The Vehicle Routing Problem with Multiple Trips

Abstract

Non-asset based third-party logistics providers manage fleets of leased vehicles for their customers. Due to high leasing costs, minimizing the number of vehicles employed in daily operations is of primary concern to such firms. Consequently, routings that involve fewer vehicles and potentially longer travel times are favored over routings that involve additional vehicles. Constructing routes that meet such criteria gives rise to *the vehicle routing problem with multiple trips* (VRPM). Unlike the classical *vehicle routing problem*, the VRPM permits vehicles to make multiple trips to and from a central depot within a given time duration. We present a heuristic solution methodology for the VRPM. In contrast to previous research on the VRPM, our approach explicitly combines vehicle routes within the route duration limit. We present preliminary results of our algorithm on benchmark data sets and outline plans to improve upon these solutions.

1. Introduction

Non-asset based third-party logistics providers manage fleets of leased vehicles for their customers. Such firms incur a significant cost for each vehicle leased. Consequently, minimizing the number of vehicles employed in daily operations is a primary concern. Furthermore, routings that involve fewer vehicles and potentially longer travel times are favored over routings that involve additional vehicles. Research aimed to address this and similar routing situations has attempted to solve what has become known as the *vehicle routing problem*.

The classical vehicle routing problem (VRP) makes deliveries of quantity q_i to each customer $i \in C$ $(i = 1, \ldots, n)$ from a central depot 0 using K independent delivery vehicles of identical capacity Q. The objective of the VRP is hierarchical – first the number of vehicles is minimized then the distance traveled is minimized. Additionally, the traditional VRP assumes a one-to-one correspondence between vehicles and routes (i.e., each vehicle is assigned to exactly one route). If the one-to-one assumption is relaxed, one obtains a variant of the VRP known as the vehicle routing problem with multiple trips per vehicle (VRPM). In other words, unlike the standard VRP, the VRPM allows vehicles to traverse multiple routes to and from the depot within a given time duration.

The flexibility to consider route design that allows vehicles to traverse multiple routes within a specific time period (e.g., a day) is beneficial when it is possible to satisfy customer demand with fewer vehicles (but perhaps the same number or more routes) than a standard VRP solution. As is the case with third-party non-asset based logistics providers, any reduction in fleet size represents a significant cost savings and far outweighs the additional travel time that may be required to meet customer demand with fewer vehicles. Consequently, these logistics companies stand to benefit from VRPM methodology.

In addition to the wide applicability and potential benefits of VRPM solutions, we observe for some data sets the route structures of VRPM solutions differ significantly from standard VRP route structures. We believe that this observation has algorithmic implications beyond the typical VRP considerations. To illustrate this difference, we repeat an example given by Olivera and Viera (2004). The best known solution to CMT-1 (a benchmark VRP problem consisting of 50 customers proposed by Christofides et al. (1979)) is depicted in Figure 1a and consists of five routes of durations of 98.45, 99.25, 99.33, 109.06, and 118.52 (a total cost of 524.61). Consider a VRPM instance over the same customers, with the same vehicle capacity, but restrict the number of vehicles to four and impose a route duration limit of 144. As shown by Olivera and Viera (2004), a feasible solution to the VRPM problem cannot be constructed by using the routes of the VRP solution. A feasible solution to this instance exists, however, and is depicted in Figure 1b. The solution consists of five routes of durations of 57.82, 82.12, 122.80, 141.76, and 141.79 (a total cost of 546.29), the first two routes being assigned to the same vehicle and the remaining three assigned each to a different vehicle. The structure of the VRPM solution in Figure 1b is noticeably different than the structure of the VRP solution in Figure 1a. In particular, the structure of the VRPM solution is characterized by embedded and intersecting routes, whereas the VRP solution structure consists entirely of non-overlapping routes. In general, embedded and intersecting route structures are more abundant in VRPM solutions than in VRP solutions.

Such a difference in route structures warrants further investigation beyond current VRP methods. The solution methodology presented in this paper takes advantage of these differences. More specifically, we construct embedded and intersecting routes which then serve as a basis for route linking (combining routes to facilitate multiple trips). Our approach, which exploits VRPM route structure and implements a procedure to explicitly link routes, is fundamentally different from the majority of algorithms presented in the literature to solve the VRPM.

Relevant VRPM literature includes work by Fleischmann (1990), Taillard et al. (1996), Golden et al. (1997), Brandão and Mercer (1997, 1998), Zhao et al. (2002), Petch and Salhi (2004), Olivera and Viera (2004), Campbell and Savelsbergh (2004), and Azi et al. (2006). While each of these authors has made contributions with respect to computational results, their methods are strikingly similar. In general, each algorithm first constructs a set of feasible VRP routes. Using a *best-fit decreasing* (BFD) bin-packing heuristic (Martello and Toth, 1990), these routes are then assigned to vehicles. The BFD heuristic first requires routes to be sorted in decreasing order by length. Routes are then assigned to a vehicle until a route duration limit (bin size) is exceeded. The remaining routes are then assigned to subsequent vehicles in a similar fashion until all routes have been assigned to a vehicle.

Although this method has proven successful, the routes created are those that minimize distance in a classical VRP. As illustrated in the above example, such routes may not be suited for a solution which employs route linking. Moreover, the vast majority of these algorithms approach the task of linking routes with a bin-packing heuristic. We hypothesize that a more careful inspection of the methodology employed in the route linking procedure is useful.

The remainder of the paper is structured as follows. In Section 2 we present a heuristic solution methodology for the VRPM. Computational results of our heuristic are given in Section 3. Concluding remarks and directions for future research are provided in Section 4.

2. VRPM Solution Methodology

Our initial efforts to solve the VRPM focused on obtaining optimal solutions via an exact solution methodology. More specifically, we formulated the VRPM as a mathematical program and employed standard branch and bound techniques to obtain optimal solutions. Such an approach, however, was only viable for small problem instances (10 or fewer customers). As the number of customers increases, the number of feasible solutions increases exponentially, thus requiring large amounts of time and computer memory to obtain optimal solutions. Due to these limitations, and the fact that many real world problem instances are very large (hundreds of customers), we focused on developing a heuristic solution methodology. Such methods are designed to produce *good* (but not necessarily optimal) solutions in reasonable computing times.

Our heuristic solution approach consists of three main phases: *route generation*, *route linking*, and *route selection*. In the route generation phase, feasible routes are constructed via a *petal* heuristic. A large number of feasible route combinations are then generated in the route linking phase. Finally, a subset of the routes generated in the previous phases is chosen in the route selection phase. A description of each phase is given in the following sections.

2.1 Route Generation

In the route generation phase, feasible routes are constructed in a fashion similar to the *sweep* and *petal* methods employed in many VRP heuristics (Gillett and Miller, 1974; Foster and Ryan, 1976; Ryan et al., 1993; Renaud et al., 1996). Such heuristics generate routes resembling the petals of a flower, hence the name (see Figure 1, for example). We begin by ordering customers in a radial order $(1, \ldots, n)$ with respect to the depot. We define $S_{i,j}$ to be a routing of the set of customers beginning with customer i and ending with customer j in the radial order. We initialize route construction with a route from the depot, to customer 1, and back to the depot ($S_{1,1}$). This route is then checked for feasibility with respect to vehicle capacity and route duration. If $S_{1,1}$ is feasible, it is added to the routes generated in this phase (otherwise, the VRPM is infeasible because the demand of customer 1 cannot be met). Route $S_{1,2}$ is then checked for feasibility. The process of constructing routes of the form $S_{1,j}$ continues until such a route is found to be infeasible. Once this occurs, the entire process repeats by beginning with customer 2 (i.e., routes of the form $S_{2,j}$ are considered). The procedure terminates after considering routes of the form $S_{n,j}$. The result of such an approach is a set of routes resembling petals.

It should be noted that the procedure outlined above generates non-overlapping routes and does not construct the embedded and intersecting route structures discussed in Section 1. The desired route structures can be obtained, however, if the ordering of customers described above is non-radial. We discuss this concept further in Section 3.1.

2.2 Route Linking

The route linking phase combines routes constructed in the previous phase such that the resulting linked routes do not exceed the overall route duration limit. Our approach to route linking (described below) constitutes a unique solution methodology for the VRPM. As described in Section 1, methods to date predominantly employ a bin-packing heuristic as the route linking mechanism. In particular, the routes to be linked by these methods are created via VRP route construction heuristics which seek to minimize the overall travel time. We anticipate that our computational results will demonstrate that route linking via the procedure outlined below is a comparable alternative to bin-packing (see Section 3 for preliminary results). Moreover, the methodology we employ exploits the characteristics observed in VRPM route structures.

The task of linking routes becomes increasingly difficult as the number of routes constructed in the route generation phase increases. More specifically, the number of potential route combinations increases exponentially with the number of routes. Consequently, an enumerative approach to route linking is not realistic. To reduce the number of route links generated in the route linking phase, we make use of two observations relating to the nature of the routes to be linked. First, we restrict route linking based on the load of the routes to be linked. More specifically, we do not link two routes whose sum capacity is less than the capacity of the vehicle (Rule 1). Zhao et al. (2002) formalize Rule 1 as a proposition and show through a direct application of the triangle inequality that linking such routes adds unnecessary additional distance.

The second rule employed is based on intuition. Vehicles return to the depot for two reasons: vehicle load is near vehicle capacity or route length is near the route duration limit (or both). In the latter case route linking is likely infeasible due to tight time constraints. In the former case, where a vehicle's load is binding and route length is not, route linking is viable. Based on this intuition, we propose the following rule for route linking: a series of r linked routes consists of at least $r-1$ *full* routes and at most 1 *non-full* route (Rule 2). We define a *full* route as a route that can accommodate no more than b additional customers without exceeding vehicle capacity or the route duration limit. A *non-full* route is a route that is not *full*. The choice of parameter b determines the number of *full* routes and therefore influences the number of potential route links.

Both Rules 1 and 2 are implemented in the route linking procedure. For the scope of this paper we provide only a brief description of this algorithm. The procedure is initialized by partitioning the routes from the route generation phase into two sets – full and non-full – based on a supplied value of b. The remainder of the algorithm is an iterative procedure consisting of two main parts. In the first part, full routes are linked with other full routes until additional route links are infeasible in terms of route duration or capacity. In accordance with Rule 2, the second portion of the iterative procedure creates route links containing at most one non-full route linked with the previously generated series of full routes. Throughout the iterative procedure, routes are only linked if they contain no common customers and if they meet the conditions of Rule 1. The result of this algorithm is a large number of linked routes. The task of selecting routes is addressed in the next section.

2.3 Route selection

In this section, we outline the methodology employed to select a subset of the linked routes generated in the route linking and rotue generation phases. We formulate the route selection problem as a set covering problem (SCP). In the context of the VRPM, the SCP calls for a minimum cost subset of routes such that each customer is visited. Similar to the mathematical program we developed for the VRPM (see Section 2), the SCP formulation can be solved exactly only for small problem instances. We therefore employ the Lagrangian-based heuristic proposed by Caprara et al. (1999) to solve the SCP and select a good subset of routes. The authors show their algorithm to be effective on very large problem instances. To date, we have employed this algorithm to obtain solutions to SCP instances with 75 customers and more than 800,000 routes.

3. Computational Results

In this section we present preliminary results of the heuristic solution methodology described in Section 2. In particular, we tested our algorithm on 50 of the benchmark data sets of Taillard et al. (1996). These VRPM instances are based on VRP instances 1-4 proposed by Christofides et al. (1979) (denoted CMT). The VRPM instances are obtained by constraining the number of vehicles (K) and by imposing various route duration limits. For each value of K, route duration limits $L_1 = [1.05z^*/K]$ and $L_2 = [1.10z^*/K]$ are proposed, where z^* is the best known solution value for the VRP instance and $[\cdot]$ denotes rounding to the nearest integer.

The preliminary results are summarized in Table 1. The columns labeled "Problem" denote the VRPM problem instance, the number of customers (n), and the best known VRP solution value (z^*). Columns labeled "K" indicate the maximum number of vehicles allowed for the problem instance and columns labeled "L" denote the corresponding route duration limit. Columns labeled "O & V" provide the recent results of Olivera and Viera (2004) on these problem instances. The numbers in these columns are the total distance traveled for K vehicles. An entry of "N/A" in these columns indicates that a feasible solution was not found. In other words, no solutions were found that meet customer demand while adhering to the route duration limit L and using at most K vehicles. Finally, columns labeled "Preliminary Results" display the best solutions obtained by the methodology presented in Section 2. These columns show the number of vehicles required and the total distance traveled by those vehicles.

Of the 50 problem instances presented in Table 1, our algorithm obtains feasible solutions in three cases: CMT-1 with $K = 3$ and $L = 192$, CMT-2 with $K = 6$ and $L = 153$, and CMT-3 with $K = 4$ and $L = 227$. Solutions are feasible because they employ K vehicles. In the second case, our solution is better than that obtained by Olivera and Viera (2004). In the remaining 47 problem instances, however, our algorithm was unable to obtain feasible solutions. This suggests that the routes created in the route generation phase were not the correct routes. In the following section we discuss a method aimed to create the desired route structures.

3.1 Next Steps

To improve upon the results presented in Section 3, we propose a modification of our current strategy. In the discussion that follows we identify two factors contributing to the low quality of solutions obtained thus far and propose approaches to overcome them.

As mentioned in the previous section, current results suggest that the routes created in the route generation phase are not the correct routes. More specifically, the algorithm presented in Section 2.1 does not produce the embedded and intersecting route structures observed in many VRPM solutions (see Section 1). To overcome this obstacle, we draw on an observation made by Ryan et al. (1993): optimal VRP solutions can be obtained via the petal generation scheme of Gillett and Miller (1974) with a non-radial ordering of the customers. Although Ryan et al. make this observation, we are unaware of any research conducted to this end. The method we propose for solving the VRPM takes advantage of this observation. Begin with a radial ordering of the customers and apply the three-phase procedure of route generation, route linking, and route selection to yield an initial VRPM solution. Perturb the customer ordering and repeat the three-phase procedure, updating the solution if a better one is found. In general, we intend to implement a local search for the perturbation of the radial order. We anticipate that perturbing the radial order within an appropriate neighborhood will result in the embedded and intersecting route structures found in many VRPM solutions.

A key ingredient for the success of this strategy will be the definition of the neighborhood over which the local search will be conducted. While we have yet to finalize this definition, we have observed that the size of the neighborhood can be constrained based on the geographic proximity of the customers. For example, a non-radial ordering that puts customers separated by long distances close to one another in the non-radial order would never be considered. On the other hand, small perturbations of the original radial order are likely to yield favorable results.

An additional factor contributing to the current results is the number of routes created during the route linking phase. Even for small problem instances (50 customers), the number of route links can exceed 1,000,000. Selecting a good subset of routes from such a large number of potential solutions is prohibitive both in terms of computation time and solution quality. We propose two courses of action to overcome this obstacle. First, we intend to reduce the number of routes created in the route linking phase by imposing additional rules in the linking procedure. In Section 2.2 we present two such rules, but we conjecture that other rules will further reduce the number of route links. For example, it is unlikely that routes containing only one customer will be included in optimal VRPM solutions. Eliminating these singleton routes from the route linking procedure should significantly reduce the total number routes generated.

In addition to decreasing the number of routes created in the route linking phase, solution quality may be improved by modifying the SCP heuristic employed in the route selection phase. While the SCP heuristic of Caprara et al. (1999) has proven effective on large scale problems, the structure of our VRPM problems may necessitate adjustments. We believe that these changes will improve the performance of the SCP heuristic, thereby improving the quality of the VRPM solutions.

4. Conclusions & Future Work

We have presented a heuristic approach to solve the VRPM. The proposed heuristic generates routes via the sweep heuristic of Gillett and Miller (1974), explicitly links these routes, and then selects a subset of routes via the SCP heuristic of Caprara et al. (1999). In contrast to other approaches presented in the VRPM literature, our approach exploits VRPM route structure and explicitly links routes. Preliminary results reveal two problems. First, the proposed heuristic is not generating the correct set of routes. Second, the number of routes generated in the route linking phase is large. As discussed in Section 3.1, we propose a local search of the radial order and a refinement of Caprara et al.'s SCP heuristic to overcome these challenges.

As we overcome these obstacles, we hope to gain more insight into when multiple use of vehicles is beneficial. In particular, we hope to develop metrics that gauge the potential of route linking for a given data set. The managerial implications of such measure(s) and the insight they may provide in solution procedures will likely be valuable. To date, this issue has only been minorly addressed by Zhao et al. (2002). Therefore, we believe we can make a significant contribution in this area.

In addition to addressing this issue, future work may incorporate stochastic demand and/or travel time in the VRPM. We anticipate that the results obtained in the current analysis will be valuable in solving a stochastic version of the VRPM.

References

Azi, N., M. Gendreau, and J. Potvin (2006). An exact algorithm for a single-vehicle routing problem with time windows and multiple routes. *European Journal of Operational Research 178*, 755–766.

Brandão, J. and A. Mercer (1997). A tabu search algorithm for the multi-trip vehicle routing and scheduling problem. *European Journal of Operational Research 100*, 180–191.

Brandão, J. and A. Mercer (1998). The multi-trip vehicle routing problem. *Journal of the Operational Research Society 49*, 799–805.

Campbell, A. and M. Savelsbergh (2004). Efficient insertion heuristics for vehicle routing and scheduling problems. *Transportation Science 38*(3), 369–378.

Caprara, A., M. Fischetti, and P. Toth (1999). A heuristic method for the set covering problem. *Operations Research 47*(5), 730–743.

Christofides, N., A. Mingozzi, and P. Toth (1979). *Combinatorial Optimization*, Chapter The vehicle routing problem, pp. 431–448. Chichester, UK: Wiley.

Fleischmann, B. (1990). The vehicle routing problem with multiple use of vehicles. Working paper, Fachbereich Wirtschaftswissenschaften, Universität Hamburg.

Foster, B. and D. Ryan (1976). An integer programming approach to the vehicle scheduling problem. *Operational Research Quarterly 27*(2), 367–384.

Gillett, B. and L. Miller (1974). A heuristic algorithm for the vehicle-dispatch problem. *Operations Research 22*(2), 340–349.

Golden, B., G. Laporte, and E. Taillard (1997). An adaptive memory heuristic for a class of vehicle routing problems with minmax objective. *Computers and Operations Research 24*(5), 445–452.

Martello, S. and P. Toth (1990). *Knapsack Problems*. Chichester, UK: Wiley.

Olivera, A. and O. Viera (2004). Adaptive memroy programming for the vehicle routing problem with multiple trips. Submitted for publication.

Petch, R. and S. Salhi (2004). A multi-phase constructive heuristic for the vehicle routing problem with multiple trips. *Discrete Applied Mathematics 133*, 69–92.

Renaud, J., F. Boctor, and G. Laporte (1996). An improved petal heuristic for the vehicle routeing problem. *Journal of the Operational Research Society 47*, 329–336.

Ryan, D., C. Hjorring, and F. Glover (1993). Extensions of the petal method for vehicle routeing. *Journal of the Operational Research Society 44*(3), 289–296.

Taillard, E., G. Laporte, and M. Gendreau (1996). Vehicle routeing with multiple use of vehicles. *Journal of the Operational Research Society 47*, 1065–1070.

Zhao, Q., S. Wang, K. Lai, and G. Xia (2002). A vehicle routing problem with multiple use of vehicles. *Advanced Modeling and Optimization 4*(3), 21–40.

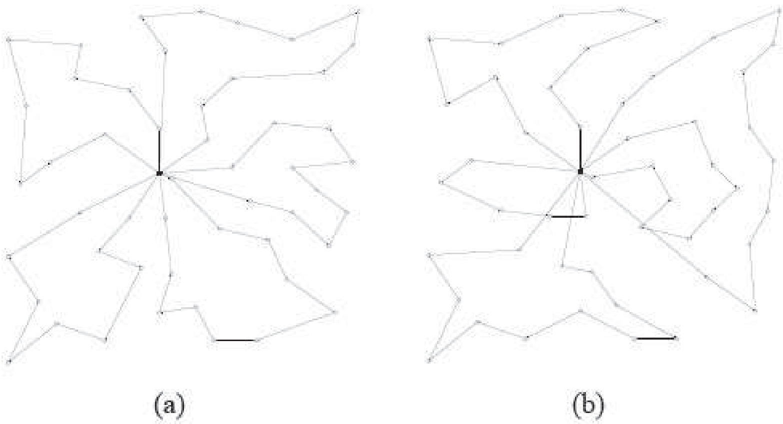

Figure 1: Best known VRP (a) and VRPM (b) solutions for CMT-1

Table 1: Preliminary Results

			O & V	Preliminary Results					O & V	Preliminary Results	
Problem	K	L	**Cost**	**# Veh.**	**Cost**	**Problem**	K	L	**Cost**	**# Veh.**	**Cost**
CMT-1	1	551	525	2	782	CMT-2	1	877	836	3	1839
$n = 50$	2	275	533	3	532	$n = 75$	2	439	843	4	1594
$z^* = 524.61$	3	184	N/A	4	553	$z^* = 835.26$	3	292	846	5	1410
	4	138	N/A	5	532		4	219	839	6	1254
	1	577	525	2	866		5	175	853	6	879
	2	289	530	3	532		6	146	N/A	7	882
	3	192	553	3	554		7	125	N/A	8	872
	4	194	547	5	532		1	919	844	3	2452
CMT-4	1	1080	1033	7	2444		2	459	841	4	1755
$n = 150$	2	540	1037	7	2409		3	306	837	5	902
$z^* = 1028.42$	3	360	1035	7	2134		4	230	836	5	890
	4	270	1036	11	1733		5	184	844	6	878
	5	216	1033	9	1174		6	153	875	6	866
	6	180	1058	12	1960		7	131	873	8	865
	7	154	N/A	13	1313	CMT-3	1	867	831	3	1186
	8	135	1065	10	1110	$n = 100$	2	434	834	3	1288
	1	1131	1042	7	2476	$z^* = 826.14$	3	289	831	5	971
	2	566	1047	7	3073		4	217	833	6	865
	3	377	1039	8	2524		5	173	851	6	855
	4	283	1039	11	1983		6	145	840	7	845
	5	226	1044	8	1134		1	909	830	3	1186
	6	189	1033	12	1817		2	454	830	3	1237
	7	162	1063	14	1964		3	303	829	5	894
	8	141	1065	10	1094		4	227	826	4	872
							5	182	833	6	854
							6	151	842	7	845