

Arc routing vehicle routing problems with vehicle/site dependencies

J. Sniezek and L. Bodin
College of Business and Management
University of Maryland
College Park, MD 20742, U.S.A.
Tel: (301) 405-2210, (301) 596-3100.
Fax: (410) 964-2688.

Abstract

In this paper, an algorithm is discussed for solving the Vehicle Arc Routing Problem when there exists the constraints of Vehicle/Site Dependencies. The traditional Capacitated Arc Routing Problem assumes a homogeneous fleet mix of vehicles available to service the arcs in the network. In essence, any vehicle in the vehicle fleet mix can travel or service any of the arcs in the network and all vehicles in the vehicle fleet mix have the same capacity. In actuality, most real world applications of the Capacitated Arc Routing Problems do not use a homogeneous fleet mix. Solid waste collection tends to utilize a heterogeneous rather than a homogeneous fleet mix. A city may have many different sized trash vehicles which pick up the garbage on the streets. As a result of the vehicles being of different sizes, the capacities of the vehicles will be different. More important is that the different size vehicles may not be able to travel on the same streets. For example, larger vehicles may not be able to travel on small alleys in the middle of the city. This second issue, where certain vehicles can not travel on certain streets is what is referred to as Vehicle/Site Dependencies. This paper analyzes and presents a practical solution to the Capacitated Arc Routing Problem while satisfying Vehicle/Site Dependencies with a heterogeneous fleet mix.

1 INTRODUCTION

Residential sanitation problems can be considered a special case of arc routing and partitioning problems. These problems are named neighborhood (or saturated) routing and scheduling problems. In a neighborhood routing and scheduling problem, most of the street segments in a geographic region require service. There is much literature on solving a neighborhood routing and scheduling problem over a

street network. The **RouteSmart™** routing and scheduling system for scheduling sanitation vehicles was developed by Bowne Distinct LTD of Columbia, MD, USA to solve neighborhood routing and scheduling problems over a street network. Several sanitation companies in the United States are using **RouteSmart™** for scheduling their sanitation vehicles. Traditionally, the solution of a neighborhood routing and scheduling problem assumes a homogeneous set of vehicles and a homogeneous street network used for travel. All vehicles are assumed to have the same capacity and all vehicles can travel on all streets in the network.

An additional complication many sanitation companies encounter is the problem of Vehicle/Site Dependencies. In this complication there exists a fleet of vehicles where each vehicles may be different in terms of size, capacity, etc. In addition, many of the vehicles in the fleet are restricted, due to size or weight constraints, from traveling on or servicing certain streets in the network. In the neighborhood routing problem for sanitation vehicles, a Vehicle/Site Dependency is when a particular street segment can only be handled by a particular subset of the available vehicles. We have developed an algorithm to handle Vehicle/Site Dependencies and this algorithm has been implemented within **RouteSmart™**.

The traditional neighborhood routing and scheduling problem is generally solved in three steps. In Step 1, a workload estimation procedure is used to determine either the number of vehicles required to service the area under consideration, given a target time for each vehicle, or a vehicle target time is determined, given a specified number of vehicles. In Step 2, the area over which the routing is to be performed is broken down in partitions where each partition represents the region to be serviced by a particular vehicle. In Step 3, an Euler circuit problem is solved over each of the partitions to give a minimum deadhead travel time path that services all of the required streets in the partition.

For sanitation scheduling, this problem can have the following complications:

1. The vehicle can become filled so that trips to the recycling facility, transshipment point or landfill could be required.
2. Some of the streets in a partition can be directed or 1-way; other streets in the partition can be bi-directional so that two traversals of the street segment become necessary; other streets in the partition can be meandered so that only one traversal of the street segment is required but this traversal can begin at either one of the intersections and end at the other intersection. As such, the determination of the travel path is significantly more complicated than the traditional directed or undirected Euler circuit problem.
3. In both Steps 2 and 3, the service network, which is the network consisting of the set of street segments to be serviced, can be disconnected. Therefore, the partitioning and travel path algorithms must be implemented considering the fact that the underlying service network can be disconnected.
4. Turn restrictions can exist at specific intersections. For example, large garbage trucks do not make U-turns unless absolutely necessary. The need to restrict U-turns and other undesirable travel patterns complicates the minimum deadhead Euler circuit problem of step 3.

In addition to the above complications, an additional complication now being addressed by **RouteSmart™** is the constraint of Vehicle/Site Dependencies. The vehicles

being used may differ in terms of size and capacity. Also, each particular street segment can only be serviced by a particular subset of the vehicles in the fleet mix. These constraints can occur because certain streets may be too narrow for certain vehicles to drive on or there may be a constraint on the weight that the street segment can handle. Thus, only vehicles in a vehicle type that meets certain weight/size restrictions can be used to service a street segment with a Vehicle/Site Dependency constraint.

A particular case of the Vehicle/Site Dependency problem is the arc routing problem with multiple vehicle types. In this problem, there is a non-homogeneous fleet mix such that the vehicles may have different capacities, but there are no Vehicle/Site Dependencies on any of the street segments. All street segments can be serviced by all of the vehicle types. The analysis being presented here applies to both the multiple vehicle type arc routing problem with Vehicle/Site Dependencies and the arc routing problem without Vehicle/Site Dependencies.

To solve this neighborhood routing problem with Vehicle/Site Dependencies adds significant complications to Steps 1 and 2 above. We have developed an algorithm, and have implemented this algorithm within **RouteSmart™**.

2 DESCRIPTION OF THE VEHICLE/SITE DEPENDENCY ROUTING PROBLEM

The Vehicle/Site Dependency routing problem being studied starts with a street network composed of individual street segments. A subset of the street segments in the network must be serviced by a vehicle from a vehicle fleet mix. Each vehicle in the vehicle fleet mix belongs to a specific vehicle class. Each street segment in the network can have restrictions as to which vehicle classes can service the street segment and which vehicle classes can drive the street segment without servicing it. Each vehicle in a given vehicle class will have a certain vehicle capacity. If this capacity is reached, a trip to a dump site is required.

The street network is represented as a set of arcs. The objective is to create subsets of the arcs, called partitions, where each partition is assigned to a vehicle in the vehicle fleet mix. An Euler circuit is then formed for each of the partitions. Every street segment which requires service must be in one of the partitions. Each partition and corresponding circuit represents the set of streets each vehicle is responsible for servicing and the line of travel the vehicle will follow when servicing those streets.

2.1 Attributes of the Arcs representing the Street Network

Each street segment in the street network is represented as an arc. The following attributes must be associated with each arc in the set of arcs to be partitioned.

1. Whether or not the arc requires service. Not all streets in a city have trash to pick up and thus not all arcs in the set of arcs to be partitioned require service. Those arcs which do not require service may still be used to travel between arcs which do require service.

2. The volume or weight associated with the arc, assuming the arc requires service. This volume or weight will represent the expected amount of trash to be picked up on that street segment. This metric is used to keep track of the total volume or weight a vehicle has picked up. This is required to account for the time associated with traveling to a dump or transfer site when the vehicle has filled itself to capacity.
3. The time it takes to service the arc assuming service is required.
4. The time it takes to travel the arc when not servicing the arc.
5. Every arc, whether it requires service or not, must contain a list of all vehicle classes that can travel on the arc when not servicing the arc. Some streets are too small for certain vehicle classes to travel on and thus the corresponding arcs can not be traveled on by any vehicles in those vehicle classes.
6. Every arc requiring service must contain a list of all vehicle classes that can service the arc. It is assumed that for every arc, this list is a subset of the list of vehicle classes that can travel the arc without servicing it.
7. The type of service associated with each service arc. The service required may be only on the right side of the street, only on the left side or on both sides of the street.
8. Is the arc directed or undirected. If a street can have the trash picked up on both sides with a single traversal of the street (meander) then the corresponding service arc will be undirected. If the street can not be meandered, then the corresponding service arcs (one for each side of the street) will be directed to reflect traffic orientation.

2.2 Attributes of the Vehicle Class

The following attributes must be associated with each vehicle class.

1. The total number of vehicles available in the vehicle class. This serves as an upper bound for the number of vehicles from each vehicle class that can be utilized.
2. The maximum weight or volume that a vehicle in the class can hold. Once a vehicle in the vehicle class has collected this maximum weight or volume, it is required to make a trip to a dump site before continuing service.
3. The target length of workday for that vehicle class plus a soft upper and lower bound for the target time.
4. All overhead time associated with the vehicle class.
5. The dumptime associated with the vehicle class. This time is the time it takes to empty the vehicle once its capacity has been reached.

2.3 Other Constraints

Other constraints associated with the Vehicle/Site Dependency Routing Problem are as follows:

1. A priority or preference list of vehicle classes is required. This will list in descending order of priority, a vehicle class, and the minimum and maximum number of vehicles to use from the vehicle class at that priority level. For example, the smallest

vehicles may be undesirable because they require several trips to the dump site. The smaller vehicles could be placed at the bottom of the preference list and thus would only be used if required due to Vehicle/Site Dependent constraints on the streets. A vehicle class may appear more than once in the preference list.

2. A single depot location is used as the starting and ending point for all vehicles.
3. A single landfill site is used where all vehicles must go when their capacity has been reached in order to empty the vehicle.
4. All vehicles must return to the landfill at the end of the route to empty the vehicle.

3 THE ALGORITHM

An algorithm has been developed which provides a practical solution to the Vehicle Arc Routing Problem while satisfying the complications of Vehicle/Site Dependencies with a non-homogeneous fleet mix. This algorithm has been implemented using **RouteSmart™** and is currently being used by a large sanitation company in the United States. The steps of the algorithm have been broken into 3 sections. Each section corresponds to the three steps generally used to solve the Vehicle Arc Routing Problem without Vehicle/Site Dependencies, as discussed in the introduction of the paper. The steps of the algorithm used to solve the Vehicle Arc Routing Problem with Vehicle/Site Dependencies are as follows:

- 1a) Determine an initial fleet mix estimate. The fleet mix is the number of vehicles from each vehicle class which will be used to solve the problem. This fleet mix is determined based upon the given fleet mix preference list, the total service time associated with all street segments requiring service, the total volume associated with all street segments requiring service, the relative connectivity of the street segments requiring service, and the time constraints on route length and dump time. This estimate will become the actual fleet mix for the first iteration of the algorithm.
- 1b) Order all vehicles in the fleet mix to be used based upon their vehicle class. The ordering is done from the least restrictive to the most restrictive vehicle class.
- 2a) At this point, only consider the street segments which have yet to be assigned to a partition and which must be serviced by a vehicle class no larger than the i 'th least restrictive vehicle class in the fleet mix. (i = the number of times we have reached step 2a in this iteration) Thus the first time step 2a is reached, only consider the streets which must be serviced by the smallest (least restrictive) vehicle class in the fleet mix. Based upon these street segments, seed a subset of the feasible vehicle classes' remaining vehicles.
- 2b) Assign each of the unassigned street segments considered in 2a to one of the vehicles seeded in 2a. Partition balance and geographic compactness is considered during the assignment of street segments to partitions.
- 2c) Now considering all remaining unassigned street segments continue assigning street segments to the partitions formed in 2b until the partitions reach a certain percentage of maximum route time. Again, partition balance and geographic compactness is considered during the assignment of street segments to partitions.

- 2d) If there are still unassigned street segments which can not be serviced by each and every unassigned vehicle classes in the fleet mix, then return to step 2a.
- 2e) Seed all remaining vehicles in the fleet mix and then assign all street segments still not assigned to a partition to any feasible partition while still taking route balance and geographic compactness into consideration.
- 2f) Perform swapping of service streets between the grown partitions in an attempt to balance the total route time for each partition. At this point, the total route times are only estimates, as the actual travel path has yet to be solved for each partition.
- 3a) Solve a Chinese Postman Problem for each partition to determine actual route times for each partition. If all partitions are balanced within an upper and lower bound, then output the travel paths and statistics for each partition. The problem is finished.
- 3b) If the partitions are out of balance, perform a revision of workload estimation based upon the out of balance partitions that currently exist. It is possible to change the fleet mix at this point. For example, it may be determined that due to Vehicle/Site Dependencies, an additional vehicle from the smallest vehicle class may be required.
- 3c) Return to step 2a for a new iteration using the revised fleet mix determined in 3b.

4 CONCLUSIONS

The algorithm presented in this paper is a modification of the traditional methods for solving Vehicle Arc Routing Problems in order to handle the real-world constraints of Vehicle/Site Dependencies. This algorithm has been implemented in RouteSmart and is currently being used by a large sanitation company in the United States. The problems being solved using this algorithm generally contain 3 different vehicle classes and require anywhere from 10-20 vehicles per solution. The problems are being solved over a street network of approximately 20,000 street segments.

There is still much work that needs to be done in the area of Vehicle/Site Dependencies. The solution approach discussed in this paper is a first attempt at handling Vehicle/Site Dependencies in an Arc Routing Vehicle Routing problem.

REFERENCES

- There are numerous papers written on Vehicle Arc Routing and Scheduling without the consideration of Vehicle/Site Dependencies. Three excellent references are the following:
1. Bodin, L., B. L. Golden, A. Assad, and M. Ball (1983), "Routing and Scheduling of Vehicles and Crews: The State of the Art". Computers and Operations Research, 10(2), 63-211.
 2. Golden, B. L. and A. Assad, eds. (1986). Special Issue on Time Windows. American Journal of Mathematical and Management Analysis, 6 (3 and 4), 251-399.

3. Golden, B. L. and A. Assad (1988). Vehicle Routing: Methods and Studies. North-Holland, Amsterdam.

To the best of our knowledge, there are no known papers addressing Vehicle Arc Routing and Scheduling which consider Vehicle/Site Dependencies. There is the following reference on Vehicle Node Routing with Vehicle/Site Dependencies:

4. Nag, B., B. L. Golden, and A. Assad 1988, "Vehicle Routing With Site Dependencies", Vehicle Routing: Methods and Studies, Ed. by B. Golden and A. Assad, 149-159.