A Generalized Assignment Heuristic for Vehicle Routing

Marshall L. Fisher

The Wharton School, University of Pennsylvania, Philadelphia, Pennsylvania 19104

Ramchandran Jaikumar

Graduate School of Business Administration, Harvard University, Boston, Massachusetts 02163

We consider a common variant of the vehicle routing problem in which a vehicle fleet delivers products stored at a central depot to satisfy customer orders. Each vehicle has a fixed capacity, and each order uses a fixed portion of vehicle capacity. The routing decision involves determining which of the demands will be satisfied by each vehicle and what route each vehicle will follow in servicing its assigned demand in order to minimize total delivery cost. We present a heuristic for this problem in which an assignment of customers to vehicles is obtained by solving a generalized assignment problem with an objective function that approximates delivery cost. This heuristic has many attractive features. It has outperformed the best existing heuristics on a sample of standard test problems. It will always find a feasible solution if one exists, something no other existing heuristic can guarantee. It can be easily adapted to accommodate many additional problem complexities. By parametrically varying the number of vehicles in the fleet, our method can be used to optimally solve the problem of finding the minimum size fleet that can feasibly service the specified demand.

I. INTRODUCTION

Vehicle routing is a challenging logistics management problem. There are many variations of the problem ranging from school bus routing to the dispatching of delivery trucks for consumer goods. In all cases, the basic components of the problem are a fleet of vehicles with fixed capabilities (capacity, speed, etc.) and a set of demands for transporting certain objects (school children, consumer goods, etc.) between specified pickup and delivery points. The routing decision involves determining which of the demands will be satisfied by each vehicle and what route each vehicle will follow in servicing its assigned demand. These decisions should be made to minimize the cost of operating the vehicle fleet. Principal cost items include fuel, personnel, and vehicle depreciation. These costs are usually large and highly sensitive to how routing decisions are made. Vehicle routes must also satisfy a variety of constraints arising from factors such as fixed vehicle capacity and union regulations on driver work schedules.

We consider a common variant of the vehicle routing problem. A vehicle fleet delivers products stored at a central depot to satisfy customer orders that cover some period of time into the future. The customers specify their orders prior to the start of each period, and the vehicles must then be scheduled to deliver the period's orders. Each vehicle has a fixed capacity, and each order uses a fixed portion of vehicle capacity. Examples include scheduling the deliveries of a large department store and of a processed food distributer.

To provide a precise statement of this problem we introduce notation and specify an integer programming formulation.

Constants

K = number of vehicles.

n = number of customers to which a delivery must be made. Customers are indexed from 1 to n and index 0 denotes the central depot.

b_k = capacity (weight or volume) of vehicle k.

a_i = size of the delivery to customer i.

c_{ij} = cost of direct travel from customer i to customer j.

Variables

$$y_{ik} = \begin{cases} 1, \text{ if the order from customer } i \text{ is delivered by vehicle } k \\ 0, \text{ otherwise.} \end{cases}$$

$$x_{ijk} = \begin{cases} 1, \text{ if vehicle } k \text{ travels directly from customer } i \text{ to customer } j \\ 0, \text{ otherwise.} \end{cases}$$

An integer programming formulation of the problem of routing to minimize cost subject to vehicle capacity constraints is given below. We denote this problem by (VRP).

Formulation of the Vehicle Routing Problem (VRP)

$$\min \sum_{ijk} c_{ij} x_{ijk} \tag{1}$$

such that

$$\sum_i a_i y_{ik} \leqslant b_k, \quad k = 1, \ldots, K \tag{2}$$

$$\sum_k y_{ik} = \begin{cases} K, & i = 0 \\ 1, & i = 1, \ldots, n \end{cases} \tag{3}$$

$$y_{ik} = 0 \text{ or } 1, \quad \begin{array}{l} i = 0, \ldots, n \\ k = 1, \ldots, K \end{array} \tag{4}$$

$$\sum_i x_{ijk} = y_{jk}, \qquad j = 0, \ldots, n \tag{5}$$

$$\sum_j x_{ijk} = y_{ik}, \qquad i = 0, \ldots, n \tag{6}$$

$$k = 1, \ldots, K.$$

$$\sum_{ij \in S \times S} x_{ijk} \leqslant |S| - 1, \qquad \begin{array}{l} S \subseteq \{1, \ldots, n\} \\ 2 \leqslant |S| \leqslant n - 1 \end{array} \tag{7}$$

$$x_{ijk} = 0 \text{ or } 1, \qquad \begin{array}{l} i = 0, \ldots, n \\ j = 0, \ldots, n \end{array} \tag{8}$$

Two well-known combinatorial optimization problems are embedded within this formulation. Constraints (2)–(4) are the constraints of a generalized assignment problem and insure that each route begins and ends at the depot (customer 0), that every customer is serviced by some vehicle, and that the load assigned to a vehicle is within its capacity. If the y_{ik} are fixed to satisfy (2)–(4), then for given k, constraints (5)–(8) define a traveling salesman problem over the customers assigned to vehicle k.

Literally person-centuries have been devoted to developing a sophisticated solution theory for the traveling salesman and generalized assignment models embedded within (VRP). By contrast, existing computer-based methods for (VRP) are relatively simple and use little of this theory. Because of the complexity of (VRP), all previously proposed practical methods have been heuristics that find an approximately optimal solution. Existing heuristic methods are reviewed in Sec. II and shown to suffer from a serious limitation. Vehicle capacity constraints do not play a central role in the decision rules of these methods so that tightly constrained problems can easily terminate with a poor or infeasible solution. This difficulty has been recognized by several researchers and practitioners as a serious barrier in the use of these methods (Christofides [3], Krolak and Nelson [12], Rau [17], Shuster and Schur [20]).

We present here a new heuristic for (VRP) in which an assignment of customers to vehicles is obtained by solving a generalized assignment problem with constraints (2)–(4) and an objective function that approximates the cost of the traveling salesman problem tours that must be made for each vehicle to service its assigned customers. Once this assignment has been made, a complete solution is obtained by applying any traveling salesman problem heuristic or optimizing algorithm to obtain the delivery sequence for the customers assigned to each vehicle.

A detailed description of this generalized assignment heuristic is given in Sec. 3. In Sec. 4 we present the results of computational testing on a sample of standard test problem taken from the literature. In these tests, the new heuristic proposed here outperformed the best existing vehicle routing heuristics.

In addition to this outstanding computational performance, our heuristic has several attractive attributes. First, because the essential feasibility constraints (2) and (3) are included in the generalized assignment problem, the heuristic will always find a feasi-

ble solution if one exists. Second, when the generalized assignment problem is solved, we are considering the impact of a customer assignment to a vehicle on every other possible assignment in light of vehicle capacity constraints. This avoids a problem faced by sequential assignment or limited adjustment hueristics that can "paint themselves into a corner" by unknowingly making initial assignments that eventually force very expensive assignments in order to maintain feasibility. Third, the method can easily be adapted to accommodate a number of important problem complexities, including multiple depots, multiple time periods, the option of not delivering to a customer at a penalty, constraints on the time duration of a vehicle route, and multiple capacity constraints (e.g., weight and volume). Finally, by applying our method a number of times for different values of K and $(b_1, \ldots, b_K)$, it is possible to determine the tradeoff between fleet size and the operating costs included in objective (1) of (VRP). This kind of parametric analysis would be useful, for example, in evaluating vehicle acquisition decisions. Also, because our method will always find a feasible solution if one exists, it is possible to use our method with parametric variation of K to optimally determine the minimum fleet size that can feasibly service a fixed set of demand requirements.

II. REVIEW OF EXISTING METHODS

Past work on (VRP) has been concerned almost exclusively with heuristics. The heuristics which have been developed for (VRP) are largely modifications of traveling salesman problem heuristics and are of four types:

    (i) tour building heuristics,
    (ii) tour improvement heuristics,
    (iii) two-phase methods, and
    (iv) incomplete optimization methods.

In the first type a link between two customers is sequentially added until all customers have been assigned to some route. Every time a link is added, the vehicle capacity constraints are checked for violation. The choice of a link is motivated by some measure of cost savings.

Tour improvement heuristics begin with a feasible vehicle schedule. At every iteration some combination of links are exchanged for another and a check is made to see if the exchange is both feasible and reduces cost.

In the two-phase method, customers are first assigned to vehicles without specifying the sequence in which they are to be visited. In phase 2, routes are obtained for each vehicle using a traveling salesman problem heuristic.

Incomplete optimization methods apply some optimization algorithm, such as branch and bound, and simply terminate prior to optimality.

The most often used tour building heuristic is the Clarke and Wright method [5]. The Clarke and Wright method begins with an infeasible solution in which every customer is supplied individually by a separate vehicle. By combining any two of the customers we would use only one of the two vehicles and also reduce the solution cost. Recall that customer 0 denotes the central depot. The cost of serving customers i and j individually by two vehicles is $c_{oi} + c_{io} + c_{oj} + c_{jo}$ while the cost of one vehicle

visiting i and j sequentially on the same route is $c_{oi} + c_{ij} + c_{jo}$. Thus, combining i and j results in a savings of

$$s_{ij} = c_{io} + c_{oj} - c_{ij}.$$

Clarke and Wright link the customers i and j with maximum s_{ij} subject to the requirement that the combined route be feasible. Customers i and j are now regarded as a single macro customer. A vehicle may travel from city l to the macro customers at a cost of c_{li} and from the macro customer to l at a cost of c_{jl}. With this convention, the route combining operation can be applied repeatedly.

Savings are ordered from the greatest to the least, and this list is scanned from the top to generate a sequence of partial routes, each time checking for feasibility. In scanning the list we can simultaneously form partial routes for all vehicles or sequentially add customers to a given route until the vehicle is loaded. The latter is called the sequential Clarke and Wright method.

There have been many modifications to the basic Clarke and Wright method. Gaskell [9] and Yellow [22] independently introduced the concept of a modified savings given by $s_{ij} - \theta c_{ij}$ where θ is a scalar parameter. By varying θ, one can place greater or less emphasis on the cost of travel between two nodes, depending on their position relative to the central depot. This parameter can be altered and different solutions obtained. The best of these is then chosen. Golden et al. [11] have used computer science techniques to substantially reduce the running time of Clarke and Wright.

The tour improvement heuristics are based on the Lin [13] and Lin-Kernighan [14] heuristic for the traveling salesman problem. Christofides and Eilon [2] and Russell [19] have modified this heuristic for vehicle routing.

Two phase methods include those of Tyagi [21], Gillett and Miller [10], and Christofides et al. [4]. The methods in [21] and [10] both use a modified Lin-Kernighan heuristic in phase 2. In phase 1, both methods use the distance between customers as cost. Tyagi assigns customers sequentially to vehicles using a nearest neighbor rule. Each customer assigned to a vehicle is chosen to be closest to the customer last assigned to that vehicle.

Gillett and Miller use a "sweep" algorithm for phase 1 in which the location of customers is represented in a polar coordinate system with origin at the central depot. A customer is chosen at random and the ray from the origin through the customer is "swept" either clockwise or counterclockwise. Customers are assigned to a given vehicle as they are swept, until the capacity constraint for that vehicle is reached. Then a new vehicle is selected and the sweep continues, with assignments now being made to the new vehicle.

The Christofides et al. [4] two phase method begins by applying a minimal insertion cost heuristic for inserting customers to emerging routes. At each step, the Lin-Kernighan heuristic is applied to the customers that have been assigned to each vehicle. In phase 2, a customer i_k is designated in each of the routes formed in phase 1. Beginning with the K routes that join the depot to i_k, $k = 1, \ldots, K$, the remaining customers are inserted using a rule based on the cost of inserting a customer in alternative routes.

The only example of a heuristic based on incomplete optimization of which we are

aware is the tree-search method reported in [4]. This is essentially a branch and bound algorithm turned into a heuristic by early termination.

We conclude this subsection with the observation that none of the methods described here place much emphasis on the vehicle capacity constraints. While these constraints are checked for violation wherever possible, they have no other influence on the choices that are made in forming a solution. For this reason, existing heuristics can easily terminate with an infeasible or bad solution if capacity constraints are moderately tight.

III. A NEW HEURISTIC FOR (VRP)

The basic idea of our approach can be described in terms of the following reformulation of (VRP) as a nonlinear generalized assignment problem.

$$\min \sum_k f(y_k) \tag{1'}$$

such that

$$\sum_i a_i y_{ik} \leqslant b_k, \qquad k = 1, \ldots, K \tag{2}$$

$$\sum_k y_{ik} = \begin{cases} K, & i = 0 \\ 1, & i = 1, \ldots, n \end{cases} \tag{3}$$

$$y_{ik} = 0 \text{ or } 1, \qquad \begin{matrix} i = 0, \ldots, n \\ k = 1, \ldots, K \end{matrix} \tag{4}$$

where $f(y_k)$ is the cost of an optimal traveling salesman problem tour of the customer in $N(y_k) = \{i \mid y_{ik} = 1\}$. The function $f(y_k)$ can be defined mathematically by

$$f(y_k) = \min \sum_{ijk} c_{ij} x_{ijk} \tag{1''}$$

such that

$$\sum_i x_{ijk} = y_{jk}, \qquad j = 0, \ldots, n \tag{5'}$$

$$\sum_j x_{ijk} = y_{ik}, \qquad i = 0, \ldots, n \tag{6'}$$

$$\sum_{ij \in SxS} x_{ijk} \leqslant |S| - 1, \qquad \begin{matrix} S \subseteq \{1, \ldots, n\} \\ 2 \leqslant |S| \leqslant n - 1 \end{matrix} \tag{7'}$$

$$x_{ijk} = 0 \text{ or } 1, \qquad \begin{matrix} i = 0, \ldots, n \\ j = 0, \ldots, n \end{matrix} \tag{8'}$$

Of course, $f(y_k)$ is an extremely complicated function and cannot even be written down for nontrivial problems. Our heuristic is based on constructing a linear approximation $\sum_{i=1}^{n} d_{ik} y_{ik}$ of $f(y_k)$ and solving (1'), (2), (3), (4) with (1') replaced by

$$\sum_{k=1}^{K} \sum_{i=1}^{n} d_{ik} y_{ik}. \qquad (1'')$$

The solution of this linear generalized assignment problem defines a feasible assignment of customers to vehicles. We then determine a delivery sequence for the customers assigned to each vehicle by applying any traveling salesman problem heuristic or optimizing algorithm.

There are many plausible methods for constructing a linear approximation of $f(y_k)$. We first describe a simple method that was used to obtain the computational results reported in the next section. Then we indicate some possible variations.

We begin with a set of "seed" customers $i_1, \ldots, i_K$ that are assigned to vehicles $1, \ldots, K$, respectively. The coefficient d_{ik} is then set to the cost of inserting customer i into the route in which vehicle k travels from the depot directly to customer i_k and back. Specifically,

$$d_{ik} = \min \left[c_{oi} + c_{ii_k} + c_{i_k o}, c_{oi_k} + c_{i_k i} + c_{io} \right] - \left[c_{oi_k} + c_{i_k o} \right].$$

Figure 1 shows this computation for an example that we created for illustrative purposes. In this example $n = 25$, $K = 7$, and the c_{ij} are given by the Euclidean distance between points. We have chosen customers 4, 7, 10, 13, 16, 20, and 24 as seed customers. The figure shows the 7 seed routes and the computation of d_{ik} for $i = 23$ and k the vehicle assigned to customer 24. The solution obtained by the generalized assignment heuristic has cost 159.59 and is shown in Figure 2. For comparison, the Sweep solution which has a cost of 166.73 is shown in Figure 3. The Clarke and Wright solution had a cost of 164.03.

Seed customers can be selected either by an automatic rule or by a scheduler who has responsibility for operating the computerized routing system and implementing its output. There are many advantages to the latter approach. Usually the scheduler has some "feel" for the problem and appreciates the opportunity to make this experience known to the computer system. There are many considerations he can use to select seed customers. For example, customers often lie along radial corridors corresponding to major thoroughfares, and the most distant customers along these corridors are natural seed customers. Customers for which $a_i > \frac{1}{2} b_k$ can also be made seed customers, since any pair of these cannot be on the same route. If our heuristic is implemented on an interactive computer system with graphics display, the scheduler can experiment with different selections of seed customers and immediately see the effect on cost and routing decisions. This gives him a sense of involvement and control that is crucial to a successful implementation.
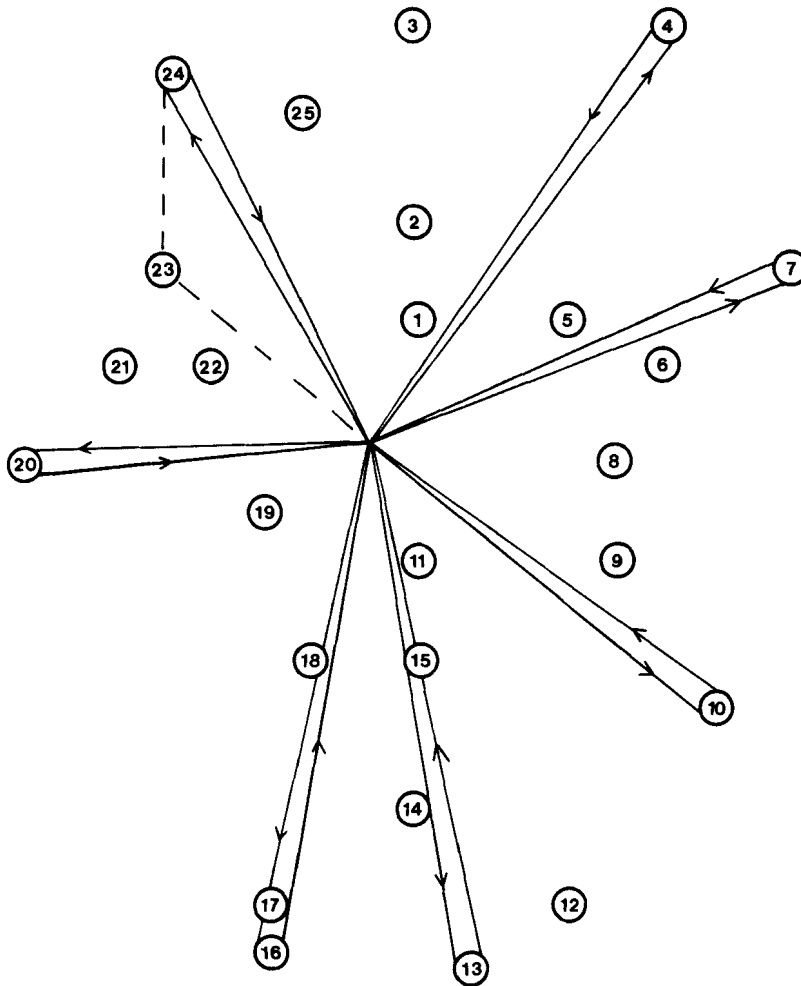
FIG. 1.   Computation of d_{ik}.

To illustrate the possibilities for automatic selection of seed customers, we will describe the rule we've used in our computational experiments. This rule will be given for the planar case (all customers are located at points in a plane and c_{ij} is the Euclidean distance between points i and j) with $b_k = b$ for all k. It can be generalized to other cases. We determine K seed points $w_1, \ldots, w_K$ in the plane rather than K seed customers. These points are used exactly like seed customers to initialize the heuristic. To determine $w_1, \ldots, w_K$, the plane is partitioned into K cones corresponding to the K vehicles. Then w_k is located on the ray bisecting cone k.

To determine these cones, we first partition the plane into n smaller cones, one for each customer. The infinite half ray forming the boundary between two customer cones is positioned to bisect the angle formed by half rays through the two immediately adjacent customers. Associate the weight a_i with customer cone i and
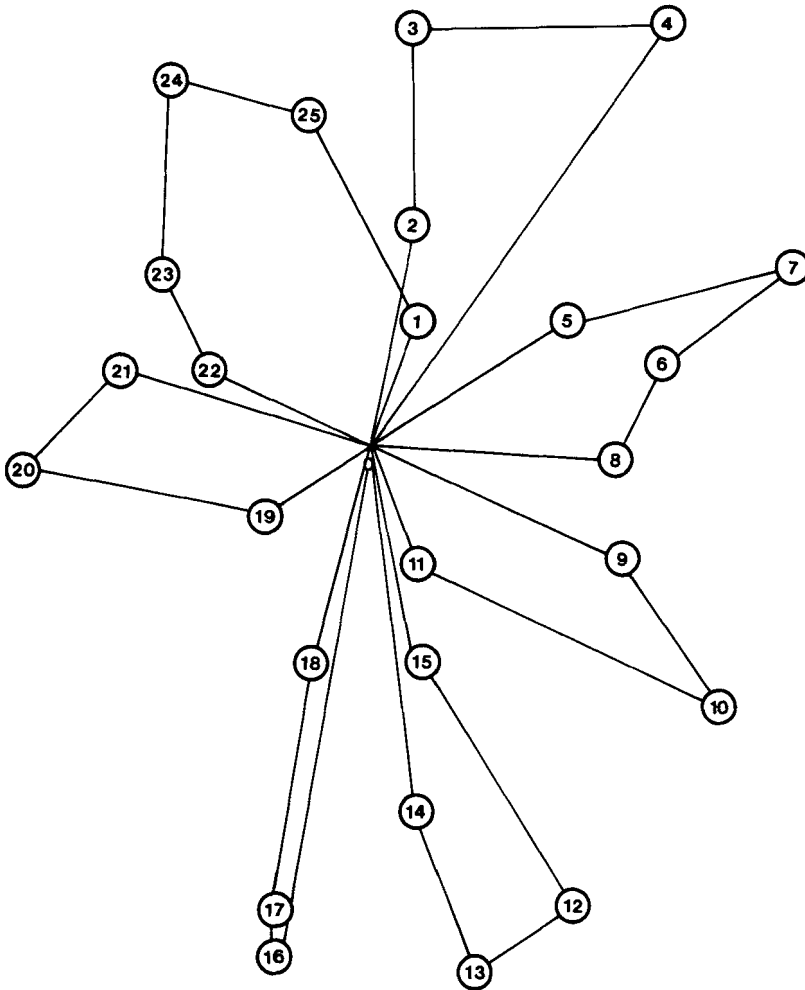
FIG. 2. Fisher–Jaikumar generalized assignment solution (Z = 159.50).

define

$$\alpha = \frac{\sum\limits_{i=1}^{n} a_i}{Kb} \; .$$

Each vehicle cone is then formed from a contiguous group of customer cones or fractions of customer cones. The weight of the group is required to equal αb. A fraction of a customer cone contributes the same fraction of its weight to the total group weight. The point w_k is located along the ray bisecting the kth cone. The distance of w_k from the origin is fixed so that the weight included inside the arc through w_k is 0.75 αb. This weight is defined to equal the sum of the a_i for all customers inside the
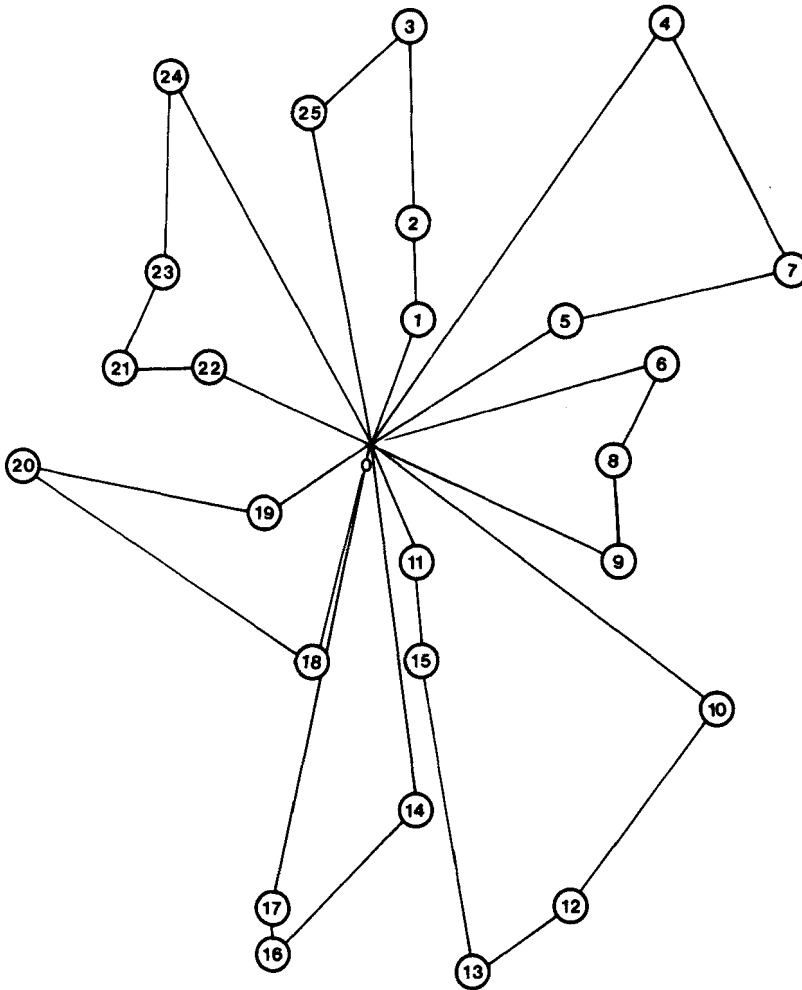
FIG. 3.   Sweep solution ($Z = 166.73$).

arc plus a fraction of a_i for the customer just outside the arc. This fraction is $A/(A + B)$, where A is the distance to the arc from the customer just inside the arc and B is the distance to the arc from the customer just outside the arc.

This process is illustrated in Figure 4 for an example with $K = 3$ and $b_k = 30$ for $k = 1, 2, 3$. The points in the figure represent customers, and the numbers next to the points give the a_i. The dashed lines define the customer cones, and the solid lines the vehicle cones.

In our current computer implementation, the traveling salesman problems for the customers assigned to each vehicle are solved optimally using an algorithm similar to the one reported in Miliotis [15], [16]. We use Gomory's Method of Integer Forms to solve formulation $(1'')$, $(5')$-$(8')$ with the y_{ik} fixed. Constraints in the set $(7')$ are generated only as needed.
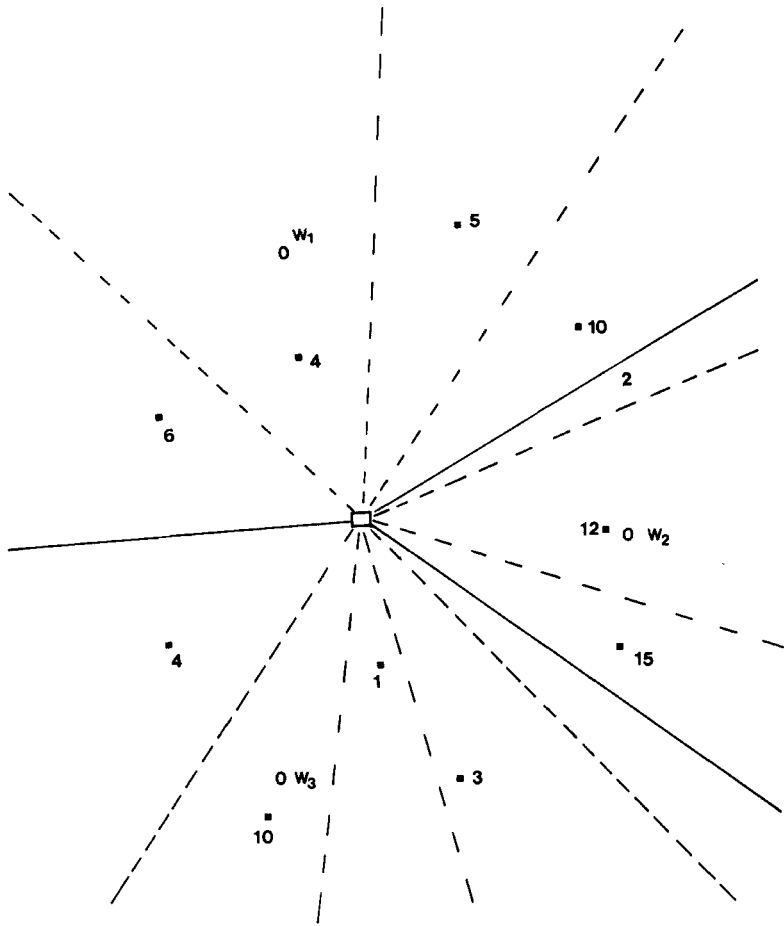
FIG. 4.   Illustration of the selection of seed points.

Our algorithm for the generalized assignment problem is based on a Lagrangian relaxation in which the multipliers are determined by a multiplier adjustment method of the type described in Fisher [6]. This algorithm has outperformed the very successful Ross and Soland [18] algorithm. A detailed description is given in Fisher, Jaikumar, and Van Wassenhove [8].

We are currently experimenting with a number of possible variations on the heuristic we have described here. These include different seed customer selection rules, the use of a modified insertion cost ala Gaskell [9] and Yellow [22], and schemes for iteratively adjusting the d_{ik}. In this last approach, we solve the generalized assignment problem several times, adjusting the d_{ik} after each solution to better approximate $f(y_k)$.

An optimizing algorithm can also be constructed by using this iterative procedure with Benders decomposition [1]. Each time the generalized assignment problem is solved, a lower linear support of $f(y_k)$ is constructed from the dual variables of $(1'')$,

$(5')$–$(8')$ and added to the generalized assignment problem. To obtain dual variables, we relax integrality on x_{ijk} or impose this integrality with standard integer programming cuts. A complete description of this approach is given in Fisher and Jaikumar [7].

IV. COMPUTATIONAL RESULTS

Computational tests were performed using 12 standard test problems taken from the literature. Our method was compared with four other heuristics:

  (i) Clarke and Wright [5]
  (ii) Sweep [10]
  (iii) Christofides et al. tree search [4]
  (iv) Christofides et al. 2 phase [4]

Summary characteristics of the 12 problems are given in Table I. In these problems, the b_k were identical for all k. The value in the column headed K^* is the minimum number of trucks that will admit a feasible solution. Specifically, K^* satisfies

$$\sum_{K=1}^{K^*-1} b_k < \sum_{i=1}^{n} a_i \leqslant \sum_{k=1}^{K^*} b_k.$$

In a few cases, Clarke and Wright required more than K^* vehicles. Otherwise, all results are with $K = K^*$. The last column gives an indication of the tightness of the vehicle capacity constraints.

All test problems were planar. That is, customers are located at points in the plane, and c_{ij} is the Euclidean distance between points i and j. Problems 1, 2, and 3 are 50, 75, and 100 customer problems, respectively. The data for these problems was randomly generated and is given in [2]. Problem 4 is a 150 customer problem produced

TABLE I.   Summary of problem characteristics.

Problem	n	K^*	$\sum_{i=1}^{n} a_i \Big/ \sum_{k=1}^{K^*} b_k$
1	50	5	0.97
2	75	10	0.97
3	100	8	0.93
4	150	12	0.94
5	199	17	0.95
6	50	6	0.81
7	75	12	0.81
8	100	9	0.83
9	150	15	0.71
10	199	19	0.77
11	100	10	0.91
12	100	10	0.91

by adding the customers of problems 1 and 3 with the depot and vehicle capacities as in problem 3. Problem 5 is a 199 customer problem produced by adding the customers of problem 4 with the first 49 customers of problem 2. Problems 6 to 10 are the same as problems 1 to 5 with additional restrictions on the maximum allowable route time for each problem. The restrictions on route times for problems 6 to 10, respectively, are 200, 160, 230, 200, and 200. Travel times between customers are assumed to be equal to the distance between the customers. Additonally, an unloading time of 10 units is incurred for each customer stop.

The introduction of a constraint on route time requires a modification of our heuristic as given in the previous section. Analogous to the function $f(y_k)$ defined in Sec. III, we define $t(y_k)$ to equal the travel time of an optimal tour of the customers in $N(y_k) = \{i \mid y_{ik} = 1\}$. A linear approximation of $t(y_k)$ is constructed using the method for approximating travel cost given in Sec. III. The linear approximation of $t(y_k)$ is used to construct a linear constraint that approximates the route time restriction. These linear route time constraints are then added to the generalized assignment problem $(1'')$, (2), (3), and (4). An algorithm to solve this generalized assignment problem with side constraints is described in [8]. This algorithm employs a dualization of the side constraints. Because the route time constraints are approximate, a final feasibility check is required.

Problems 11 and 12 are structured problems in which customers are grouped in clusters. The problems seem to resemble real problems more closely than problems 1 to 10. The data for these problems is given in [4].

A comparison of the five methods is reported in Table II. Results for our method were obtained by us; results for the other four methods are taken from [4].

Table III provides a comparison of solution quality and running time for the five methods. In terms of solution quality, the Fisher–Jaikumar generalized assignment method clearly outperformed the other four. The generalized assignment method found the best solution in 9 of the 12 problems and had the lowest average solution value over all 12 problems.

The closest competitor was the Christofides et al. tree search method. This is a restricted branch and bound method and is theoretically capable of solving the problem optimally if the restrictions on branching and searching the tree are removed. The quality of the results depends on how much restriction is placed on the tree search. With increasing quality, one suffers the progressive increase in computation times. Besides, the form of this algorithm implies that computational times will increase exponentially with the number of customers, as can be seen in Table I.

The Christofides et al. two phase method is similar to the method presented here in that both methods select seed customers, compute insertion costs, and then assign customers to trucks based on these costs. The Christofides et al. two phase uses a heuristic to make this assignment, while the Fisher–Jaikumar method optimally solves a generalized assignment problem. If the insertion costs are reasonable approximations of the true costs, then the generalized assignment method should perform uniformly better than the two phase method. This is substantiated by the results. The two phase method obtains the best solution in only one of the twelve problems and is 2.4% more expensive on average than the generalized assignment method. In the two problems where the two phase method outperforms the generalized assignment method, we con-

TABLE II. Results of computational testing—cost and CPU time.

Problem	n	Clarke–Wright		Sweep		Christofides et al. tree search		Christofides et al. 2 phase		Fisher–Jaikumar	
		Cost	CPU[a]	Cost	CPU[a]	Cost	CPU[a]	Cost	CPU[a]	Cost	CPU[a]
1	50	585	0.8	532	12.2	534	7.1	550	2.5	524	9.33
2	75	900	1.7	874	24.3	871	15.6	883	4.2	857	11.95
3	100	886	2.4	851	65.1	851	38.2	851	9.7	833	17.7
4	150	1204	6.6	1079	142.0	1064	81.1	1093	11.8	1014	33.6
5	199	1540	11.0	1389	252.2	1386	138.4	1418	16.7	1420	40.1
6	50	619	1.3	560	11.4	560	5.3	565	2.6	560	15.2
7	75	976	2.6	933	23.8	924	13.6	969	4.4	916	20.6
8	100	973	3.5	888	58.5	885	33.4	915	7.0	885	52.2
9	150	1426	11.2	1230	134.7	1217	74.0	1245	10.1	1230	121.3
10	199	1800	14.0	1518	238.5	1509	135.6	1508	15.8	1518	136.6
11	100	831	2.4	937	50.8	859	39.3	827	6.4	824	6.1
12	100	877	2.9	949	53.6	878	45.2	876	6.3	848	13.6

[a] CPU times for Fisher–Jaikumar are on the DEC-10. All other times are for the CDC 6600, which is considered to be seven times faster than the DEC-10. All times are in seconds.

TABLE III. Comparison of the five methods.

Method	Average solution value for 12 problems	% Over Fisher–Jaikumar average solution value	Number of problems on which method found the best solution	Average CPU[a] time (sec)
Clarke–Wright	1051.42	10.4%	0	5.03
Sweep	978.33	2.7%	1	88.93
Christofides et al. tree search	961.33	0.94%	4	52.23
Christofides et al. 2-phase	975.00	2.4%	1	8.13
Fisher–Jaikumar generalized assignment	952.42	0%	9	5.69

[a]Generalized Assignment time has been divided by seven to allow for the difference in speed between the DEC-10 and the CDC 6600.

jecture that the difference is probably due to a different method of selecting the seed customers and if the same seed customers and insertion costs are used, the generalized assignment method should do better.

The version of Sweep implemented in [4] applies the basic sweep iteration described in Sec. II for many different starting rays. The solution values reported are the best of these numerous runs. The solution times are the sums of the times for different starting rays. The sweep method did uniformly better than the Clarke and Wright method for the random problems 1 to 10, but worse for the structured problems 11 and 12. Both methods were substantially poorer than the generalized assignment method.

CPU times for the generalized assignment method (see Table II) are smaller on average than all other methods except Clarke and Wright. The Clarke and Wright times are only slightly smaller, and its performance in terms of solution quality is much worse.

We also note that Russell [19] has obtained solutions to the first 3 problems with costs of 524, 854, and 833, respectively. However, the computational requirements of his method are prohibitive. The CPU times on an IBM 370/168 (a faster machine than the DEC 10) for these 3 problems were 15, 245, and 100 sec, respectively.

In conclusion, the Fisher–Jaikumar generalized assignment method has demonstrated impressive computational performance on a wide range of test problems.

References

[1] J. F. Benders, "Partitioning Procedures for Solving Mixed-Variables Programming Problems," *Numer. Math.*, 4 238–252 (1962).
[2] N. Christofides and S. Eilon, "An Algorithm for the Vehicle Dispatching Problems," *Oper. Res. Q.* 20 309 (1969).
[3] Christofides, N. "The Traveling Salesman Problem and Its Applications," ORSA/TIMS/AIIE Distribution Conference, Feb. 1978.
[4] N. Christofides, A. Mingozzi, and P. Toth, "The Vehicle Routing Problem," Urbino Working Paper, July, 1978.

[5] G. Clarke and J. Wright, "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points," *Oper. Res.*, 12(4), 568–581 (1964).

[6] M. L. Fisher, "The Lagrangian Relaxation Method for Solving Integer Programming Problems," *Manag. Sci.*, 27, No. 1 (1981).

[7] M. L. Fisher and R. Jaikumar, "A Decomposition Algorithm for Large-Scale Vehicle Routing," Decision Sciences Working Paper 78-11-05, University of Pennsylvania, July 1978.

[8] M. L. Fisher, R. Jaikumar, and L. Van Wassenhove, "A Multiplier Adjustment Method for the Generalized Assignment Problem," Decision Sciences Working Paper, University of Pennsylvania, March, 1981.

[9] T. J. Gaskell, "Basis for Vehicle Fleet Scheduling," *Oper. Res. Q.*, 18 281 (1967).

[10] B. Gillett and L. Miller, "A Heuristic Algorithm for the Vehicle Dispatch Problem," *Oper. Res.*, 22 340–349 (1974).

[11] B. L. Golden, T. L. Magnanti, and H. Q. Nguyen, "Implementing Vehicle Routing Algorithms," *Networks*, 7(2), 113–148 (1977).

[12] P. Krolak and J. Nelson, "A Family of Truck Load Clustering Heuristics for Solving Vehicle Scheduling Problems," Working Paper, Vanderbilt University (1978).

[13] S. Lin, "Computer Solution of the Traveling Salesman Problem," *Bell. Syst. Tech. J.*, 44, 2245–2269 (1965).

[14] S. Lin and B. Kernighan, "An Effective Heuristic Algorithm for the Traveling Salesman Problem," *Oper. Res.*, 21, 498–516 (1973).

[15] P. Miliotis, "Combining Cutting-Plane and Branch and Bound Methods to Solve Integer Programming Problems," London School of Economics working paper (1977).

[16] P. Miliotis, "Integer Programming Approaches to the Traveling Salesman Problem," *Math. Program.*, 10, 367–378 (1976).

[17] U. R. Rau, private communication.

[18] G. T. Ross and R. M. Soland, "A Branch and Bound Algorithm for the Generalized Assignment Problem," *Math. Program.*, 8, 91–105 (1975).

[19] R. A. Russell, "An Effective Heuristic for the M-Tour Traveling Salesman Problem with Some Side Conditions," *Oper. Res.*, 25(3), 517–524 (1977).

[20] K. A. Shuster, and D. A. Schur, "A Heuristic Approach to Routing Solid Waste Collection Vehicles," U.S. Environmental Protection Agency (1972).

[21] M. Tyagi, "A Practical Method for the Truck Dispatching Problem," *J. Oper. Res. Soc. Jpn.*, 10, 76–92 (1968).

[22] P. Yellow, "A Computational Modification to the Savings Method of Vehicle Scheduling," *Oper. Res. Q.*, 21, 281 (1970).