SOLVING A VEHICLE ROUTING PROBLEM ARISING IN SOFT DRINK DISTRIBUTION

by

Julie Privé[1]
Jacques Renaud[1]
Fayez Boctor [1]
Gilbert Laporte[2]

August 2004

[1] Network Organisation Technology Research Center (CENTOR), Faculté des Sciences de l'administration, Université Laval, Québec, Canada G1K 7P4.
[2] Canada Research Chair in Distribution Management, HEC Montréal, 3000 Chemin de la Côte-Sainte-Catherine, Montréal, Québec, Canada H3T 2A7.

SOLVING A VEHICLE ROUTING PROBLEM ARISING IN SOFT DRINK DISTRIBUTION

by

Julie Privé[1]
Jacques Renaud[1]
Fayez Boctor[1]
Gilbert Laporte[2]

1. Network Organisation Technology Research Center (CENTOR), Faculté des Sciences de l'administration, Université Laval, Québec, Canada G1K 7P4.
2. Canada Research Chair in Distribution Management, HEC Montréal, 3000 Chemin de la Côte-Sainte-Catherine, Montréal, Québec, Canada H3T 2A7.

Abstract

The problem studied in this article arises from the distribution of soft drinks and collection of recyclable containers in a Quebec based company. It can be modeled as a variant of the vehicle routing problem with a heterogeneous vehicle fleet, time windows, capacity and volume constraints, and an objective function combining routing costs and the revenue resulting from the sale of recyclable material. Three construction heuristics and an improvement procedure are developed for the problem. Comparative tests are performed on a real-life instance and on ten randomly generated instances.

Key words : heterogeneous vehicle routing problem, time windows, soft drink distribution, heuristics.

INTRODUCTION

The purpose of this article is to describe and solve a complex problem arising in soft drink distribution. The problem is that of a distributor called "*Distribution Jacques Dubois*" serving close to 500 customers in the Quebec City area. The company is engaged in the delivery of a variety of non-alcoholic drinks and in the pickup of empty recyclable cans and bottles. As will be seen, the problem can be modeled as a multi-product large scale pickup and delivery heterogeneous *vehicle routing problem* (VRP) subject to a variety of side constraints. The objective is the minimization of routing costs, minus the revenue derived from the collection of empty recyclable containers. The expression "pickup and delivery" must be interpreted with care in our context. It does not refer to situation where vehicles pickup goods at some locations in order to deliver them to others. In the problem considered in this article, all delivered goods originate from a unique *distribution center* (DC) and all collected goods end up at the DC. In other words, the quantity of goods to deliver goes down as vehicles complete their routes and, conversely, that of collected goods goes up. For simplicity we will refer to this problem as the *Soft Drink Distribution Problem* (SDP).

Several authors have previously addressed distribution management problems arising in the food and soft drink industry. See for example Golden and Wasil[1], Carter *et al.*[2], Tarantilis and Kiranoudis[3] and Golden *et al.*[4] as well as the references quoted in these papers. From a methodological point of view, the SDP shares some features of the VRP with time windows[5], the VRP with backhauls[6] and the VRP with heterogeneous vehicles[7].

The SDP will be tackled by means of three heuristics based on classical construction and improvement procedures[8]. The first heuristic is a nearest neighbour procedure while the other two make a selection of vehicle routes from a set of good candidate solutions. The improvement procedure uses a combination of 3-opt and 2-interchange moves as well as route merges. The proposed heuristics are more complicated than those used for vehicle routing problems involving only deliveries or collections. The presence of combined deliveries and collections in our problem mean that additional tests are required to preserve feasibility. Computational experiment will be performed on instances derived from real-life instances and on real-life data as well. The

SDP statement and model, algorithms, and computational results will be presented in the following three sections respectively, followed by the conclusion.

PROBLEM STATEMENT AND MODEL

The SDP is defined on a directed graph $G = (V, A)$, where $V = \{0, 1, \ldots, n\}$ is the vertex set and A is the arc set. Vertex 0 corresponds to the DC while the remaining vertices represent customers. With each arc (i, j) is associated a duration t_{ij} equal to the travel time from i to j plus the service time at customer i. The service of customer i must start within a time window $[a_i, b_i]$. A heterogeneous vehicle fleet is based at the DC. Vehicle types are labelled by k ($k = 1, \ldots, K$) and the number of vehicles of type k is noted m_k. A vehicle of type k has a fixed cost C_k and a variable travel cost c_{ijk} for arc (i, j). All costs are scaled and apply to the same period, e.g., one working day. Vehicles of type k have the same volume capacity V_k, the same weight capacity W_k and a maximum working time D_k. There are H product types, indexed by h, to be delivered to customers. The demand of customer i for product h is denoted by q_{ih}. Product h has a unit weight w_h and a unit volume v_h. In our context products are palletized, and the product volume that can be loaded on a pallet is equal to v. It follows that the number P_S of pallets loaded on a vehicle visiting a subset S of customers is equal to:

$$P_S = \left\lceil \sum_{h=1}^{H} \sum_{i \in S} v_h q_{ih} / v \right\rceil .$$

Denoting by w the weight of a pallet, these P_S pallets generates an extra weight wP_S for the vehicle visiting the customer set S. Each customer i generates a quantity of recyclable material (empty drink containers) to be collected. Denote by r_i and s_i the volume and weight of recyclable material generated by customer i.

The delivery and pickup operations are organized as follows. Once a day each vehicle is loaded at the DC with the products to be delivered to its customers. After delivering the demand of customer i, all or a part of the recyclable material available at this customer site is loaded in the vehicle depending on the unused space and weight capacity of the vehicle at this moment. In practice, the weight of the material collected at customer i is always less than the weight of what is delivered to this same customer. Thus s_i is not binding and will be ignored in our model. The recyclable material is brought back to the DC where it is eventually collected by a recycler. The distributor makes a revenue equal to f per unit volume of recyclable material.

The SDP consists of designing optimal delivery and pickup routes: 1) starting and ending at the DC, 2) visiting each customer exactly once, 3) delivering all demands, and 4) satisfying vehicle availability, time window, duration, volume and weight constraints. The solution cost is equal to the sum of vehicle fixed and variable costs, minus the revenue generated by the collection of recyclable material.

The problem can be formulated as an integer program. While we will not use the formulation to solve the problem optimally, we believe it is useful to formulate it in order to remove any ambiguity regarding its precise definition. Define the following binary variables:

x_{ijkl} = 1 if the l^{th} vehicle of type k travels directly from customer i to customer j ($i \neq j$) and equal 0 otherwise;

y_{ikl} = 1 if customer i is visited by the l^{th} vehicle of type k and equal 0 otherwise;

w_{kl} = 1 if the l^{th} vehicle of type k is used and 0 otherwise;

and define the continuous variables:

z_i = the volume of recyclable material collected at customer i.

In addition, define the auxiliary continuous decision variables u_i^t and u_i^r equal to the vehicle arrival time and to the volume of recyclable material in the vehicle when leaving i, respectively; similarly let u_{ik}^v be the volume of goods remaining to be delivered by the vehicle of type k upon arriving at i. The formulation is then:

$$\textbf{\textit{Minimize}} \quad \sum_{k=1}^{K}\sum_{l=1}^{m_k} C_k w_{kl} + \sum_{i=0}^{n}\sum_{j=0}^{n}\sum_{k=1}^{K}\sum_{l=1}^{m_k} c_{ijk} x_{ijkl} - \sum_{i=1}^{n} f z_i \tag{1}$$

$$\textbf{\textit{subject to}} \quad \sum_{k=1}^{K}\sum_{l=1}^{m_k} y_{ikl} = 1 \qquad i = 1, \ldots, n \tag{2}$$

$$\sum_{i=0}^{n} x_{ijkl} = \sum_{i=0}^{n} x_{jikl} \qquad j = 1, \ldots, n,\ k = 1, \ldots, K,\ l = 1, \ldots, m_k \tag{3}$$

$$\sum_{j=0}^{n} x_{ijkl} = y_{ikl} \qquad i = 1, \ldots, n,\ k = 1, \ldots, K,\ l = 1, \ldots, m_k \tag{4}$$

$$\sum_{i=1}^{n} y_{ikl} \le n w_{kl} \qquad k = 1, \dots, K,\ l = 1, \dots, m_k \tag{5}$$

$$\sum_{i=0}^{n} \sum_{j=0}^{n} t_{ij} x_{ijkl} \le D_k \qquad k = 1, \dots, K,\ l = 1, \dots, m_k \tag{6}$$

$$a_i \le u_i^t \le b_i \qquad i = 1, \dots, n \tag{7}$$

$$u_i^t \ge t_{0i} \qquad i = 1, \dots, n \tag{8}$$

$$u_i^t + t_{ij} - u_j^t \le \left(1 - \sum_{k=1}^{K} \sum_{l=1}^{m_k} x_{ijkl}\right) D_k \qquad i = 1, \dots, n,\ j = 2, \dots, n \tag{9}$$

$$u_{0k}^v \le V_k \qquad k = 1, \dots, K \tag{10}$$

$$u_{ik}^v - \sum_{h=1}^{H} v_h q_{ih} - u_{jk}^v \ge \left(1 - \sum_{l=1}^{m_k} x_{ijkl}\right) V_k \quad i = 0, \dots, n,\ j = 1, \dots, n,\ k = 1, \dots, K \tag{11}$$

$$w \left\lceil \sum_{h=1}^{H} \sum_{i=1}^{n} v_h q_{ih} y_{ikl} \Big/ v \right\rceil + \sum_{h=1}^{H} \sum_{i=1}^{n} w_h q_{ih} y_{ikl} \le W_k \qquad k = 1, \dots, K,\ l = 1, \dots, m_k \tag{12}$$

$$z_i \le r_i \qquad i = 1, \dots, n \tag{13}$$

$$u_i^r + z_i - u_j^r \le \left(1 - \sum_{k=1}^{K} \sum_{l=1}^{m_k} x_{ijkl}\right) V_k \qquad i = 1, \dots, n,\ j = 2, \dots, n \tag{14}$$

$$u_{ik}^v + u_i^r \le V_k + \left(1 - \sum_{l=1}^{m_k} y_{ikl}\right) V_k \qquad i = 1, \dots, n,\ k = 1, \dots, K \tag{15}$$

$$x_{ijkl}, y_{ikl}, w_{kl} = 0 \text{ or } 1 \qquad i = 0, \dots, n,\ j = 0, \dots, n,\ k = 1, \dots, K,\ l = 1, \dots, m_k \tag{16}$$

$$u_i^t, u_{ik}^v, u_i^r \ge 0 \qquad i = 0, \dots, n,\ k = 1, \dots, K. \tag{17}$$

In this formulation, constraints (2) specify that each customer is visited exactly once while constraints (3) are flow conservation equations. Constraints (4) express the y_{ikl} variables in terms of the x_{ijkl} variables. Note that these constraints and the y_{ikl} variables could be eliminated from the model by using a substitution mechanism; however, their presence makes the formulation easier to follow. Constraints (5) mean that no customer can be visited by the l^{th} vehicle of type k if this vehicle is not used. Constraints (6) specify the duration constraints on vehicle routes. Constraints (7), (8) and (9) ensure that all time windows are respected. Constraints (10) and (11) ensure that the volume of goods to be delivered upon leaving each customer is always feasible. By constraints (12) the weight constraint on each vehicle is always satisfied.

Theses constraints could easily be linearized through the introduction of dummy variables. Constraints (13) and (14) enforce the restrictions on the volume of recyclable material to be picked up, while constraints (15) impose, for each customer, a limit on the combined volume of products to be delivered and recyclable material. It is worth observing that to enforce the weight restrictions it is sufficient to control the load of each vehicle when it leaves the DC whereas the volume restrictions require the imposition of a separate constraint for each customer.

As is common in similar models derived from the Miller-Tucker-Zemlin[9] formulation for the *Traveling Salesman Problem*, no subtour elimination constraints are required because of (9). To our knowledge, our formulation is the first of this type to explicitly handle a heterogeneous vehicle fleet. To take vehicle specific capacities into account, a second index is needed in the u_{ik}^{v} variables in constraints (10) and (11). As in the work of Desrochers and Laporte[10] several of the constraints can easily be lifted.

ALGORITHMS

We now describe the heuristics we have developed to solve the SDP, each consisting of a construction phase followed by the same improvement phase. The first heuristic is a nearest neighbour procedure while the remaining two are inspired from petal algorithms for the VRP[11,12].

Nearest Neighbour Heuristic

The nearest neighbour heuristic (NNH) iteratively constructs vehicle routes on the set U of yet unvisited customers which is initialized as $U = V \setminus \{0\}$. Contrary to what is usually done in nearest neighbour heuristics, our NNH constructs several tentative routes before selecting the best one according to an average cost criterion. Starting from the DC, the NNH constructs a feasible route in a greedy fashion using the smallest vehicle available (i.e., the vehicle having the smallest volume) until no more customers can be added to the route. The NNH then extends the same route by moving to the smallest available vehicle having a larger size, and repeats this process until no larger vehicle is available. Among all routes constructed, the NNH selects the route having the smallest average cost per visited customer, where the cost is defined as in (1). The set of unvisited customers is updated and the process is repeated until $U = \emptyset$ or until no vehicle remains, in which case, as in Cordeau, Gendreau and Laporte[13], an arbitrary large artificial vehicle is used to accommodate all unrouted customers. Next, we apply a 2-opt[14] procedure to each constructed route. An attempt to regain feasibility is later made in the improvement phase.

Because the SDP involves deliveries and pickups along the same vehicle route, the volume remaining available in the vehicle does not necessarily changes in a monotonic fashion. As a result, adding a new customer to a partially constructed route may cause infeasibilities in the already constructed part of the route. In such a case, previously made decisions on recyclable material volumes must be modified. To illustrate, consider the four customer example of Table 1 and suppose a vehicle of volume 20 units is used. Then the first three customers can feasibly be served by this vehicle as shown in Table 2. However, the vehicle cannot feasibly visits all four customers while keeping the pickup quantities unchanged, as shown in Table 3. Adding customer 4 to the route therefore requires modifying the quantity of recyclable material to be picked up or left behind at customers 2 and 3, as shown in Table 4, which also has an impact on the route cost. In other words, contrary to the standard nearest neighbour heuristic, our NNH may update previously made decisions. The same feature exists in our two petal heuristics and in our proposed improvement phase.

Table 1: Delivery and pickup volumes for the four-customer example

Customer	1	2	3	4
Delivery volume	5	5	5	5
Available pickup volume	5	6	6	3

Table 2: Volume in vehicle if all recyclable material of the first three customers is picked up

Customer	1	2	3
Product volume in vehicle at arrival	15	10	5
Product volume delivered	5	5	5
Product volume in vehicle at departure	10	5	0
Recyclable material volume in vehicle at arrival	0	5	11
Recyclable material volume picked up	5	6	6
Recyclable material volume in vehicle at departure	5	11	17
Total volume in vehicle at departure	15	16	17

Table 3: Volume in vehicle if all recyclable material of four customers is picked up

Customer	1	2	3	4
Product volume in vehicle at arrival	20	15	10	5
Product volume delivered	5	5	5	5
Product volume in vehicle at departure	15	10	5	0
Recyclable material volume in vehicle at arrival	0	5	11	17
Recyclable material volume picked up	5	6	6	3
Recyclable material volume in vehicle at departure	5	11	17	20
Total volume in vehicle at departure	20	21	22	20

Table 4: Volume of recyclable material to be left in order to regain feasibility

Customer	1	2	3	4
Product volume in vehicle at arrival	20	15	10	5
Product volume delivered	5	5	5	5
Product volume in vehicle at departure	15	10	5	0
Recyclable material volume in vehicle at arrival	0	5	10	15
Recyclable material volume picked up	5	5	5	3
Recyclable material volume uncollected	0	1	1	0
Recyclable material volume in vehicle at departure	5	10	15	18
Total volume in vehicle at departure	20	20	20	18

First Petal Heuristic

The first petal heuristic (FPH) constructs a set R of feasible routes based on NNH and then makes an optimal selection of the routes to be used by solving a generalized set partitioning problem. To construct the set of candidate routes, NNH is initiated n times from a seed vertex of $V \setminus \{0\}$. Denote by U the set of unrouted customers, by F the set of vertices already used as seed vertex and by R the set of generated routes. The steps of the proposed heuristic are as follows.

Step 1: *Initialization*

Set $F = \{0\}$ and $R = \emptyset$.

Step 2: *Selection of a seed vertex*

If $F = V$, go to Step 4. Otherwise, select from $V \setminus F$ the next seed vertex i defined as the closest vertex to the DC. Set $F = F \cup \{i\}$ and $U = V \setminus \{0, i\}$. Include the route $(0, i, 0)$ into R.

Step 3: *Route expansion*

Starting from i, iteratively expand the route in a greedy fashion by selecting at each iteration a vertex j that can feasibly be included in the route. If $U \setminus F \neq \emptyset$, then j is selected from $U \setminus F$; otherwise, j is selected from $F \setminus \{i\}$. Set $U = U \setminus \{j\}$. Apply a 2-opt procedure[13] to route (0, i, …, j, 0) and insert the resulting route into R. Go to Step 2.

Step 4: *Route evaluation*

Successively assign to each route in R all vehicles that can feasibly make all its deliveries and compute the corresponding cost using (1). Let E be the set of all route-vehicle combinations thus constructed.

Step 5: *Generalized set partitioning algorithm*

Select from E an optimal subset of routes by solving a generalized set partitioning problem. As in the NNH, all routes selected by the model will later be post-optimized. Define:

x_e: a binary variable equal to 1 if and only if rout $e \in E$ is selected, 0 otherwise;

π_e: the cost of route e;

α_{ei}: a binary parameter equal to 1 if and only if customer i is visited by route e, 0 otherwise;

β_{ek}: a binary parameter equal to 1 if and only if route e is assigned to a vehicle of type k, 0 otherwise.

The model to solve is then:

Minimize $$\sum_{e \in E} \pi_e x_e \tag{18}$$

subject to $$\sum_{e \in E} \alpha_{ei} x_e = 1 \qquad i = 1, \ldots, n \tag{19}$$

$$\sum_{e \in E} \beta_{ek} x_e \leq m_k \qquad k = 1, \ldots, K \tag{20}$$

$$x_e = 0 \text{ or } 1 \qquad e \in E. \tag{21}$$

Second Petal Heuristic

The second petal heuristic (SPH) first constructs an initlal set R of routes by applying only once Steps 1 to 3 of the FPH. Let T be the ordered set of vertices appearing on the longest route, where the first vertex is the one closest to the DC. Then apply FPH with the following modifications in Steps 2 and 3:

Step 2': *Selection of a seed vertex*

Here vertex i is selected from T, and F is an ordered set so that the step becomes: if $F = V$, go to Step 4. Otherwise, designate the next vertex of T as the seed vertex i. Append i in the last position of F and set $U = V \setminus \{0, i\}$. Include the route $(0, i, 0)$ into R.

Step 3': *Route expansion*

Starting from vertex i, iteratively expand the route in a greedy fashion by selecting at each iteration a vertex j that can feasibly be included in the route. If $U \setminus F \neq \emptyset$, then j is selected from $V \setminus F$; otherwise, j is next yet unselected element of $F \setminus \{i\}$. Set $U = U \setminus \{j\}$. If $j \notin T$ append it to T in the last position. Apply a 2-opt[13] procedure to route $(0, i, \ldots, j, 0)$ and insert the resulting route into R. Go to Step 2'.

Both FPH and SPH perform a polar sweep based on the distance matrix. The difference between the two methods lies in the way the set F of vertices already used as a seed are selected in Step 3. In FPH, when $U \setminus F$ is empty, all vertices previously used as a seed are considered in an order determined by their distance to the DC. In SPH, when $U \setminus F$ is empty, the seed vertices are used in the same order in which they have been previously been considered.

Improvement Phase

The following improvement phase is applied to each of the three solutions generated by the NNH, FPH and SPH. It iteratively applies the following three steps until no further improvement can be achieved.

Step 1: *3-opt*

Apply a 3-opt[14] procedure to each route of the solution.

Step 2: *Restricted 2-interchange*

Apply a restricted 2-interchange[15] procedure to the solution. The λ-interchange procedure proposed by Osman takes two routes at a time and attempt all possible transfers and exchanges of up to λ vertices from each route, into all possible positions of the other route. Our implementation of this procedure is restricted in the sense that we only move chains containing at most 2 consecutive vertices. The set of feasible transfers and exchanges is depicted in Figure 1.

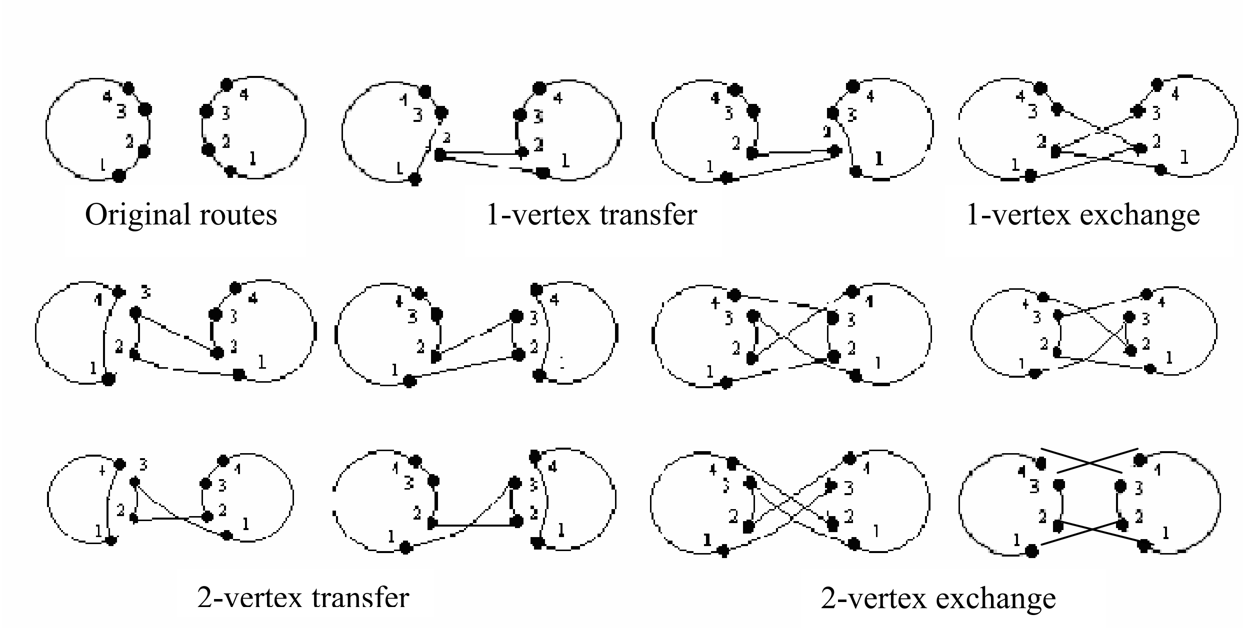

Figure 1: Implemented 2-interchange moves

Step 3: *Route merge*

Consider in turn all route pairs. If there exists an available vehicle that can feasibly serve all customers of the considered two routes, use the NNH to construct a single route visiting all customers of the two routes. If this results in a cost saving, the combined route then replaces the two individual routes.

COMPUTATIONAL RESULTS

The algorithms were coded in *Visual Basic* 6.0 on a COMPAQ Presario 2100 desktop with a Celeron 1,60 GHz, 192 Mo RAM and run under Microsoft Windows XP 2002. The set partitioning models were solved using Cplex 8.0. The proposed heuristics were tested on a real-life case and on ten randomly generated instances.

For the real-life case, we designed routes to deliver the demand of the 164 customers of *Distribution Jacques Dubois* over a one-week period. Customer locations, their demands of 125 different products, the fleet composition and capacity, time windows and route maximum duration were provided by the company. We also obtained the exact delivery plan designed by the company. This plan is composed of ten routes having a total length of 2514 km. The total distance was computed using the Microsoft mapping and location platform *MapPoint*. The original distances computed by *MapPoint* are based on the use of a ferry boat on some itineraries. Because the company's vehicles cannot use this ferry boat, changes had to be made to some of the *MapPoint* shortest distances. By applying the NNH we obtained a plan composed of nine routes totalizing 1927 km. Thus our plan reduces the current distance by about 23%. Both FPH and SPH generate 12 routes of length 2254 km and 2013 km respectively, representing a reduction of 10.3% and 19.9%, respectively. The respective computing times for NNH, FPH and SPH are 2524, 3895 and 2591 seconds. The number of routes generated by FPH and SPH are 8444 and 8584 respectively. The resulting set partitioning models, which contain 8444 or 8584 variables and 167 constraints, were solved in only a few seconds with Cplex. Cost comparisons cannot be made since the company did not provide its cost data.

A set of ten randomly generated instances with 150 customers each was then used to assess the relative performance of the three proposed heuristics. These test problems were generated as follows. First, we randomly drew 150 customers among the 527 customers contained in customer data base of the company. We then used *MapPoint* to determine the distance matrix. The travel time was then estimated assuming an average speed of 60 km/h. Customers were then randomly assigned to one of the four customers categories : supermarkets, grocery stores, convenience stores, others. The probability of assigning a customer to any category was the same as the actual percentage of customers belonging to that category. The demand and recycling quantities of these four categories are presented in Table 5, for each of three product families: small containers, medium containers and large containers. The demand for each of these three product families was randomly generated in the ranges given in the table. Note that in this industry volumes are measured in cubic meters while weights are measured in pounds (in Canada a metrification plan was initiated in the early 1970s but it has to be interrupted in mid-course for political reasons). Finally, time windows were randomly imposed to 20% of the customers; in such cases we imposed that the customer should be reached within the first 360 minutes of the route start.

Table 5: Demand and recycling material ranges for the generated instances

Customer category	Percentage in the company data base	Generated recyclable material (m^3)	Demand for small containers (pounds)	Demand for medium containers (pounds)	Demand for large containers (pounds)
Supermarkets	10	[4, 6]	[0, 25]	[0, 2]	[0, 30]
Grocery stores	25	[2, 4]	[0, 15]	[0, 5]	[0, 25]
Convenience stores	45	[0, 2]	[0, 10]	[0, 8]	[0, 20]
Others	20	[0, 0.2]	[0, 5]	[0, 10]	[0, 10]
		Unit weight (pounds)	15	20	40
		Unit volume (m^3)	[0.005, 0.02]	[0.02, 0.03]	[0.03, 0.07]

Table 6 gives the main characteristics of the vehicle fleet for the test problems. The maximum route duration is set equal to 480 minutes and service time is set to ten minutes. The average speed is 60 km/h, the weight of an empty pallet is 76 pounds, and the maximum volume to be loaded on a pallet is 2 m^3. Finally, the revenue of recyclable material is \$ 10/m^3.

Table 6: Fleet composition and characteristics

Vehicle type	Number of vehicles	Volume capacity (m^3)	Weight capacity (pounds)	Fixed cost (\$)	Variable cost (\$/km)
1	6	22	6 000	100	1.00
2	6	28	8 000	150	1.10
3	6	36	16 000	175	1.20

Each of the three heuristics was used to solve each of the ten test instances. Table 7 shows the net cost (fleet cost minus revenue of collected recyclable material) of the solutions obtained by each heuristic used alone, and by the same heuristics followed by the proposed improvement procedure. This table shows that SPH performs best followed by the NNH, and FPH. We observe that the SPH followed by the improvement procedure obtains the lowest cost for eight of the ten instances, while each of the other two heuristics produces the best solution for only one instance. The computational times of these heuristics, when followed by the proposed improvement heuristic, are very similar to each other. For FPH and SPH, the average number of generated routes ranges from 5000 to 6000 and the resulting models were always solved within 12 seconds.

If we take the best solution for each problem as a reference, the average deviations over the minimum for NNH, FPH and SPH are 7.46%, 15.32% and 0.79%, respectively.

Other computational results are provided in Table 8. The first row gives the average fleet utilisation cost (fixed cost plus variable cost) over the ten instances, the second row shows the average revenue generated by the collected recyclable material and the third correspond to the average net cost. The fourth row gives the percentage of recyclable material that could be collected. The fifth row shows the percentage of vehicle weight capacity at time of departure, while the sixth row gives the average number of vehicle used in the solution.

CONCLUSION

We have described and solved a complex real-life problem arising in the distribution of soft drinks. A distinguishing feature of this problem is that vehicles must deliver goods and pickup recyclable materials at customer locations. We have modeled this problem as a mixed integer program and we have proposed three heuristics for its resolution. A construction heuristic and two petal based heuristics were developed. These were tested on a real-life case and on ten randomly generated instances. Results obtained on the real-life case show that a 23 % distance reduction can be achieved.

Table 7: Total solution costs. Boldface figures correspond to the best results

Instance	Heuristic alone			Heuristic followed by the improvement procedure		
	NNH	FPH	SPH	NNH	FPH	SPH
1	2758.8	3583.3	2471.1	2318.6	2317.6	**2263.0**
2	2132.7	3141.1	2394.6	1896.4	2151.4	**1837.2**
3	2074.8	2397.4	2028.6	1625.4	**1530.2**	1597.8
4	2590.2	2826.0	2137.3	2058.8	2164.6	**1820.4**
5	2231.5	3372.5	1866.8	1719.9	2077.3	**1483.7**
6	2205.9	2739.3	2021.1	**1673.5**	2033.4	1736.9
7	2078.9	2601.4	2027.5	1565.0	1748.3	**1516.8**
8	2104.1	2586.7	1642.7	1700.9	1912.2	**1469.4**
9	2724.3	3393.0	2592.4	1934.6	2188.0	**1858.2**
10	3120.6	3234.0	2347.6	2339.2	2499.7	**1935.4**
Average net cost	2402.2	2987.5	2152.9	1883.2	2062.3	**1751.8**
Average computational time (sec)	76	490	511	1474	1465	1503

Table 8: Additional output data

	Heuristic alone			Heuristic followed by the improvement procedure		
	NNH	FPH	SPH	NNH	FPH	SPH
Average fleet fixed and variable cost	4866.8	5583.6	4749.5	4514.2	4707.1	4382.0
Average recyclable material revenue	2464.6	2596.1	2596.5	2631.0	2644.8	2630.2
Average net cost (fleet cost minus revenue)	2402.2	2987.5	2153.0	1883.2	2062.3	1751.8
Percentage of collected recyclable material	93.2	98.3	98.1	99.4	99.9	99.5
Percentage usage of vehicle capacity	80.2	56.6	76.9	82.8	75.9	85.7
Average number of vehicles in solution	11.7	16.7	13.2	11.6	13.9	12.2

Acknowledgement

This research was partially supported by Grants OPG0036509, OPG0039682 and OPG0172633 from the Canadian Natural Sciences and Engineering Research Council (NSERC). This support is gratefully acknowledged.

REFERENCES

[1] Golden B.L. and Wasil E.A., "Computerized vehicle routing in the soft drink industry", *Operation Research* (1987) 35: 6-17.

[2] Carter M.W., Favolden J.M., Laporte G. and Xu J., "Solving an integrated logistic problem arising in grocery distribution", *INFOR* (1996) 34: 290-306.

[3] Tarantilis C.D. and Kiranoudis C.T., "A meta-heuristic algorithm for the efficient distribution of perishable foods", *Journal of Food Engineering* (2001) 50: 1-9.

[4] Golden B.L., Assad A.A., and Wasil E.A., "Routing vehicles in the real world: Applications in the solid waste, beverage, food, dairy, and newspaper industries", in *The Vehicle Routing Problem*, P. Toth and D. Vigo (eds). SIAM Monographs on Discrete Mathematics and Applications, Philadelphia (2002): 245-286.

[5] Cordeau J.-F., Desaulniers G., Desrosiers J., Solomon M.M. and Soumis F., "Vehicle routing problem with time windows", in *The Vehicle Routing Problem*, P. Toth and D. Vigo (eds), SIAM Monographs on Discrete Mathematics and Applications, Philadelphia (2002): 157-193.

[6] Toth P. and Vigo D., "Vehicle routing problem with backhauls", in *The Vehicle Routing Problem*, P. Toth and D. Vigo (eds), SIAM Monographs on Discrete Mathematics and Applications, Philadelphia (2002): 195-224.

[7] Tarantilis C.D., Kiranoudis C.T. and Vassiliadis V.S., "A threshold accepting metaheuristic for the heterogeneous fixed vehicle routing problem", *European Journal of Operational Research* (2004) 152: 148-158.

[8] Laporte G. and Semet F. "Classical heuristics for the capacitated vehicle routing problem", in *The Vehicle Routing Problem*, P. Toth and D. Vigo (eds), SIAM Monographs on Discrete Mathematics and Applications, Philadelphia (2002): 109-128.

[9] Miller C.E., Tucker A.W. and Zemlin R.A., "Integer programming formulations and traveling salesman problems", *Journal of the Association for Computing Machinery* (1960) 7: 326-329.

[10] Desrochers M. and Laporte G., “Improvements and extensions to the Miller-Tucker-Zemlin subtour elimination constraints”, *Operations Research Letters* (1991) 10: 27-36.

[11] Ryan D.M., Hjorring C. and Glover F., “Extensions of the petal method for vehicle routing”, *Journal of the Operational Research Society* (1993) 44: 289-296.

[12] Renaud J., Boctor F.F. and Laporte G., “An improved petal heuristic for the vehicle routing problem”, *Journal of the Operational Research Society* (1996) 47: 329-336.

[13] Cordeau J.-F., Gendreau M. and Laporte G., “A tabu search heuristic for periodic and multi-depot vehicle routing problems”, *Networks* (1997) 30: 105-119.

[14] Lin S., “Computer solutions of the traveling salesman problem”, *The Bell System Technical Journal* (1965): 2245-2269.

[15] Osman I.H., “Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem”, *Annals of Operations Research* (1993) 41: 421-451.

Table 1: Delivery and pickup volumes for the four-customer example

Customer	1	2	3	4
Delivery volume	5	5	5	5
Available pickup volume	5	6	6	3

Table 2: Volume in vehicle if all recyclable material of the first three customers is picked up

Customer	1	2	3
Product volume in vehicle at arrival	15	10	5
Product volume delivered	5	5	5
Product volume in vehicle at departure	10	5	0
Recyclable material volume in vehicle at arrival	0	5	11
Recyclable material volume picked up	5	6	6
Recyclable material volume in vehicle at departure	5	11	17
Total volume in vehicle at departure	15	16	17

Table 3: Volume in vehicle if all recyclable material of four customers is picked up

Customer	1	2	3	4
Product volume in vehicle at arrival	20	15	10	5
Product volume delivered	5	5	5	5
Product volume in vehicle at departure	15	10	5	0
Recyclable material volume in vehicle at arrival	0	5	11	17
Recyclable material volume picked up	5	6	6	3
Recyclable material volume in vehicle at departure	5	11	17	20
Total volume in vehicle at departure	20	21	22	20

Table 4: Volume of recyclable material to be left in order to regain feasibility

Customer	1	2	3	4
Product volume in vehicle at arrival	20	15	10	5
Product volume delivered	5	5	5	5
Product volume in vehicle at departure	15	10	5	0
Recyclable material volume in vehicle at arrival	0	5	10	15
Recyclable material volume picked up	5	5	5	3
Recyclable material volume uncollected	0	1	1	0
Recyclable material volume in vehicle at departure	5	10	15	18
Total volume in vehicle at departure	20	20	20	18

Table 5: Demand and recycling material ranges for the generated instances

Customer category	Percentage in the company data base	Generated recyclable material (m^3)	Demand for small containers (pounds)	Demand for medium containers (pounds)	Demand for large containers (pounds)
Supermarkets	10	[4, 6]	[0, 25]	[0, 2]	[0, 30]
Grocery stores	25	[2, 4]	[0, 15]	[0, 5]	[0, 25]
Convenience stores	45	[0, 2]	[0, 10]	[0, 8]	[0, 20]
Others	20	[0, 0.2]	[0, 5]	[0, 10]	[0, 10]
		Unit weight (pounds)	15	20	40
		Unit volume (m^3)	[0.005, 0.02]	[0.02, 0.03]	[0.03, 0.07]

Table 6: Fleet composition and characteristics

Vehicle type	Number of vehicles	Volume capacity (m^3)	Weight capacity (pounds)	Fixed cost ($)	Variable cost ($/km)
1	6	22	6 000	100	1.00
2	6	28	8 000	150	1.10
3	6	36	16 000	175	1.20

Table 7: Total solution costs. Boldface figures correspond to the best results

Instance	Heuristic alone			Heuristic followed by the improvement procedure		
	NNH	FPH	SPH	NNH	FPH	SPH
1	2758.8	3583.3	2471.1	2318.6	2317.6	**2263.0**
2	2132.7	3141.1	2394.6	1896.4	2151.4	**1837.2**
3	2074.8	2397.4	2028.6	1625.4	**1530.2**	1597.8
4	2590.2	2826.0	2137.3	2058.8	2164.6	**1820.4**
5	2231.5	3372.5	1866.8	1719.9	2077.3	**1483.7**
6	2205.9	2739.3	2021.1	**1673.5**	2033.4	1736.9
7	2078.9	2601.4	2027.5	1565.0	1748.3	**1516.8**
8	2104.1	2586.7	1642.7	1700.9	1912.2	**1469.4**
9	2724.3	3393.0	2592.4	1934.6	2188.0	**1858.2**
10	3120.6	3234.0	2347.6	2339.2	2499.7	**1935.4**
Average net cost	2402.2	2987.5	2152.9	1883.2	2062.3	**1751.8**
Average computational time (sec)	76	490	511	1474	1465	1503

Table 8: Additional output data

	Heuristic alone			Heuristic followed by the improvement procedure		
	NNH	FPH	SPH	NNH	FPH	SPH
Average fleet fixed and variable cost	4866.8	5583.6	4749.5	4514.2	4707.1	4382.0
Average recyclable material revenue	2464.6	2596.1	2596.5	2631.0	2644.8	2630.2
Average net cost (fleet cost minus revenue)	2402.2	2987.5	2153.0	1883.2	2062.3	1751.8
Percentage of collected recyclable material	93.2	98.3	98.1	99.4	99.9	99.5
Percentage usage of vehicle capacity	80.2	56.6	76.9	82.8	75.9	85.7
Average number of vehicles in solution	11.7	16.7	13.2	11.6	13.9	12.2