_{k=1}^{n} d_k$ to insure route integrity, then Benders decomposition again will be generating subtour breaking constraints. That is, the vehicle flow based formulation can be viewed as a manifestation of a particular algorithmic strategy, Benders decomposition, when applied to the commodity flow based formulation.

Set Covering Formulation

As a prelude to applying Benders decomposition to the set covering formulation, let us first slightly revise the formulation. We assume that the vehicle fleet is nonhomogeneous and that the model is written with constraints (1), (3) and (4). We first multiply the constraints (4), i.e. $\sum_i a_{ij} d_i \leq K_v$, by z_j for $j \in \mathcal{C}_v$ and use (2), i.e. $\sum_{j \in \mathcal{C}_v} z_j = 1$, to give:

$$\sum_{i=1}^{n} \sum_{j\varepsilon\mathcal{C}_v} a_{ij} z_j d_i \leq K_v \sum_{j\varepsilon\mathcal{C}_v} z_j = K_v \tag{10}$$

Now define $y_{iv} = \sum_{j\varepsilon\mathcal{C}_v} a_{ij} z_j$; these variables specify whether or not demand point i is assigned to vehicle v, $y_{iv} = 1$, or not, $y_{iv} = 0$. With the y_{iv}'s substituted for the z_j's in (1) and (10), the model becomes:

$$\text{Minimize} \quad \sum_{v=1}^{NV} f(y_{1v}, y_{2v}, \ldots, y_{nv})$$

$$\text{subject to:} \quad \sum_{v=1}^{NV} y_{iv} = 1 \qquad (i = 1,2,\ldots,n) \tag{11.1}$$

$$\sum_{i=1}^{n} y_{iv} d_i \leq K_v \qquad (v = 1,2,\ldots,NV) \tag{11.2}$$

$$y_{iv} = 0 \text{ or } 1 \quad \text{for all i and v.}$$

In this formulation $f(y_{1v}, y_{2v}, \ldots, y_{nv}) = \sum_{j\varepsilon\mathcal{C}_v} \hat{c}_j z_j$ is the cost in the objective function of (1) for the routing of vehicle v. As before, this cost is determined by solving a traveling salesman problem defined on the demand points i assigned to v; that is, those demand points i with $y_{iv} = 1$. The precise mathematical formulation would be a modified version of the vehicle flow based model (5.1) - (5.8) or a commodity flow based model (6.1) - (6.8). For example, $f(y_{1v}, y_{2v}, \ldots, y_{nv})$ is the optimal objective value to (6.1) - (6.8) when $NV = 1$, $d_k = y_{kv}$, $K = n$ and y_{iv} replaces the right-hand side value of 1 in (6.1) and y_{jv} replaces the right-hand side value of 1 in (6.2).

As Fisher and Jaikumar [40], who proposed the use of this modeling approach, have noted, one very significant advantage to this formulation is that *any* feasible solution to (11) prescribes an assignment of demand points to vehicles consistent with the fleet's capacity limitations. Many of the most popular vehicle routing routines currently used in practice, and particularly those based upon the Clarke-Wright savings approach [23], do not provide this guarantee.

Moreover, this formulation highlights the fact that the vehicle routing problem can be viewed as a composite of two standard, and well-studied, problems in combinatorial optimization - the traveling salesman problem and a generalized assignment problem (11).

To apply Benders decomposition to this formulation, Fisher and Jaikumar suggest starting with a feasible solution y^*_{iv} to (11) and then solving a traveling salesman problem for each vehicle to compute each $f(y^*_{1v},y^*_{2v},\ldots,y^*_{nv})$. They solve each traveling salesman problem by using linear programming and a cutting plane technique so that the optimal dual variables to the linear programs define a subgradient (that is, a linear and lower bound) approximation to $f(y_{1v},y_{2v},\ldots,y_{nv})$ that equals f at $y_{iv} = y^*_{iv}$. Next they solve (11) with the subgradient approximation in place of f. Repeating the procedure with the optimal solution to this problem defining new values for y^*_{iv}, they obtain a further subgradient approximation to f. They then continue to iterate between (11), with all of the previously generated subgradients used to construct a piecewise linear approximation to f, and the traveling salesman problems until the optimal solution, or a near optimum, has been identified. For more details, we refer the reader to the original paper ([40]).

Recently, Federgruen and Zipkin [38] have extended this approach for problems with stochastic demand. In this setting, decision making is complicated by the choices to be made as to how much w_i of the good is to be delivered to every demand point. In addition, the costs now include an inventory carrying and storage cost $q_i(w_i)$ at each delivery point i, which is a strictly convex function defined by the stochastic demand pattern. The model is much the same as before except that now the variable w_i replaces the constant demand d_i in the constraint (11.2) and the decision variables w_i are further constrained by an allocation constraint $\sum_{i=1}^{n} w_i \leq A$ where A is the total amount of the good that is available at a central depot. For this problem, for any fixed values of the demand point to vehicle assignment variables y_{iv}, the model decomposes into a traveling salesman problem for each vehicle and an inventory allocation problem in the variables w_i. The solution to each of these subproblems defines a subgradient approximation to the inventory allocation costs as a function of the variables y_{iv} as well as to the routing cost $f(y_{1v}, y_{2v}, \ldots, y_{nv})$. Otherwise the algorithm is conceptually similar to that proposed by Fisher and Jaikumar for applications with a fixed demand pattern.

3.2.3 Optimization-Based Heuristics

The various problem formulations of the vehicle routing problem suggest not only integer programming problem decompositions, but also heuristic methods based upon embedded optimization procedures. In this section, we show how the set covering formulation, and its reformulation as a generalized assignment/traveling salesman model, leads to new and powerful heuristics.

We first consider a hueristic due to Fisher and Jaikumar [41]. For simplicity, we will assume that the costs c_{ij} are given by the Euclidean distances between points i and j. As we have noted previously, the difficulty in solving the generalized assignment formulation (11) lies in the highly nonlinear and complex nature of the objective function which requires for each assignment of the demand points to vehicles the solution of a traveling salesman problem for each vehicle. Any method for approximating this function would lead to a heuristic solution procedure. Fisher and Jaikumar suggest a linear approximation $\hat{f}(y_{1v},y_{2v},\ldots,y_{nv}) = \sum_{i=1}^{n} \sum_{v=1}^{NV} q_{iv}y_{iv}$ determined as follows. First, select the coordinates of one "cluster point" in the plane to represent each vehicle; then the surrogate distance q_{iv} is the added cost (or distance) caused by inserting demand point i on the route from the depot to cluster point v; that is, the cost of a route with the cluster point v, point i, and the depot minus the cost of the route without demand point i. Once the values of q_{iv} have been defined in this way, problem (11) with $\hat{f}$ in place of f is a linear generalized assignment problem. After solving this problem and assigning the demand points to vehicles, the authors then solve, by a cutting plane algorithm, a traveling salesman problem for each vehicle to evaluate $f(y_{1v},y_{2v},\ldots,y_{nv})$ and define the vehicles's delivery sequence through its demand points.

Fisher and Jaikumar describe several methods for determining the location of the cluster points. For example, experienced dispatchers might make these selections. One automated procedure that has worked well in their experimentation is to first partition the plane into NV conical regions about the depot, each with approximately the same total demand requirements. The cluster point for each cone is placed along the ray from the depot bisecting that cone and at a radial distance from the depot equal to that of some demand point so that (approximately) 25% of all that cone's total demand is farther than the cluster point from the depot. For further details on this procedure and other methods for locating the cluster points, we refer the reader to the original paper ([41]).

Cullen, Jarvis and Ratliff [28] have proposed and tested a two-pronged heuristic combining a procedure similar to that just described with an approximate solution procedure for solving the set covering problem. The first phase solves an approximation to (11) with the cluster points chosen by a dispatcher or an analyst and with q_{iv} set equal to the Euclidean distance between demand point i and cluster point v. After solving the generalized assignment problem†, though, the authors then redefine the location of each cluster point v so as to minimize the total distance from this point to all of the demand points assigned to it (this is convex optimization problem for each v). They then iterate between the assignment and location problem until no further improvements are possible.

The second phase of their procedure uses the solution to the first phase together with any other tentative solution generated previously, possibly by the dispatcher, to define columns for the set partitioning problem (1). Rather than generate all possible columns, they start with this small subset. This problem is solved approximately as follows. Starting with any feasible solution S_0, for example that generated from phase 1, we will construct a new and, hopefully, improved solution S one route at a time. First subdivide the routing cost $\hat{c}_i$ for any vehicle j among its demand points in S_0, defining a "price," or revenue, p_i for serving each demand point i. These prices are analogous to the simplex multipliers of linear programming. One choice for the prices that has worked well is to apportion the cost $\hat{c}_j$ to the demand points i assigned to vehicle j in proportion to the cost of servicing the points individually on a one-drop route. Next compute the "relative profit" $p_k = \sum_{i=1}^{n} p_i a_{ik} - \hat{c}_k$ for every column of k of the set partitioning problem. If all of these relative profits are nonpositive, then the current solution is optimal (in fact, in the linear programming relaxation as well). Otherwise, find the column with largest relative profit and use it as the route of the first vehicle in a new solution. Then select the column with the next best relative

---

†The authors assumed that all demands d_i were equal to 1 so that the generalized assignment problem reduces to a usual assignment problem. We will cast our discussion in the broader context with (possibly) unequal demands.

profit and add it to the new solution and so forth. If any new route R generated in this way contains a demand point already assigned to a vehicle in the new solution S being generated, then delete the demand point from R. When every demand point has been assigned to a route in S, we set S_0 equal to S and iterate on the entire procedure continuing until S does not improve upon S_0. Although this procedure is not guaranteed to generate a better solution at each step, even if one exists, it has proved to be effective in generating near optimal solutions.

At this point, we could terminate the algorithm with the last solution generated from the set covering problem. Or, we could repeat the entire procedure, using phase 1 to generate a new column for the set partitioning problem. At the second and every subsequent iteration of the overall process, the linear approximation $f = \sum_{iv}\sum q_{iv} y_{iv}$ would be modified by subtracting $\sum_{i=1}^{n} \sum_{v=1}^{NV} p_i y_{iv}$ where $p_1, p_2, \ldots p_n$ are the latest prices determined by the set partitioning algorithm. In this way, any solution to the phase 1 problem will have a nonnegative relative profit with respect to the surrogate distances and, hence, be an attractive column to add to the set partitioning problem (considering surrogate distances). Fisher and Jaikumar [43] have recently found that other approaches to solving the set covering problem can also be effective. Cullen, Jarvis and Ratliff [28] have also discovered that interactive heuristics stimulated by graphical display of tentative solutions can further enhance this solution approach.

Both of these assignment/generalized-assignment heuristics have proved to be very successful in solving a wide range of test problems with up to as many as 200 demand points. These methods are competitive in running time with most standard heuristics that do not incorporate embedded optimization procedures, e.g. the Clarke-Wright savings method, and yet provide better cost solutions in almost all instances.

4. OPTIMIZATION AND VEHICLE FLEET PLANNING

Although vehicle routing problems are central to vehicle fleet planning, they are not entirely representative. Certain simplifications and apparently minor variations of the routing models considered in the last section arise frequently in practice and are much better suited for optimization. In this section, we describe a variety of these applications as well as other uses of exact methods for vehicle routing and scheduling. To help focus the discussion, we classify contributions according to the type of optimization methods employed, namely combinatorial methods, network flow techniques, branch and bound methods, dynamic programming, and integer programming decomposition. Rather than attempting to be exhaustive, we describe representative applications from each of these categories.

Generally, three types of modifications to the generic vehicle routing problem lead to more easily solved planning problems. These modifications (i) permit the vehicles to circulate among their service destinations without returning to a fixed depot(s); (ii) permit the vehicles to visit only the most profitable of the demand points; and (iii) impose special network structure that attenuates the combinatorial explosiveness of route selection. The first of these modifications arises in many airline or railway systems where traffic originates and departs from each stop and every demand point can function economically on a day to day basis as a depot for refueling, basic maintenance and the like. The second modification arises in longer range planning efforts when the firm providing the transportation/distribution services has the flexibility to choose its markets. The third simplification is common in many main haul and commuter train systems where traffic is restricted to straight line networks or hub networks (straight lines emanating from a common central depot). Special network structure also arises when certain types of subordinate decisions such as time table setting are made subsequent to earlier route selection and sequencing decisions that define a network route structure for each vehicle.

Another special network structure, the space-time diagram, emerges from scheduling considerations. These networks distinguish nodes by both geographical locations and points in time. Service arcs connect two locations whenever a potential or required service connects the locations at the appropriate times and lay-over arcs connect the same location at different points in time.

4.1 Combinatorial Methods

Matching theory, a cornerstone of combinatorial optimization, is intrinsic to many vehicle planning models. The assignment relaxation (see section 3.2.1) of the vehicle routing problem is but one, albeit very important, application. The following scenario is illustrative of other applications. Government safety regulations frequently limit the size of vehicles delivering potentially hazardous materials as when they restrict tanker trucks to carrying no more gasoline than is necessary to satisfy the demand of two service stations. If each truck visits exactly two stations, then every route pairs, or matches, two demand points i and j at a cost c_{ij} equal to the total routing cost of a truck traveling from the depot through stations i and j and back to the depot. As a consequence, this *2-delivery problem* becomes an easily solved ([78]) nonbipartite matching model. This problem also models situations in which each truck route can service either one or two stations - conceptually we add a duplicate copy of the depot for each demand point with arcs of zero cost connecting the copies to each other, and solve the resulting 2-delivery problem. It is interesting to note that the 2-delivery problem was the genesis of Dantzig and Ramser's [32] original study of the vehicle routing problem.

Another application of nonbipartite matching is the *Chinese postman problem*† in which a single vehicle (e.g. postman or fuel oil delivery truck) is to cover every arc of a distribution network, with backtracking if necessary, in order to minimize total routing costs. The solution to this problem, which is a hallmark in the application of combinatorial methods to vehicle routing problems, combines shortest path and matching computations as follows ([35]). First, identify and compute the shortest path distance between every pair of odd-degree nodes in the distribution network. Using these shortest path distances as surrogate arc costs, find the minimum cost matching joining the odd degree nodes. Next, define a new network with the arcs in the original distribution network together with duplicate copies of those in the shortest paths corresponding to arcs chosen in the minimum matching. The new network contains an Eulerian circuit, since every node has an even degree. That is, it has a circuit in which every arc, including any duplicate, is used exactly once. This circuit solves the postman problem ([35], [79]). This example might suggest that arc routing models are easier to solve than their node routing counterparts. Unfortunately, in most instances both versions of routing problems are NP-complete ([81]).

One other arc routing model that is easy to solve is the directed Chinese postman problem in which every arc must be traversed in a given direction. This model is a simple network flow problem with a lower bound of one unit of flow on every arc.

Combinatorial methods also arise in specialized problem contexts. For example, Assad [5] has studied a single track or corridor rail network with

---

†This name refers to a Chinese researcher, Mei-ko Kwan, who first identified the problem.

demand (measured, say, in rail cars) generated at the first station on the network and destined for stations further "down the line." He shows that a *greedy* or *insertion* algorithm solves the problem of finding the best stop sequences for two trains in order to minimize the total car-delay; that is, the sum over all cars of the number of intermediate stops a car makes prior to its destination. The heuristic starts with train 1 servicing all stations and then re-assigns stations one at a time from train 1 to train 2 until no further improvement is made, at each step choosing the station that minimizes total delay with those stations previously re-assigned. He also shows that this algorithm solves the problem if all destinations have the same demand and per car delay varies from station to station, but need not be optimal when both the traffic demands and station delays vary. In the later case, he proposes a branch and bound procedure. Lakshminarayan et. al. [77] study a similar model formulated from a machine scheduling problem. Denardo, Huberman and Rothblum [34] establish an interleaving property for bus stop location problems, and other applications, that is related to the insertion property. They wish to locate bus stops in order to minimize the walking distance for user trips on a single-line bus route. Let x_j^n denote the location of the $j^{\underline{th}}$ stop in an optimal n-stop schedule. They show that the optimal stop schedules can be chosen for every $n = 1,2,\ldots$ so that an (n-1)-stop schedule is interleaved between the stops of an n-stop schedule; that is, $x_j^n \leq x_j^{n-1} \leq x_j^n$ for all $j = 1,2,\ldots,n-1$.

These studies of insertion and interleaving properties are intimately related mathematically. Each can be viewed as a combinatorial optimization problem involving the maximization of a submodular set-valued function. Facility location models ([26]) have the same underlying mathematical structure.

4.2 Network Flows and Linear Programming

Possibly the easiest of all vehicle fleet planning problems to solve are those that can be formulated as network flow models or as linear programs. Many of these problems focus on vehicle fleet assignment and vehicle scheduling issues and can be viewed as extensions and variations of the directed Chinese Postman problem formulated on space-time networks. In this setting, the "Postman's path(s)" are required to cover arcs corresponding to service legs of a vehicle's itinerary (e.g., fly from Boston at 9:00 a.m. to New York at 10:00 a.m.). Typically, the network representations of these space-time scheduling networks are acyclic. At times, though, the networks model repetitive planning periods and contain cycles formed by the arcs "returning" vehicles from the end of one planning period to the beginning of the next period.

One of the most basic versions of these models, which is central to much of vehicle scheduling, is the minimum fleet size problem: what is the fewest number of vehicles required to cover each of the service legs on an acyclic network? This problem is easily formulated and solved as a maximum (or, minimum) flow problem ([30]) and, by virtue of the max flow - min cut theorem, is equivalent to finding the maximum number of service legs no two of which can be covered on the same vehicle route. Since this property is but one manifestation of a celebrated theorem due to Dilworth, the minimum fleet size problem is often referred to as a Dilworth scheduling problem.

A number of airlines and aircraft manufacturers use the planning cycle version of the scheduling model for vehicle fleet management and for planning of fleet acquisition. They associate revenue with the service legs (which need not be covered), associate costs with the lay-over legs, and choose the flight schedule that maximizes profit. Simpson [116] describes a series of network flow and linear programming problems that are enhancements of this model. Similar models arise in other contexts as well. For example, Arisawa and Elmaghraby [2] formulate a network flow model to minimize expected operating costs for a 2-period model of hub transportation systems with uncertain demand in the second period.

Each of these finite time horizon models has infinite dimensional analogues in which the length of the planning or scheduling period is not fixed a priori, but is determined by a choice of the best periodic schedule. One example, the "tramp-steamer" problem, requires a periodic route for a single vehicle (i.e., a cycle through the scheduling network) that maximizes profit per unit time. This problem can be formulated as a linear network flow model with a single additional constraint ([30]). Or, it can be cast as a network flow model with a linear fractional objective function and can be solved efficiently as a sequence of linear network flow models ([79], [92]). A more general version of this problem is obtained by imposing lower and upper bounds on arc flows. Generally, any solution satisfying these conditions, and particularly the desired solution minimizing average costs over an infinite planning horizon, requires multiple vehicles. Orlin [101] shows that a solution can be found by first solving a finite node network flow model with one additional side constraint, as in the tramp-steamer problem, and then rounding its (fractional) solution appropriately.

One specialization of the last model is a periodic version of the Dilworth scheduling problem. The problem is to find the fewest number of vehicles required to cover a periodic schedule of service legs (e.g., fixed daily and Sunday services repeating each week) over an infinite planning horizon. If deadheading is forbidden, minimizing the number of vehicles is equivalent to minimizing vehicle idle time at the service locations and a FIFO scheduling policy is optimal ([9], [10]). If deadheading is permitted, the problem becomes more complicated, yet still can be cast as a minimum network flow problem on a finite network ([100]). See [54] for a different perspective on this problem related to the minimization of vehicle idle time as in the no deadheading case.

Another class of vehicle fleet management problems that can be formulated as network flow models arises from imbalances inherent in empty rail car distribution. How are empty freight cars (locomotives, cabooses) accumulating at rail yards near heavy demand centers to be transported back to those yards that serve as origin points for traffic entering the rail system? A number of papers ([61], [69], [80], [91], [121], [122], [126]) treat various aspects of this problem. Most of these papers formulate the problem as a standard minimum cost network flow model or as multi-commodity network flow problems.

Brown and Graves [18] describe quite a different use of network flow models. They consider the routing of petroleum tank trucks from bulk terminals for an application in which each route contains a single drop point. The problem is to minimize transportation costs while scheduling each truck within an available shift interval. Although the problem is an integer program, the authors' experience indicates that network flow models provide good approximate solutions.

4.3 Branch and Bound

Although branch and bound methods have generally been ineffective in solving vehicle fleet planning problems, they have been applied successfully in limited situations. Baker [6] has used branch and bound codes equipped with special purpose branching logic to solve a variety of tanker ship scheduling models, addressing such issues as tanker routing, tanker loading, and allocation of tankers to terminals. The traveling salesman problem and multiple traveling salesman problem are major components of these models, though the models contain other types of "complicating" constraints as well. He has also solved oil rig scheduling applications that are quite similar to vehicle routing problems (identify rigs with vehicles and platforms with demand points).

Several studies of rail system planning have relied on branch and bound techniques. Achermann [1] has applied the algorithm to plan train formulation, rail car grouping and blocking strategies, and train routing for rail freight systems. He describes an application to a 20 yard Swiss Railway problem. Bodin et. al. [17] proposed a more elaborate model developed for the Norfork and Western Railroad and discuss the application of an interactive branch and bound algorithm for a 33 yard problem. Salzborn [111] describes an application to suburban rail systems designed as a hub network. He has considered the construction of train stop-schedules and, particularly, the issue of when a train emanating from the hub should return there rather than continue on to the end of the line. Szpigel [120] has studied the problem of arranging for meets and passes on a single track rail line. His computational experience was very limited, though. The largest problem that he solved by branch and bound involved 5 truck sections and 10 trains.

Mairs et. al. [89] have applied branch and bound to a combined production allocation and distribution model developed for Frito-Lay. The model is very large. It has approximately 2000 0-1 variables and was solved in from 40 minutes to 2 hours on an IBM 370/168. One lesson emerged from this study. Different formulations of the problem led to large variations in running time, even for the linear programming relaxation of the model. In this case, a more disaggregate formulation led to better algorithmic performance. This type of behavior is not uncommon in integer programming (see [53]), though only very recently has any theoretical foundation ([88]) helped to explain it.

4.4 Dynamic Programming

Typically, when contrasted with other optimization methods and their use within the broad hierarchy of vehicle fleet planning, dynamic programming is most useful for operational decision making, rather than higher level tactical and strategic planning. In particular, dynamic programming algorithms are often applied to problems formulated on a single track or on a hub network, which are often defined by routes determined by higher level vehicle fleet assignment decisions. These simplified network structures require smaller state space descriptions than do more elaborate networks and, therefore, their use helps to ameliorate the "curse of dimensionality" inherent to dynamic programming.

Two examples are the trade-offs between local and express services for commuter trains/operating on a straight line network or time table scheduling in a straight line network environment. Nemhauser [97] has developed an efficient dynamic programming algorithm for the first of these problems. His objective was to schedule services in order to maximize revenue while accounting for lost sales due to excessive wait times. Morlok et. al. [95] have studied the time table scheduling problem for suburban railway systems. The state variables in their dynamic programming algorithm correspond to discrete train arrival times. Salzborn [111] and Young [124] have considered similar applications.

In the context of hub networks, Saha [110] has used dynamic programming to determine loading patterms of capacitated trains at stops in order to maximize the total number of passengers served by the rail system. Vuchic and Newell [123] have studied the design of station locations to minimize total passenger travel time. They show that a dynamic programming version of the problem reduces to solving a set of simultaneous linear equations.

Quite likely, dynamic programming will continue to be an effective planning tool for similar problems arising in specialized networks. This type of algorithm might prove to be useful in other settings as well. For example, Psaraftis [106] has developed a dynamic programming algorithm to establish schedules for a single vehicle in small dial-a-ride transit systems. He feels that the algorithm provides a useful benchmark against which to test other planning methods, such as heuristics. Psaraftis and Tharakan [107] have extended this approach to multi-vehicle versions of the problem, using the single vehicle solution procedure as a subroutine.

4.5 Integer Programming Decomposition

In section 3, we described several algorithms based upon price directive (Lagrangian relaxation) and resource directive (Benders algorithm) decomposition. Although these algorithms appear to be promising, their development is still in its infancy. Much testing and further investigation remains to be done.

In only a few cases has integer programming decomposition been applied to realistic vehicle fleet planning problems. As mentioned earlier, Geoffrion and Graves [53] have devised an implementation of Benders decomposition for a combined production and distribution system. The success of this study has been a major stimulant in renewing interest in resource directive decomposition for integer programming models. Richardson [108] has applied Benders decompsition to an aircraft routing problem and Florian, Guerin and Bushel [44] have applied the algorithm to the scheduling of engine distribution in rail systems. The last study resulted in mixed success. The algorithm worked well on a 718 node, 986 arc space-time diagram representing a region of the Canadian National Railways, but was much less successful when applied to a larger 2000 arc problem.

Even though Lagrangian methods like those that we have considered earlier might prove to be computationally superior to resource directive techniques, resource decomposition might become increasingly attractive on other grounds. For example, both of these algorithmic strategies could become useful ingredients of heuristic methods. For example, Sexton [113] has recently demonstrated the success of heuristics based upon Benders decomposition for certain vehicle scheduling and routing problems.

5. PROSPECTS

As we have noted in sections 3 and 4, certain types of vehicle fleet planning problems are well-suited for optimization (generally, those that can be formulated as "nicely-structured" network flow or matching type combinatorial models), whereas others, particularly those involving vehicle routing, are considerably more difficult to solve exactly. For the second category, heuristics have proved to be an attractive alternative to exact methods; generally, heuristics (i) are easy to understand and, consequently, are more readily accepted by managers, (ii) are less sophisticated algorithmically and, therefore, are easier to program and maintain for computerized planning, and (iii) are effective in solving a wide range of practical problems and provide solutions that are usually accepted as "good" or "reasonable." These compelling arguments on behalf of heuristics suggest that they will continue to be instrumental in the analysis of vehicle fleet planning problems. Nevertheless, there are reasons to prefer exact solutions and to expect an increasing reliance on exact methods for vehicle fleet planning. The following brief list delineates some of these reasons. In many instances the arguments apply to optimization, in general, and not just to vehicle fleet planning.

1) *Cost-Benefits.* In some situations, such as the scheduling of large tankers, the expense of implementing and utilizing costly exact methods with a guarantee of optimality is more than justified by the large expenditures involved. As computer costs continue to plummet and vehicle operating costs continue to rise, the distinction between these applications and others will begin to blur and optimization should become increasingly more viable economically.

2) *Performance Guarantees and Sensitivity Analysis.* Frequently, model validation, policy studies, and other uses of models for furnishing problem insight require knowledge about the sensitivity of a model to varying input parameters. Optimization, and re-optimization, is perfectly tailored for this purpose. Heuristics, even if supported by error bound analysis, are much less useful. For example, suppose that we solve a problem and obtain a solution guaranteed to be within ten percent of optimality, change a parameter, and then resolve to obtain another solution with the same performance guarantee. How do we evaluate the sensitivity of the solution to this parameter change? That is, how do we discern whether any change in the approximate solution is due to the parameter change or to some peculiarity of the heuristic being employed.

3) *Exact Models and Exact Algorithms.* Common wisdom states: "since any model and its accompanying data merely approximates reality, can the effort required to find an optimal solution, as opposed to a more easily obtained approximate solution, ever be justified?" As tempting as it is to accept this argument, it should be applied with caution. Suppose that the solution to a model *approximates* the solution to a real problem, and that a heuristic algorithm solves *the model* approximately. Isn't it conceivable that the modeling approximation and solution approximation are compounding? Though exact solutions might be difficult to obtain, they still provide the best approximation to the real problem obtainable within the confines of the model. Exact methods permit the analyst to focus on the quality of the model per se, without having to account for secondary approximations of the model itself.

4) *Improved Technology.* New solution techniques in integer programming such as Lagrangian relaxation and resource directive decomposition enhance the prospects for successful application of optimization methodologies for vehicle fleet planning problems. These technological advances combined with improvements in computer technology should foster greater use of exact methods in the future (though, undoubtedly, some optimization problems will always challenge computer capabilities and will always be limited by their computer expense).

5) *Optimization-Based Heuristics.* As we have seen in section 3.2.3, optimization methods can be used to guide the design of heuristic procedures and to improve upon their performance. As another recent illustration, Ball et. al. [8] demonstrates the advantages of using matching algorithms as an ingredient of a heuristic for vehicle scheduling. The prospects for this type of analysis seem far from fully realized. Indeed, it is conceivable that the master-slave relationship between heuristics and optimization might be most fruitful when inverted. That is, it is possible that further study might show that heuristics could be used much more productively than they are at present as components of optimization procedures.

6) *The Modeling-Algorithm Inteface.* As we have noted several times earlier, the mere way in which a model is formulated can have a pronounced effect upon how it is solved. This type of modeling-algorithm interface is a well-known characteristic of integer programming and, yet, seems not to be fully understood. For example, despite the enormous previous effort devoted to the traveling salesman problem, Jonker et. al. [70] have recently shown that a new formulation for this problem leads to sharper Lagrangian bounds and to significant improvements in algorithmic performance of Lagrangian methods. Is there some methodology for deciding when a formulation is a

good one, algorithmically, and, possibly, for stimulating new and improved formulations for a given problem? Magnanti and Wong [87] have taken some steps in this direction by studying model formulation and its effect on Benders decomposition. Further studies of this nature might lead to new and better understanding of integer programming models and to improved performance of exact methods.

Although these arguments do not fully neutralize the advantages of heuristic methods, they do suggest that exact methods might become an increasingly important factor in vehicle fleet planning. In particular, as our discussion has demonstrated, for the first time new advances in integer programming, like Lagrangian relaxation, that have been very successful in other application areas of combinatorial optimization are now being applied to vehicle routing and other vehicle fleet planning problems. As researchers become more attracted to this type of analysis, motivated in part by the early successes, it is likely that new innovations will surface and that branch and bound and other optimization methods will evolve significantly. In fact, it would not be surprising to find that the gap between algorithmic capabilities for the traveling salesman problem and for other vehicle fleet planning problems will narrow, and that exact methods or optimizaton-based heuristics might progress to the point where a variety of vehicle fleet planning problems with up to 100 or more demand points would be solved routinely.

Of course, our discussion in this paper does not exhaust all possibilities for improving vehicle fleet planning. For example, continuous approximation methods (see [98]) or probabilistic and asymptotic analysis (e.g., Stein's studies [117], [118] of the dial-a-ride problem) might prove to be useful in solving very large vehicle routing and scheduling problems. Or, geometric type methods might prove to be useful alternatives to the algorithmically oriented combinatorial optimization methods that we have considered here.

As Pollack [105] has observed, the challenges for richer and more realistic modeling of vehicle fleet planning problems are enormous. So too, then, are the challenges for solving these problems.

ACKNOWLEDGEMENTS

I am grateful to Bruce Golden, Steve Graves, Jan Hammond, and Paul Mireault for their comments on an earlier version of this paper.

REFERENCES

[1] J. ACHERMANN, "Model and Calculation Program Relating to the Optimum Formulation of Trunk Haul Freight Trains," *Bulletin of the International Railway Congress Association, Cybernetics and Electronics on the Railways,* 6, 181-226 (1969).

[2] S, ARISAWA AND S. ELMAGHRABY, "The 'Hub' and 'Wheel' Scheduling Problems, Parts I and II," *Transportation Science,* 2, 124-165 (1977).

[3] A. ASSAD, "Analytic Models in Rail Transportation: An Annotated Bibliography," Working Paper OR 066-77, Operations Research Center, MIT (Oct. 1977).

[4] A.A. ASSAD, "Models For Rail Transportation," *Transportation Research,* 14A, 205-220 (1980).

[5] A. ASSAD, "Modeling Rail Freight Management," Ph.D. Thesis, Sloan School of Management, MIT (September 1978).

[6] T.E. BAKER, "Mixed Integer Programming At Exxon," Seminar, Operations Research Center, MIT (Nov. 1979).

[7] M.L. BALINSKI and R.E. QUANDT, "On an Integer Program for a Delivery Problem," *Operations Research,* 12, 300-304 (1964).

[8] M. BALL, L. BODIN AND R. DIAL, "Experimentation With a Computerized System For Scheduling Mass Transit Vehicles and Crew," presented at the International Workshop on Urban Passenger, Vehicle and Crew Scheduling, The University of Leeds, England (July 1980).

[9] T.E. BARTLETT, "An Algorithm for the Minimum Number of Transport Units to Maintain a Fixed Schedule," *Naval Res. Log. Quart.,* 4, 139-149 (1957).

[10] T.E. BARTLETT AND A. CHARNES, "Cyclic Scheduling and Combinatorial Topology: Assignment of Routing and Motive Power to Meet Scheduling and Maintenance Requirements: Part II: Generalizations and Analysis," *Naval Res. Log. Quart.,* 4. 207-220 (1957).

[11] M. BAZARAA AND J. GOODE, "The Traveling Salesman: A Duality Approach," *Mathematical Programming,* 13, 221-237 (1977).

[12] J. BEARDWOOD, H. HALTON AND J.M. HAMMERSLEY, "The Shortest Path Through Many Points," *Proc. Cambridge Phil. Society,* (Math. and Phys. Sci.), 55, 299-327 (1959).

[13] M. BELLMORE AND G.L. NEMHAUSER, "The Traveling Salesman Problem: A Survey," *Operations Research,* 16, 538-558 (1968).

[14] E. BELTRAMI AND L. BODIN, "Network and Vehicle Routing for Municipal Waste Collection," *Networks,* 4, 65-94 (1974).

[15] L. BODIN AND L. BERMAN, "Routing and Scheduling of School Buses by Computer," *Transportation Science*, 13, 113-129 (1979).

[16] L. BODIN AND B. GOLDEN, "Classification in Vehicle Routing and Scheduling," *Networks* (to appear).

[17] L. BODIN, B. GOLDEN, A. SCHUSTER AND W. ROMIG, "A Model For the Blocking of Trains," *Transportation Research*, 14B, 115-120 (1980).

[18] G.G. BROWN AND G.W. GRAVES, "Real Time Dispatch of Petroluem Tank Trucks," *Management Science* (to appear).

[19] N. CHRISTOFIDES, "The Vehicle Routing Problem," *R.A.I.R.O. Recherche Operationelle*, 10, 55-70 (1976).

[20] N. CHRISTOFIDES, "Worst-Case Analysis of a New Heuristic for the Traveling Salesman Problem," Report 388, Graduate School of Industrial Administration, Carnegie Mellon University (Feb. 1976).

[21] N. CHRISTOFIDES AND S. EILON, "An Algorithm For the Vehicle Dispatching Problem," *Operational Research Quarterly*, 20, 309-318 (1969).

[22] N. CHRISTOFIDES, A. MINGOZZI AND P. TOTH, "Exact Algorithms for the Vehicle Routing Problem," *Mathematical Programming* (to appear).

[23] G. CLARKE AND J. WRIGHT, "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points," *Operations Research*, 12, 568-581 (1964).

[24] R.W. CONWAY, W.L. MAXWELL AND L.W. MILLER, *Theory of Scheduling*, Addison-Wesley, Reading, MA (1967).

[25] S.A. COOK, "The Complexity of Theorem-Proving Procedures," *Proc. 3rd Annual ACM Symp. on Theory of Computing*, Association for Computing Machinery, New York, 151-158 (1971).

[26] G. CORNUEJOLS, M.L. FISHER AND G.L. NEMHAUSER, "Location of Bank Accounts to Optimize Float: An Analytic Study of Exact and Approximate Algorithms," *Management Science*, 23, 789-810 (1977).

[27] H. CROWDER AND M. PADBERG, "Solving Large-Scale Symmetric Traveling Salesman Problems to Optimality," *Management Science*, 26, 495-509 (1980).

[28] F. CULLEN, J. JARVIS AND H.D. RATLIFF, "Set Partitioning Based Heuristics for Interactive Routing," *Networks* (to appear).

[29] G.B. DANTZIG, W. BLATTNER AND M.R. RAO, "Finding a Cycle in a Graph with Minimum Cost to Times Ratio with Applications to a Ship Routing Problem," in *Theory of Graphs* (P. Rosenstiehl, ed.), Dunrod, Paris, and Gordon and Beach, New York, 77-84 (1967).

[30] G.B. DANTZIG AND D.R. FULKERSON, "Minimizing the Number of Tankers to Meet a Fixed Schedule," *Naval Res. Log. Quart.*, 1, 217-222 (1954).

[31] G.B. DANTZIG, D.R. FULKERSON AND S.M. JOHNSON, "Solution of a Large Scale Traveling Salesman Problem," *Operations Research,* 2, 393-410 (1954).

[32] G.B. DANTZIG AND J.H. RAMSER, "The Truck Dispatching Problem," *Management Science,* 6, 80-91 (1959).

[33] P.S. DAVIS AND T.L. RAY, "A Branch-Bound Algorithm for the Capacitated Facilities Location Problem," *Naval Research Logistics Quarterly,* 16, 331-344 (1969).

[34] E.V. DENARDO, G. HUBERMAN AND U. ROTHBLUM, "Optimal Location on a Line are Interleaved," Technical Report, School of Organization and Management, Yale Univeristy (1980).

[35] J.EDMONDS AND E.L. JOHNSON, "Matching, Euler Tours, and the Chinese Postman," *Mathematical Programming,* 5, 88-124 (1973).

[36] S. EILON, C.D.T. WATSON-GANDY AND N. CHRISTOFIDES, *Distribution Management:* Mathematical Modeling and Practical Analysis," Hafner Publishing Co., New York (1971).

[37] M.A. EFFROYMSON AND T.L. RAY, "A Branch-Bound Algorithm for Plant Location," *Operations Research,* 14, 361-368 (1966).

[38] A. FEDERGRUEN AND P. ZIPKIN, "A Combined Vehicle Routing and Inventory Allocation Problem," Research Working Paper No. 345A, Graduate School of Business, Columbia University (June 1980).

[39] M.L. FISHER, "Lagrangian Relaxation For Combinatorial Optimization," Working Paper 78-10-06, Dept. of Decision Sciences, The Wharton School, University of Pennsylvania, June 1978 (Revised October 1978).

[40] M.L. FISHER AND R. JAIKUMAR, "A Decomposition Algorithm for Large-Scale Vehicle Routing," Technical Report 78-11-05, Dept. of Decision Sciences, The Wharton School, University of Pennsylvania (July 1978).

[41] M.L. FISHER AND R. JAIKUMAR, "A Generalized Assignment Heuristic For Vehicle Routing," *Networks* (to appear).

[42] M.L. FISHER AND R. JAIKUMAR, Private Communication (1980).

[43] M.L. FISHER AND J.R. SHAPIRO, "Constructive Duality in Integer Programming," *SIAM J. on Applied Math.,* 27, 31-52 (1974).

[44] M.G. FLORIAN, G. GUERIN AND G. BUSHEL, "The Engine Scheduling Problem in a Railway Network," *INFOR Journal,* 14, 121-138 (1976).

[45] L.R. FORD AND D.R. FULKERSON, *Flows in Networks,* Princeton University Press, Princeton, New Jersey (1962).

[46] B.A. FOSTER AND D.M. RYAN, "An Integer Programming Approach to the Vehicle Scheduling Problem," *Operation Research Quarterly,* 27, 367-384 (1976).

[47] B. FOX, "Data Structures and Computer Science Techniques in Operation Research," *Operations Research*, 26, 686-717 (1978).

[48] M.R. GAREY AND D.S. JOHNSON, *Computers and Intractibility: A Guide to the Theory of NP-Completeness*, Will Freeman and Company, San Francisco (1979).

[49] W.W. GARVIN, H.W. CRANDALL, J.B. JOHN AND R.A. SPELLMAN, "Applications of Vehicle Routing in the Oil Industry," *Management Science*, 3, 407-430 (1957).

[50] B. GAVISH, "Topological Design of Centralized Computer Networks-Formulations and Algorithms," *Networks* (to appear).

[51] B. GAVISH AND S.C. GRAVES, "The Traveling Salesman Problem and Related Problems," Working Paper OR 078-78, Operations Research Center, MIT (July 1978).

[52] A.M. GEOFFRION, "Lagrangian Relaxations and Its Uses in Linear Programming," *Math. Programming Study 2*, 82-114 (1974).

[53] A.M. GEOFFRION AND G.W. GRAVES, "Multicommodity Distribution System Design By Benders Decomposition," *Management Science*, 20, 822-844 (1974).

[54] I. GERTSBACH AND Y. GUREVICH, "Constructing an Optimal Fleet For a Transportation Schedule," *Transportation Science*, 11, 20-36 (1977).

[55] B. GILLETT AND L. MILLER, "A Heuristic Algorithm For the Vehicle Dispatching Problem," *Operations Research*, 22, 340-349 (1974).

[56] B.L. GOLDEN, "A Statistical Approach to the TSP," *Networks*, 7, 209-225 (1977).

[57] B.L. GOLDEN, "Evaluating a Sequential Vehicle Routing Algorithm," *AIIE Transactions*, 9, 204-208 (1977).

[58] B. GOLDEN, "Large Scale Vehicle Routing and Related Combinatorial Problems," Ph.D. Thesis, Operations Research Center, MIT (June 1976).

[59] B. GOLDEN AND T. MAGNANTI, "Deterministic Network Optimization - A Bibliography," *Networks*, 7, 149-183 (1977).

[60] B. GOLDEN, T. MAGNANTI AND H. NGUYEN, "Implementing Vehicle Routing Algorithms," *Networks*, 7, 113-148 (1977).

[61] S. GORENSTEIN, S. POLEY AND W.W. WHITE, "On the Scheduling of Railroad Freight Operations," Technical Report No. 320-2999, IBM Philadelphia Scientific Center (Jan. 1971).

[62] S.C. GRAVES AND T.L. MAGNANTI, "An Optimization-Based Approach to Vehicle Routing," ORSA Bulletin, 10, National Meeting, Colorado Springs (Nov. 1980).

[63] K. HANSEN AND J. KRARUP, "Improvements of the Held-Karp Algorithm for the Symmetric Traveling Salesman Problem," *Mathematical Programming*, 7 87-96 (1974).

[64] H. HARRISON, "A Planning System For Facilities and Resources in Distribution Networks," *Interfaces*, 9, 6-22 (1979).

[65] M. HELD AND R.M. KARP, "A Dynamic Programming Approach to Sequencing Problems," *SIAM*, 10, 196-210 (1962).

[66] M. HELD AND R. KARP, "The Traveling Salesman Problem and Minimum Spanning Trees," *Operations Research*, 18, 1138-1162 (1970).

[67] M. HELD AND R. KARP, "The Traveling Salesman Problem and Minimum Spanning Trees, Part II," *Mathematical Programming*, 1, 6-25 (1971).

[68] H. HERREN, "Computer Controlled Empty Wagon Distribution on The SBB," *Rail International*, 8, 25-32 (1977).

[69] J. HOLT, "Locomotive Scheduling By Computer <<Bashpeak>>," *Rail International*, 4, 1053-1058 (1973).

[70] R. JONKER, G. DE LEVE, J.A. VAN DER VELDE AND A. VOLGENANT, "Rounding Symmetric Traveling Salesman Problems with an Asymmetric Assignment Problem," *Operations Research*, 28, 623-627 (1980).

[71] R.M. KARP, "A Patching Algorithm For the Nonsymmetric Traveling-Salesman Problem," *SIAM J. Computing*, 8, 561-573 (1979).

[72] R. KARP, "Probabilistic Analysis of Partitioning Algorithms for the Traveling Salesman Problem in the Plane," *Mathematics of Operations Research*, 2, 209-244 (1977).

[73] R.M. KARP, "Reducibility Among Combinatorial Problems," in Complexity of Computer Computations, (R.E. Miller and J.W. Thatcher, eds.), Plenum Press, New York, 85-103 (1972).

[74] J.L. KENNINGTON AND R.V. HELGASON, *Algorithms for Network Programming*, John Wiley and Sons, New York (1980).

[75] V. KLEE, "Combinatorial Optimization: What Is the State of the Art?" *Mathematics of Operations Research*, 5, 1-26 (1980).

[76] A. KUEHN AND M. HAMBURGER, "A Heuristic Program For Locating Warehouses," *Management Science*, 9, 643-666 (1963).

[77] S. LAKSHMINARAYAN, R. LAKSHMANAD, R. PUPINEAU AND R. ROCHETTE, "Order Preserving Allocation of Jobs to Two Non-Identical Parallel Machines - A Solvable Case of the Maximum Cut Problem," Dept. d'ingeniere, Univ. du Québec à Trois-Rivières (Preprint).

[78] E.L. LAWLER, *Combinatorial Optimization: Networks and Matroids*, Holt, Rinehart and Winston, New York (1976).

[79] E.L. LAWLER, "Optimal Cycles in Graphs and the Minimal Cost-to-Time Ratio Problem," *Proceedings of a Conference on Periodic Optimization*, CISM, Udine, Italy (1973).

[80] C.D. LEDDON AND E. WRATHALL, "Scheduling Empty Freight Car Fleets on the Louisville and Nashville Railroad," *Proceedings of the Second International Symposium on the Use of Cybernetics on the Railroads*, 154-158 (1967).

[81] J.L. LENSTRA AND A.H.G. RINNOOY KAN, "Complexity of Vehicle Routing and Scheduling Problems," *Networks* (to appear).

[82] A. LEVIN, "Scheduling and Fleet Routing Models for Transportation Systems," *Transportation Science*, 5, 232-255 (1971).

[83] S. LIN, "Computer Solutions of the TSP," *Bell System Technical Journal*, 44, 2245-2269 (1975).

[84] J.D.C. LITTLE, K.G. MURTY, D.W. SWEENEY AND C. KAREL, "An Algorithm for the Traveling Salesman Problem," *Operations Research*, 11, 972-989 (1963).

[85] T.L. MAGNANTI, "Optimization For Sparse Systems," in *Sparse Matrix Computations*, (J.R. Bunch and D.J. Rose, eds.), Academic Press, 147-176 (1976).

[86] T.L. MAGNANTI AND B.L. GOLDEN, "Transportation Planning: Network Models and Their Implementation," Ch. 16 in *Studies in Operations Management*, (A.C. Hax, ed.), North-Holland, Amsterdam (1978).

[87] T.L. MAGNANTI, J.F. SHAPIRO AND M.H. WAGNER, "Generalized Linear Programming Solves the Dual," *Management Science*, 11, 1195-1203 (1976).

[88] T. MAGNANTI AND R. WONG, "Accelerating Benders Decomposition: Algorithmic Enhancements and Model Selection Criteria," *Operation Research* (to appear).

[89] T.G. MAIRS, G.W. WAKEFIELD, E.L. JOHNSON AND K. SPIELBERG, "On a Production Allocation and Distribution Problem," *Management Science*, 24, 1622-1630 (1978).

[90] A.S. MANNE, "Plant Location Under Economies-of-Scale-Decentralization and Computation," *Management Science*, 11, 213-235 (1964).

[91] R.S. MC GAUGHEY, K.W. GOHRING AND R.N. MC BRAYER, "Planning Locomotive and Caboose Distribution," *Rail International*, 4, 1213-1218 (1973).

[92] N. MEGIDDO, "Combinatorial Optimization with Rational Objective Functions," *Mathematics of Operations Research*, 4, 414-424 (1979).

[93] J. MINAS AND L. MITTEN, "The Hub Operations Scheduling Problem," *Operations Research*, 6, 329-345 (1958).

[94] R.H. MOLE, "A Survey of Local Delivery Vehicle Routing Methodology," *Jour. Operational Research Soc.*, 30, 245-252 (1979).

[95] E.K. MORLOK, W.P. PIERSKALLA AND W.L. VANDERSYPEN, "Schedule Planning and Timetable Construction for Suburban Railways," *Proceedings of the International Conference on Transportation Research,* Bruges, Belgium, 757-767 (1974).

[96] J.A. MUCKSTAT AND S.A. KOENIG, "An Application of Lagrangian Relaxation to Scheduling in Power Generation Systems," *Operations Research,* 25, 387-403 (1977).

[97] G.L. NEMHAUSER, "Scheduling Local and Express Trains," *Transportation Science,* 3, 164-175 (1969).

[98] G. NEWELL, *Traffic Flow,* MIT Press, Cambridge, Mass. (1980).

[99] R. NEWTON AND W. THOMAS, "Bus Routing in a Multi-School System," *Computers and Operations Research,* 1, 213-222 (1974).

[100] J.B. ORLIN, "Minimizing the Number of Vehicles to Meet a Fixed Periodic Schedule: An Application of Periodic Posets," Working Paper, Sloan School of Management, MIT (July 1980).

[101] J.B. ORLIN, "Minimum Convex-Cost Dynamic Network Flows," Technical Report No. 172, Operations Research Center, MIT (March 1980).

[102] M. PADBERG AND S. HONG, "On the Symmetric Traveling Salesman Problem: A Computational Study," *Mathematical Programming Study 12,* 78-107 (1980).

[103] E.R. PETERSEN AND H.V. FULLERTON, "An Optimizing Network Model for the Canadian Railways," *Rail International,* 4, 1187-1192 (1973).

[104] J.F. PIERCE, "Direct Search Algorithms For Truck Dispatching Problems," *Transportation Research,* 3, 1-42 (1969).

[105] M. POLLACK, "Some Elements of the Airline Fleet Planning Problem," *Transportation Research,* 11, 301-310 (1977).

[106] H. PSARAFTIS, "A Dynamic Programming Solution to the Single Vehicle Many-to-Many Immediate Request Dial-a-Ride Problem," *Transportation Science,* 2, 130-154 (1980).

[107] H. PSARAFTIS AND G. THARAKAN, "A Dynamic Programming Approach to the Dial-a-Ride Problem: An Extension to the Multi-Vehicle Case," Report R79-39, Dept. of Civil Engineering, MIT (August 1979).

[108] R. RICHARDSON, "An Optimization Approach to Routing Aircraft," *Transportation Science,* 10, 52-71 (1976).

[109] D.J. ROSENKRANTZ, R.E. STEARNS AND P.M. LEWIS, "An Analysis of Several Heuristics for the Traveling Salesman Problem," *SIAM J. Computing,* 6, 563-581 (1977).

[110] J.L. SAHA, "On Some Problems in Railway Networks," Ph.D. Dissertation, Dept. of Operations Research, Case Western Reserve University (June 1975).

[111] F.J.M. SALZBORN, "The Minimum Fleetsize for a Suburban Railway System," *Transportation Science*, 4, 383-402 (1970).

[112] L. SCHRAGE, "Formulation and Structure of More Complex/Realistic Routing and Scheduling Problems," *Networks* (to appear).

[113] T. SEXTON, "The Single Vehicle Many to Many Routing and Scheduling Problem," Ph.D. Thesis, State University of New York at Stony Brook, Stony Brook, New York (1979).

[114] J.F. SHAPIRO, "A Survey of Lagrangian Techniques For Discrete Optimization," *Annals of Discrete Mathematics - No. 5 Discrete Optimization*, (P. Hammer, E. Johnson and B. Korte, eds.) North Holland Publ. Co. 113-138 (1979).

[115] W.B. SHEPARDSON AND R.E. MARSTEN, "A Langrangian Relaxation Algorithm for the Two Duty Period Scheduling Problem," *Management Science*, 26, 274-281 (1980).

[116] R. SIMPSON, "Scheduling and Routing Models for Airline Systems," Unpublished Report, Flight Transportation Laboratory, MIT (1969).

[117] D. STEIN, "An Asymptotic, Probabilistic Analysis of a Routing Problem," *Mathematics of Operations Research*, 3, 89-101 (1978).

[118] D. STEIN, "Scheduling Dial-a-Ride Transportation Systems," *Transportation Science*, 12, 232-249 (1978).

[119] W. STEWART AND B. GOLDEN, "A Vehicle Routing Algorithm Based on Generalized Lagrange Multipliers," *Proc. of AIDS 1979 Annual Convention* (L. Moore, K. Moore and B. Taylor, eds.), New Orleans, 2, 108-110 (1979).

[120] B. SZPIGEL, "Optimal Train Scheduling on a Single Track Railway," *Operations Research, 1972* (M. Ross, ed.), 343-352 (1972).

[121] J. TOUNGE, "Automatic Allocation of Locomotives on the SNCF," *Proceedings of the Third International Symposium on Rail Cybernetics*, 179-182 (1970).

[122] A. TRUSKOLASKI AND W. GRABOWSKI, "Mathematical Model of Empty-Wagon Distribution in n-day Planning Conditions," *Proceedings of the Symposium on the use of Cybernetics on the Railways*, 104-108 (1963).

[123] V.R. VUCHIC AND G.F. NEWELL, "Rapid Transit Interstation Spacings for Minimum Travel Time," *Transportation Science*, 2, 303-339 (1968).

[124] D.R. YOUNG, "Scheduling a Fixed Schedule, Common Carrier Passenger Transportation Systems," *Transportation Science*, 4, 243-269 (1970).

[125] H.M. WAGNER AND T.M. WHITIN, "Dynamic Version of the Economic Lot Size Model," *Management Science,* 5, 89-96 (1958).

[126] W.W. WHITE AND A.M. BOMBERAULT, "A Network Algorithm for Empty Freight Car Allocation," *IBM Systems Journal,* 9, 147-169 (1969).

[127] S.C. WIRASINGHE, V.F. HURDLE AND G.F. NEWELL, "Optimal Parameters for a Coordinated Rail and Bus Transit System," *Transportation Science,* 11, 359-374 (1977).

[128] R.T. WONG, "Integer Programming Formulations of the Traveling Salesman Problem," *Proc. of IEEE International Conference on Circuits and Computers, 1980,* 149-152 (1980).