Belgian Journal of Operations Research, Statistics and Computer Science, Vol 31, n° 1-2

PUBLIC WASTE COLLECTION: A CASE STUDY

Ludo F. GELDERS
and
Dirk G. CATTRYSSE

Katholieke Universiteit Leuven
Afdeling Industrieel Beleid
Celestijnenlaan 300 A, B-3001 Leuven-Heverlee, Belgium

ABSTRACT

This paper deals with the waste collection in the N.E. area of Brussels. It summarizes a study carried out as a master's thesis project. Until now the collection scheme was based upon experience. Management felt that a more normative and systematic approach was needed. This paper discusses the modeling of the real-life problem based on the capacitated arc routing problem. The solution approach is based on the path scanning algorithm. The problem solver was coded in Pascal and linked with dBase-files which contain all information on the collection area. A reduction of approximately 15% in distance travelled was achieved.

1. PROBLEM DESCRIPTION.

The current problem deals with the waste collection of a set of 5 municipalities in the N.E. area of Brussels. Between 4 and 11 waste collection trucks have to serve these communities daily. On a yearly basis, these lorries total 360.000 km. These 360.000 km are divided into 60.000 km for the actual waste collection and 300.000 km for trips to the waste disposal area, which is located south of Brussels. The distance from the last collection point to the disposal area varies between 25 and 50 km.

Until now, the collection scheme was based upon experience. Each truck was assigned a set of streets to be served. In the current system, a truck crew is free to go home whenever the job is finished. This work organization relies upon the hypothesis that the work force will try to optimize its routing. Under current operations, the company employs 41 people, 34 being crew members of the trucks.

As always in garbage collection problems, the experience based work schemes take into account a very broad and complicated set of constraints, e.g. :

- one-way streets
- fluctuating quantities to be collected
- given collection frequency (twice per week)
- capacity limitation of trucks
- truck maintenance
- special collection services due to special events
- etc...etc...

Management felt however that a more normative and systematic approach might give some insight to do the job more efficient, through a better use of resources (decreasing the total travel distance, the number of trucks used, etc...). Moreover a need was felt to have an interactive decision tool which might provide quick answers to certain urgent problems (e.g. unavailability of a crew or a truck).

2. MODELING THE PROBLEM.

In literature one can find discussions on similar problems, e.g. : sanitation vehicle routing [2], vehicle routing for municipal waste collection [1] and routing electric meter readers [9].

The above waste collection problem can be modelled as the Capacitated Arc Routing Problem (CARP) : given an undirected network G(N,E,C) with arc demands $q_{ij} \geq 0$ for each arc (i,j) which must be satisfied by one of a fleet of vehicles of capacity W, find a number of cycles each of which passes through the depot (node 1) which satisfy demands at minimal total cost. The CARP can be formulated as follows (see Golden and Wong [7]) :

$$\min \sum_{i=1}^{N} \sum_{j=1}^{N} \sum_{k=1}^{K} c_{ij} \quad x_{ijk} \qquad (1)$$

$$\text{s.t.} \quad \sum_{j=1}^{N} x_{jik} - \sum_{j=1}^{N} x_{ijk} = 0 \qquad i = 1,..,N \quad k = 1,..,K \qquad (2)$$

$$\sum_{k=1}^{K} (l_{ijk} + l_{jik}) = \lceil q_{ij}/W \rceil \qquad (i,j) \in E \qquad (3)$$

$$x_{ijk} \geq l_{ijk} \qquad (i,j) \in E, \ k = 1,..,K \qquad (4)$$

$$\sum_{i=1}^{N} \sum_{j=1}^{N} l_{ijk} \, q_{ij} \leq W \qquad k = 1,..,K \qquad (5)$$

$$\sum_{i \in Q} \sum_{j \in Q} x_{ijk} - N^2 y_{1qk} \leq |Q| - 1$$

$$\sum_{i \in Q} \sum_{j \notin Q} x_{ijk} + y_{2qk} \geq 1 \qquad (6)$$

$$y_{1qk} + y_{2qk} \leq 1$$

$$y_{1qk}, y_{2qk}, x_{ijk}, l_{ijk} \in \{0,1\} \qquad (7)$$

for (6)–(7): $k = 1,..,K$; $q = 1,..,2^{N-1}-1$ and every nonempty subset Q of {2,3,..,N} where N = the number of nodes,

K = the number of available vehicles,

q_{ij} = the demand on arc (i,j),

W = the vehicle capacity ($W \geq \max q_{ij}$),

c_{ij} = the length of arc (i,j),

x_{ijk} = 1, if arc (i,j) is traversed by vehicle k, 0 otherwise,

l_{ijk} = 1, if vehicle k services arc (i,j), 0 otherwise,

$\lceil z \rceil$ = the smallest integer greater than or equal to z,

Q = subset of N with cardinality between 2 and $|N|$.

The objective function (1) seeks to minimize total distance travelled. Equations (2) ensure route continuity. Equations (3) state that each arc with positive demand is serviced exactly once. Equations (4) guarantee that arc (i,j) can be serviced by vehicle k only if it covers arc (i,j). Vehicle capacity is not violated on account of equations (5). Equations (6) prohibit the formation of (infeasible) subtours. Every index q corresponds to a set Q. Integrality restrictions are given in (7).

3. SELECTION OF SOLUTION METHODOLOGY.

Given the computational complexity of the CARP (which is an NP-hard problem (Golden and Wong [7])) it becomes necessary to apply approximate solution techniques or heuristics. Golden, De Armon and Baker [6] compare some heuristics developed to solve the CCPP (Capacitated Chinese Postman Problem). The CCPP is a special form of the CARP : $q_{ij} > 0$ for all i and j (instead of $q_{ij} \geq 0$). In general, 3 different approaches are available when dealing with multi-vehicle problems , i.e.,

- Cluster First-Route Second : the area is divided in subareas in such a way that every subarea can be serviced by one vehicle. For every subarea a Chinese Postman Problem (CPP) is solved.
- Route First-Cluster Second : first the problem is solved as a CPP. The complete cycle is then split in several routes which do not violate the capacity constraints of the vehicles and minimize the distance travelled.
- Route and Cluster Together : routing and clustering is done at the same time.

All three approaches have advantages and disadvantages. The first method creates smaller problems which are easy to solve and requires less data handling but yields suboptimal solutions and does not take the stochastic character of the demand into account. The second method has the advantage that the first phase remains the same even if demand varies. However a larger network has to be optimized and the relation between the tours and the depot (start node) and the different tours is gone (remember that the distance to the depot contributes the most to the total travelling distance). This method works rather well when every tour consists of only a few arcs [9]. This is not the case in this study. The third approach is well adapted to problems where a vast physical area contains a very large number of arcs. It allows for a good overall and integrated view of the problem. This third approach is chosen and three algorithms in that class are briefly discussed :

- Path Scanning [6] : the basic idea is to construct a tour at a time by adding arcs sequentially till the capacity is exhausted. Then the shortest return path to the depot is followed. The criteria for choosing an arc (i,j) are : the distance, c_{ij}, per unit remaining demand is minimized (a) or maximized (b), the distance from node j back to the depot is minimized (c) or maximized (d), if the vehicle is less than half-full maximize the distance (e); (a) looks for a large and quick payoff while (b) seeks to incur the larger expenses early; (c) tends to obtain shorter cycles; whereas (d) in general, yields longer cycles; (e) represents a hybrid approach. The set of cycles with smallest total distance is selected as outcome for this simple and rather fast algorithm.

- Construct and Strike [4, 6] : this algorithm repeatedly constructs feasible cycles and then strikes or removes them. The following steps are required :
  step 1 : construct a cycle fulfilling the capacity constraint of the vehicle.
  step 2 : set the demand of the arcs belonging to this cycle equal to 0.
  step 3 : repeat steps 1 and 2 till there are no arcs, to be serviced, left in the network.

This procedure has one drawback : how to construct good cycles ?

- Augment - Merge [3, 7] : this procedure merges several smaller cycles in larges ones :
  step 1 : start with as much cycles as arcs to be served.
  step 2 : starting with the largest cycle available, see if a demand arc on a smaller cycle can be serviced on a larger cycle (= AUGMENT).
  step 3 : evaluate the merging of any two cycles, subject to capacity constraints or additional restrictions. Merge the two cycles which yield the largest positive savings (= MERGE).
  step 4 : repeat step 3 until finished.

Computational experience done by Golden, De Armon and Baker [6] indicates that the augment-merge strategy yields the best results on average, but is much more complicated and time consuming than the two other types. The path scanning algorithm has the advantages of simplicity and minor CPU-time requirements. This algorithm can easily be adapted to all the necessary needs and criteria as required by the problem stated above. The construct and strike algorithm has the same advantages but yields results which are worse.
Since management needs an interactive decision tool which provides quick answers to urgent problems, an approach based on the path scanning algorithm is chosen. Moreover, a number of different criteria for choosing the next arc in every step of the algorithm exist (they are mentioned earlier). By simulating some scenarios it is found that the following combination of two criteria leads to the best results.
Given a provisional path (start node = 1) arrives in node i, we add the arc (i,j) satisfying one of the following criteria :
- if the truck load is less than 50 %, maximize the shortest distance from j to the end-node, distance of arc (i,j) included.
- if the truck load is 50 % or more, minimize the shortest distance from j to the end-node, distance of arc (i,j) included.

This combination leads to cycles with a reasonable length without augmenting markedly the distance from the last collection point back to the start or end (waste disposal) node.

4. IMPLEMENTATION ISSUES.

4.1 CLUSTERING.

Although the path scanning algorithm is a route and cluster together method, we will start dividing the total collection area in a number of sectors. These sectors are much larger than the clusters in the cluster first - route second approach. For a particular day, all lorries will be send to the same sector. Our major concern is to minimize the number of trips from and to the waste disposal node. In this way a lorry can take over a part of the tour of another one without driving a long distance (interaction between tours). Demand is stochastic and a vehicle can reach its maximum allowable load before finishing its cycle.
Adding up the collection amounts to obtain the total daily collection quantity per sector, levels out a large number of these stochastic elements. Nevertheless some time-depending parameters remain. There is a seasonal dependency because people produce more garbage in summer time than in winter. Also on the long term, the total amount of garbage to collect yearly increases. To start, we will define the sectors by aggregating the present sectors.

4.2 INPUT DATA.

It was an enormous task to collect all input data necessary for solving this problem. And as the accuracy of these data will influence to a large extent the accuracy of the solution, it is of major concern to keep these inputs up-to-date. So a well developed database structure had to be set up. The input data can be devided into three classes:

- data related to the network :
  - an adjacency list for every node
  - cost c_{ij} for every arc, expressed as:
    1. a collection cost (time)
    2. a driving cost (time to drive through the arc without collection)
  - quantity garbage q_{ij} to be collected for arc (i,j)
  - one-way streets or not
  - both sides of the streets can or can not be served simultaneously
  - collection arcs and arcs connecting one node to another without collecting garbage
  - start and end node for every sector.

- general data :
  - number of lorries available for the different types of garbage
  - maximum capacity (load) for every lorry and an average lorry capacity W as used in the program
  - as the density of garbage is not always the same, not weight but volume can be the binding constraint. We do not only need an average W but also a margin on W, to express the variability
  - seasonal coefficient on quantity of garbage produced.

- a number of parameters tested in the different scenarios to evaluate their sensitivity on the result.

The most important data in this model are the costs c_{ij} and the quantities q_{ij} related to every arc. c_{ij} can be expressed as the time necessary to serve an arc, with or without collection. The time can be derived by measuring the distance (l_{ij}) and multiplying it by the average velocity of a lorry. Distances of the arc were measured on detailed maps of the area and corrected with information from the drivers who noted down distance travelled. There are 2 velocities:

v_d = velocity with collection --> $t_{dij} = l_{ij} / v_d$

v_b = velocity without collection --> $t_{bij} = l_{ij} / v_b$

In the article of Bodin et al. [2] the velocity v_b is even devided into 4 classes depending on the types of streets.
The quantity q_{ij} is the most difficult element in gathering the data, because q_{ij} has a stochastic nature and a direct measurement (estimation) of q_{ij} is impossible for practical reasons.
q_{ij} was estimated in the following way. From historical data, total weakly collection volumes V per sector S will be determined.

$q_{ij} = f_{ij} * V$ with $0 \leq f_{ij} \leq 1$,

($f_{ij} = 0$ if arc (i,j) belongs to the sector S and has to be serviced by another sector S' or arc (i,j) does not belong to the sector S)

When garbage is collected in mini-containers, f_{ij} can be easily estimated as follows :

$$f_{ij} = \frac{\#\text{ containers for arc } (i,j)}{\#\text{ containers for sector S (arc } (i,j) \in S)}$$

It is clear that the costs c_{ij} can be estimated more accurately than the collection quantities q_{ij} per arc. Nevertheless, with regard to the accuracy of the data, following equation is very important to keep in mind :

$e_r(q) = e_r(q_{ij}) * n^{-1/2}$ with: q = collection quantity per trip
n = number of arcs in one trip
$e_r(X)$ = relative error on X

Analogously we have:

$e_r(l) = e_r(l_{ij}) * n^{-1/2}$ with: l = driving distance per trip
l_{ij} = driving distance per arc

Example: l = 20.000 m
l_{ij} = 200 m -----> $e_r(l) = e_r(l_{ij}) * 10$
$n = l/l_{ij} = 100$

This means that when a relative error of 5 % on l with a confidence interval of 95 % is required, a relative error on l_{ij} up to 50 % is allowed. As a consequence the way we estimated both distances and collection quantities per arc leads to rather accurate results when aggregated over a collection trip.

5. COMPUTATIONAL ASPECTS.

The software program can be separated into two parts:

- a general database program (in dBase III), to store and update all input data.
- a route generation program written in Pascal.

5.1 GENERAL DATABASE PROGRAM.

Every arc is defined by two nodes, the start- and end-node. The following information is given per arc :

- two costs : a collection cost and an empty cost (cost for driving through an arc without collecting garbage)
- capacity q_{ij} : quantity of garbage to collect for arc (i,j)
- type : 1 = directed arc
  2 = undirected arc

- ten fields S1, S2 .... S10, corresponding to ten sectors with each field containing one of the following codes :
  - 1 = a collection arc for the corresponding sector.
  - 2 = arc belongs to the sector, but no collection is necessary.
  - 3 = arc does not belong to the sector.

This is a general database, containing all the possible streets in the collection problem. From this data base we can easily select the arcs belonging to one or more sectors (1 ... 10), these arcs will be copied into a sector data base and used as input for the route generation program. Whenever there is a change in the street pattern or collection quantities, the general data base will be updated and new partial data bases per sector will be derived. Organizing the input structure in this way will never lead to data inconsistency.

5.2 THE ROUTE GENERATION PROGRAM.

The program is built in a modular way. The first module reads all the network data out of the sector data base and doubles the undirected arcs. The second module reads the general data related to the sector (start-node, end-node, collection quantity) and other data like number of lorries, their capacity, etc...
After the user has chosen an optimization criterion, one arc at a time will be selected in module 3 to generate the different trips. In every node the path scanning algorithm is used when one of the related arcs is not yet served. The Dijkstra [5] algorithm is used in all other cases, to find the shortest way to another unserved arc.
When lorry capacity is reached, the program also uses the Dijkstra algorithm to find the shortest way back to the end-point.
A new trip will start in the end- or start-node of the network and module 3 will again generate a new routing. New trips will be generated as long as unserved arcs in the network exist. After every trip, data with regard to collection quantity q and length of the trip (l) are stored.

5.3 FLEXIBILITY OF THE PROGRAM.

The program is built in such a way that it is very easy to generate a number of scenarios. Following options are possible :
- changing the area and shape of the sectors

- choosing a start- and end-point of a trip in one node or in two different nodes. In this way a lorry can start its first trip at the depot and end it at the waste collection area (from where the second trip starts)
- different criteria to execute the path scanning algorithm
- introduction of a seasonal coefficient to adapt collection quantities per arc over the year
- a lorry, close to the end-point of the network has two options. It can start a new cycle, or it can drive directly to the waste collection area. A preference for one of these options can be implemented by adding a parameter α. Instead of having a lorry capacity W, we will give the truck in the end-point a capacity αW. If $q < \alpha W$, the truck will start a new cycle. If $q > \alpha W$, the truck will go directly to the collection area.

  By giving α a number between 0 and 1 we can influence thoroughly the length and the number of trips
- simultaneous garbage collection on both sides of the street
- blind alleys are automatically added to an adjacent arc. In this way the network is simplified and empty driving time is minimized.

Elements not included in the program are : time constraints, holidays, rough garbage, traffic jams, right turns.

6. CONCLUSIONS

A combination of six current sectors was selected as a testing area. These sectors give a good representation of the whole collection area. Total weekly collection in the testing area is approximately 5000 tons.

6.1 INFLUENCE OF LORRY CAPACITY (W).

Initially W was fixed on 9.330 Kg. The total driving distance decreases by increasing W. From fig.1 we see that the decrease in driving distance is not continuous. The explanation is very simple. Whenever a lorry A reaches its capacity constraint W, it drives directly to the waste disposal area. Another lorry B (operating in the neighbourhood) will continue and serve the remaining collection quantity of the arc (i,j). Increasing W will only decrease total driving distance when arc (i,j) can be collected completely by lorry A.

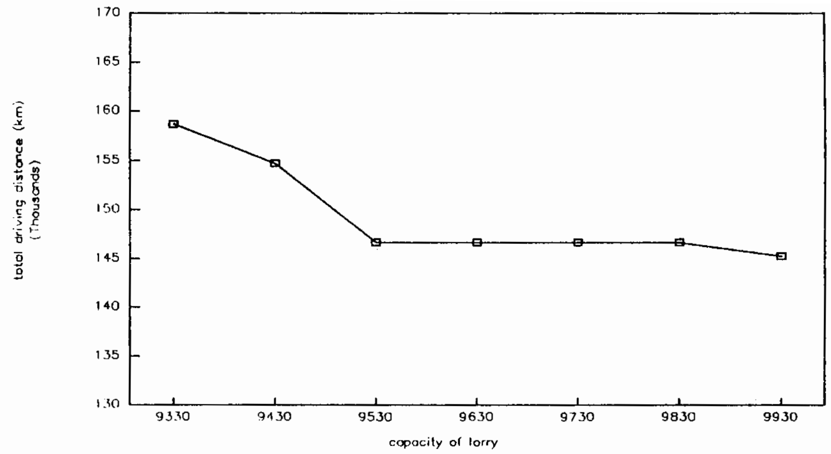

Fig.1 : Sensitivity of the lorry capacity on the total driving distance.

6.2 OPTIMAL CRITERION FOR OPEN AND CLOSED TRIPS.

A closed trip starts and ends in the same point of the network whereas, an open trip has a different start- and end-point. Intuitively it is clear that open trips give better results than closed trips for all possible criteria. However for combination tours, open as well as closed trips, using one and the same criterion for both types of trips gives better results than using two different criteria. This was a rather remarkable result we found from the simulations of the testing area.

6.3 INFLUENCE OF PARAMETER α.

As mentioned before, parameter α is of major influence on the number of trips and on the collection distance. The smaller α, the larger the number of trips (trips are shorter). It is more important to reduce the number of trips than to reduce the total collection distance, as the waste collection point is situated a long way from the collection area. Therefore parameter α close to 1 yields the best results in this case.

6.4 COMPARISON WITH THE CURRENT SYSTEM.

Using the aggregated collection area of the six current sectors, an overall reduction in driving distance of 15% was found. Due to a reduction in the number of trips (10 trips/week against minimum 12 trips/week in the current system) total driving distance to and from the collection area was reduced by 14.5%. Another 4.5% was saved on collection distance. For the individual sectors there was no clear improvement. Whereas the Path Scanning Algorithm was developed for a CCPP problem, the scenario of individual sectors is more related to a CPP problem as one trip is mostly sufficient for collecting all waste in a sector. Aggregating sectors gives a better overall result as it reduces the number of partly loaded trucks which has an influence on the total number of trips.

References

[1] Beltrami, E.J., Bodin, L.D., Networks and vehicle routing for municipal waste collection, Networks, 4 (1974), pp 65-94.

[2] Bodin, L., Fagin, G., Welebny, R., Greenberg, J., The design of a computerized sanitation vehicle routing and scheduling system for the town of OYSTER BAY, New York, Computer & Operations Research, 16 (1989), pp 45-54.

[3] Bodin, L., Golden, B., Assad, A., Ball, M., Routing and scheduling of vehicles and crews: the state-of-the-art, Computer & Operations Research, 10 (1983), pp 63-211.

[4] Christofides, N., The optimal traversal of a graph, Omega, 1.6 (1973), pp 719-732.

[5] Dijkstra, E., A note on two problems in connection with graphs, Numerische Mathematik, 1 (1959), pp 269-271.

[6] Golden, B., De Armon, J., Baker, E., Computational experiment with algorithms for a class of routing problems, Computer & Operations Research, 10.1 (1983), pp 47-59.

[7] Golden, B., Wong, R., Capacitated arc routing problems, Networks, 11 (1981), pp. 305-315.

[8] Minieka, E., Optimisation algorithms for network and graphs, Marcel Dekker, New York, 1978.

[9] Stern, H., Dror, M., Routing electric meter readers, Computer & Operations Research, 6 (1979), pp 209-223.

[10] de Favereau de Jeneret, H., Oreins, R., Optimalisatie van de huisvuilophaling, Eindwerk Industrieel Beleid, K.U.Leuven, 1989.