Optimization of Encashment Routs in ATM Network

Elena Gubar, Maria Zubareva

St.Petersburg University,
Faculty of Applied Mathematics and Control Processes,
Universitetskiy pr. 35, St.Petersburg, 198504, Russia
fax:+7 (812) 428 71 59
http://www.apmath.spbu.ru
alyona.gubar@gmail.com
zubareva_ml@mail.ru

Abstract The main purpose of this work is to optimize cash flow in case of the encashment process in the ATM network. The solution of these problems is based on some modified algorithms for the Vehicle Routing Problem with Time Windows. A numerical example is considered.

Keywords: ATM network, route optimization, Vehicle Routing Problem with Time Windows.

1. Introduction

Nowadays ATM network and credit cards are the essential parts of modern lifestyle, and one of the most actual problem in the bank's ATM network is optimization of cash flow and organization of uninterrupted work. Serving the ATMs network is a costly task: it takes employees' time to supervise the network and make decisions about cash management and it involves high operating costs (financial, transport, etc.). Banks could reduce their costs applying competent encashment strategy and optimizing encashment routes in ATM network.

For the purpose of reducing bank's costs we could use algorithms for solving Vehicle Routing Problems (VRP). According to (Toth, 2001), the Vehicle Routing Problem is a problem of designing optimal routes for servicing a set of customers by a set of vehicles. The solution of the VRP calls for determination of a set of routes, each route is performed by a single vehicle that starts and ends in its own depot. This set of routes must satisfy the following conditions: all the requirements of the customers are fulfilled, all the operational constraints are satisfied, and the global transportation cost is minimized.

In previous paper (Gubar et al., 2011) we explore one of the modifications of VRPs, the Capacitated Vehicle Routing Problem, where the capacity restrictions for each vehicle are essential. Now we take under consideration the Vehicle Routing Problem with Time Windows (VRPTW) and focus on the fact that additionally each customer is associated with a time interval, called a time window. The service of each customer must start within a given time window. Such additional constraints allow to satisfy the requirements of real-life situations more carefully.

Thus, in this work we consider a problem in which a set of geographically dispersed ATMs with known requirements must be served with a fleet of money collector teams stationed in the depot in such a way as to minimize some distribution objective. We assume that the money collector teams are identical with the equal capacity and must start and finish their routes at the depot.

2. Formulation of the Vehicle Routing Problem with Time Windows

Consider the presentation of the VRPTW, where $V = (0, 1, \dots, n)$ is the complete set of vertices, each vertex corresponds to an ATM, vertex 0 corresponds to the depot. For each pair of ATMs, or ATMs and the depot, there is an associated cost c_{ij}. Each stop i requires a supply of q_i units from depot 0. A set of M identical vehicles of capacity Q is located at the depot and is used to service the stops; these M vehicles comprise the homogeneous vehicle fleet. It is required that every vehicle route starts and ends at the depot and that the load carried by each vehicle is no greater than Q.

A travel time between ATMs i and j is denoted as t_{ij}. Each stop is associated with a service time σ_i required by a vehicle to visit the ATM and to unload the quantity q_i (we assume $\sigma_0 = 0$). The start time of the service at stop i must be within a given time window $[a_i, b_i]$. A vehicle is permitted to arrive at stop i before the beginning of the time window and wait at no cost until time a_i. Also vehicles are time-constrained at the depot in that each vehicle must leave the depot and return back within the time window $[a_0, b_0]$.

The variable x_{ijk} is $0-1$ binary, it equals to 1 if and only if vehicle k visits stop j immediately after visiting stop i and 0 if not. The continuous variable s_{ik} denotes the time vehicle k begins service at stop i. It is assumed that s_{0k} is the departure time of vehicle k from the depot.

Here we present the formalization of the basic VRPTW problem (Hall, 2003):

$$min \sum_{k=1}^{M} \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ijk}, \tag{1}$$

$$\sum_{k=1}^{M} \sum_{j \in V} x_{ijk} = 1, \qquad i \in V_c, \tag{2}$$

$$\sum_{i \in V_c} q_i \sum_{j \in V} x_{ijk} \leq Q, \qquad k = 1, \dots, M, \tag{3}$$

$$\sum_{j \in V_c} x_{0jk} \leq 1, \qquad k = 1, \dots, M, \tag{4}$$

$$\sum_{i \in V} x_{ijk} - \sum_{i \in V} x_{jik} = 0, \qquad j \in V_c,\ k = 1, \dots, M, \tag{5}$$

$$s_{ik} + \sigma_i + t_{ij} - L(1 - x_{ijk}) \leq s_{jk}, \qquad i \in V,\ j \in V_c,\ k = 1, \dots, M, \tag{6}$$

$$s_{ik} + \sigma_i + t_{i0} - L(1 - x_{i0k}) \leq b_0, \qquad i \in V_c,\ k = 1, \dots, M, \tag{7}$$

$$a_i \leq s_{ik} \leq b_i, \qquad i \in V,\ k = 1, \dots, M, \tag{8}$$

$$x_{ijk} \in \{0, 1\}, \qquad i, j \in V,\ k = 1, \dots, M. \tag{9}$$

Constraints (2) state that each ATM must be visited exactly once. Constraints (3) are the capacity limitation on the vehicles. Constraints (4) force each vehicle to be used at most once and constraints (5) state that if a vehicle visits ATM, it must also depart from it. Constraints (6) impose that vehicle k cannot arrive at stop j before $s_{ik} + \sigma_i + t_{ij}$, if it travels from i to j. Constraints (7) force each vehicle k to return to the depot before time b_0. The scalar L can be any large number.

Constraints (8) ensure that all time windows are respected and constraints (9) are the integrality constraints.

2.1. Methods for solving VRPTW

General approaches for solving Vehicle Routing Problem with Time Windows could be divided into three groups: exact methods, heuristic and metaheuristic methods.

In exact methods the mixed-integer programming formulation of the VRPTW is solved. Such methods include branch-and-bound, branch-and-cut algorithms, and other techniques for solving integer programming problems. But the VRPTW is considered NP-hard and for problems of practical size computing exact solutions could be too complicated.

Because of the high complexity level of the VRPTW approximate heuristic and metaheuristic methods are of prime importance. Heuristic methods search for not optimal, but approximately optimal high-quality solution in acceptable time.

Heuristics methods for solving VRPTW could be divided into following groups:

1. **Route construction heuristics:** select stops sequentially until a feasible solution has been created. Stops are chosen based on some cost minimization criterion, often subject to the restriction that the selection does not create a violation of vehicle capacity or time window constraints. Among these methods are known:
   - extension to the savings heuristic of Clarke and Wright (Clarke et al., 1964);
   - time-oriented nearest neighbor;
   - Solomons time-oriented sweep heuristic (Solomon, 1987).
2. **Solution Improvement Methods:** based on the concept of iteratively improving the solution to a problem by exploring neighboring vertices.

Metaheuristic methods are the next step in development of heuristic methods. They try overcome the local minima in the searching process, while solution improvement methods stop after finding local solutions in the selected neighborhood. Among metaheuristic methods are known:

- ant colony optimization;
- simulated annealing;
- tabu search;
- genetic algorithms.

In current work we focus on the simulated annealing metaheuristcs for Vehicle Routing Problem with Time Windows and apply it for designing optimal routes for money collector teams.

2.2. Simulated Annealing

Simulated Annealing is an algorithmic approach to solving combinatorial optimization problems (Woch et al., 2009). The name of the algorithm derives from an analogy between solving optimization problems and simulating the annealing of solids. This method accepts search movements that temporarily produces degradations in a current solution found to a problem as a way to escape from local minima.

The simulated annealing algorithm is as follows (Chiang et al., 1996):

Step 1. Obtain an initial feasible solution S for the VRPTW
Step 2. Set the cooling parameters including the initial temperature T, the cooling ratio r, and the epoch length Len
Step 3.
3.1 For $1 < i < Len$ do
3.1.1 Pick a random neighbor solution S'
3.1.2 Let $A = Cost(S') - Cost(S)$
3.1.3 If $A < 0$, then set $S = S'$
3.1.4 If $A > 0$, then set $S = S'$ with probability
3.2. Set $T = rT$
Step 4. Return S

The simulated annealing algorithm starts with the initial feasible solution. To find this initial routes we use a time-oriented nearest-neighbor heuristic method, that belongs to the class of route construction algorithms.

2.3. A Time-Oriented Nearest-Neighbor Heuristic

In terms of our problem of designing optimal routes in ATM network the nearest-neighbor heuristic could be described in the following way. This heuristic starts every route by searching the unrouted ATM "closest" to the bank or the last ATM added without violating feasibility. This search is performed among all the ATMs who can feasibly be added to the end of the emerging route. A new route is started any time the search fails, unless there are no more ATMs to add (Solomon, 1987).

The metric used in this approach tries to account for both geographical and temporal closeness of ATMs. Let the last ATM on the current partial route be ATM i and let j denote any unrouted ATM that could be visited next. Let the metric c_{ij} measures the distance between two ATMs, T_{ij} — the time difference between the end of service at i and the beginning of service at j, and v_{ij} — the urgency of delivery to ATM j:

$$T_{ij} = g_j - (g_i + \sigma_i), \qquad v_{ij} = b_j - (g_i + \sigma_i + t_{ij}), \tag{10}$$

where g_i — the time of beginning servicing ATM i and g_j — the time of beginning servicing ATM j.

$$g_j = max\{a_j, g_i + \sigma_i + t_{ij}\}, \tag{11}$$

where a_i — the lower bound of time window, σ_i — service time of ATM i, and t_{ij} — travel time between ATMs i and j. Then the metric for searching "closest" ATM is:

$$d_{ij} = \delta_1 c_{ij} + \delta_2 T_{ij} + \delta_3 v_{ij}, \qquad \begin{array}{l} \delta_1 + \delta_2 + \delta_3 = 1, \\ \delta_1 \geq 0, \quad \delta_2 \geq 0, \quad \delta_3 \geq 0. \end{array} \tag{12}$$

3. Numerical simulation

Here we represent the application of simulated annealing heuristic for certain ATM network. We assume that the bank has three collector teams with equal vehicle capacity $Q = 12$ cartridges and each ATM requires $q_i = 3$ cartridges. Suppose that money collector teams should service 9 ATMs located at the different subway

Table1: Distances between ATMs and bank, m

	Bank	2	3	4	5	6	7	8	9	10
Bank	0	3250	6530	9000	5005	10007	6680	7810	7650	3940
2	3250	0	2930	10000	4870	13500	5480	3860	6770	1280
3	6530	2930	0	10120	7940	13070	10610	5410	9180	4050
4	9000	10000	10120	0	13690	6000	11900	14500	15100	10540
5	5005	4870	7940	13690	0	15300	5990	5970	2750	4030
6	10007	13500	13070	6000	15300	0	11100	14560	14600	10480
7	6680	5480	10610	11900	5990	11100	0	9070	4690	6500
8	7810	3860	5410	14500	5970	14560	9070	0	8300	4670
9	7650	6770	9180	15100	2750	14600	4690	8300	0	5010
10	3940	1280	4050	10540	4030	10480	6500	4670	5010	0

stations of St.Petersburg: 2 – Tekhnologicheskiy Institut, 3 – Moskovskie Vorota, 4 – Lomonosovskaya, 5 – Vasileostrovskaya, 6 – Prospekt Bol'shevikov, 7 – Ploschad' Lenina, 8 – Narvskaja, 9 – Chkalovskaja and 10 – Sennaja Ploschad'. Distances between ATMs and the Bank are given in the Table 1.

Time windows for each ATM are given in the Table 2.

Table2: Time windows, h

i	2	3	4	5	6	7	8	9	10
a_i	10	11	10	11	13	13	10	10	10
b_i	13	18	13	18	18	16	13	18	13

Suppose that working day of money collector teams starts at 10:00 and ends at 18:00, which means that $[a_0, b_0] = [10, 18]$, and average speed of teams is $v_a = 20$ km/h. We also take into account traffic, route features, etc.

We construct the initial solution using the nearest neighbor heuristic with parameters $\delta_1 = 0.4$, $\delta_2 = 0.4$, $\delta_3 = 0.2$. Routes, which were constructed are represented in the Table 3 and Figure 1. The distance travelled on these routes corresponds to 71657 meters. The initial solution was simulated in Maple system.

Table3: The initial solution

Route 1	0-2-10-5-9-0
Route 2	0-3-8-7-0
Route 3	0-4-6-0

Then we apply the simulated annealing heuristic for this initial solution with given parameters of the initial temperature $T = 1000$, the cooling ratio $\alpha = 0.99$, and the epoch length $Len = 500$. Routes that we received in Maple system are represented in the Table 4 and Figure 2.

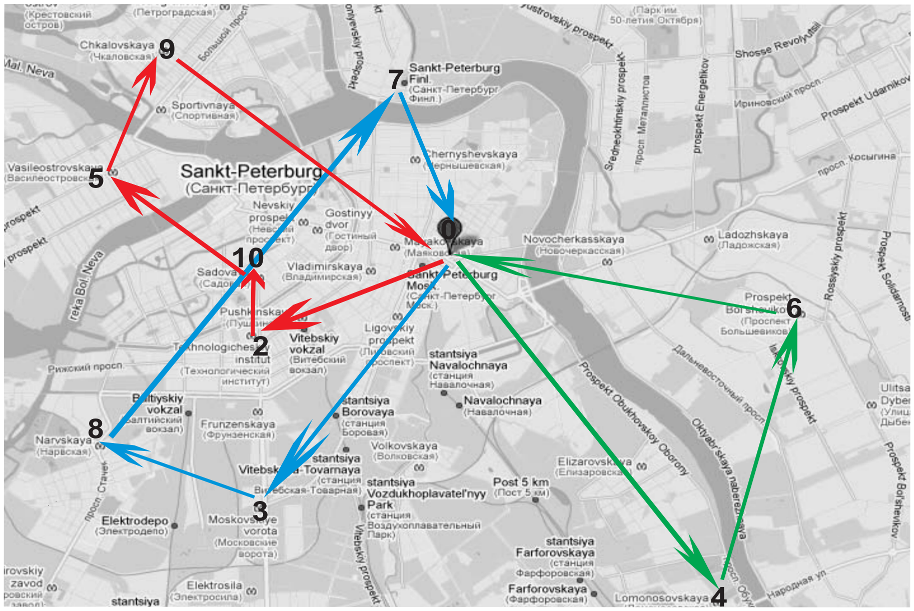

Figure1: The initial solution

Table4: Solution obtained by simulated annealing algorithm

Route 1	0-10-5-9-7-0
Route 2	0-3-8-2-0
Route 3	0-4-6-0

The optimal solution in the current model consists of three routes, one for each collector team. The first team drives through ATMs 10-5-9-7 (subway stations: Sennaja Ploschad', Vasileostrovskaya, Chkalovskaja, Ploschad' Lenina), the second team goes through ATMs 3-8-2(subway stations: Moskovskie Vorota, Narvskaja, Tekhnologicheskiy Institut) and the third team goes through ATMs 4-6 (subway stations: Lomonosovskaya, Prospekt Bol'shevikov). Every route begins and ends at the bank, vehicle capacity on each route is not exceeded, and time windows are satisfied (see Tables 2 and 5).

Table5: Time of beginning servicing ATMs, h

	Route 1				Route 2			Route 3	
i	10	5	9	7	3	8	2	4	6
g_i	10:12	11:00	11:39	13:00	11:00	11:47	12:28	10:27	13:00

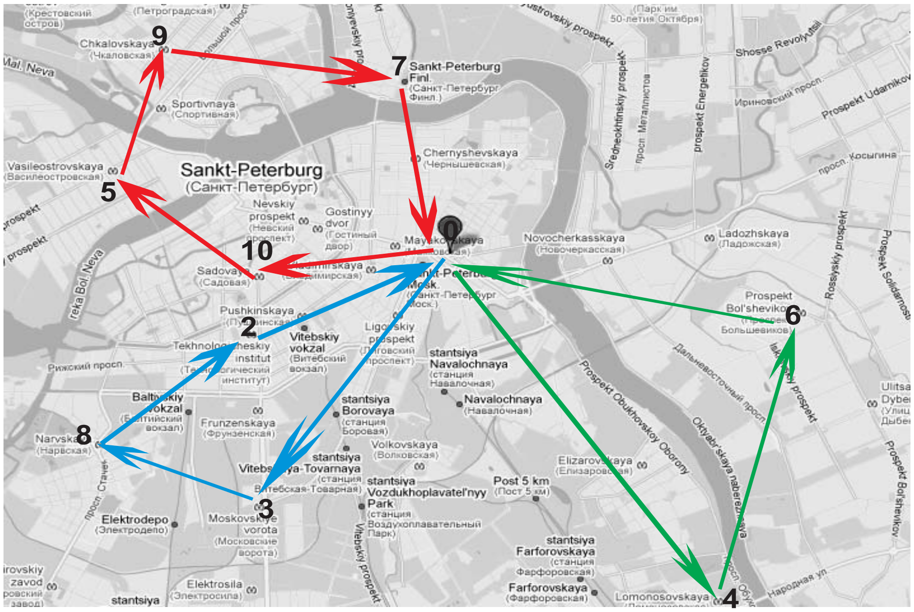

Figure2: Solution obtained by simulated annealing algorithm

The first money collector team returns to the bank at 13:20, the second — at 12:38, the third — at 13:30. That means that the time window of working day is also satisfied.

All ATMs are assigned to a route and total travel costs are minimized. Thus, we got optimal routes for the current request. The distance travelled on this optimal route corresponds to 66147 meters, this is a minimal length of all possible routes for the money collector teams.

References

Clarke, G., Wright, J. W. (1964). *Scheduling of Vehicles from a Central Depot to a Number of Delivery Points*, Operations Research, **12**, 568–58.

Chiang, W. C., Russell, R. A. (1996). *Simulated annealing metaheuristics for the vehicle routing problem with time windows.* Annals of Operations Research., **63**, 3–27.

Gubar, E., Zubareva, M., Merzljakova, J. (2011). *Cash Flow Optimization in ATM Network Model.* Contributions to Game Theory and Management. Vol. 4. Eds. Leon A. Petrosyan, Nikolay A. Zenkevich – SPB.: Graduate School of Management, SPbU.

Hall, R. W. (2003). *Handbook of Transportation Science.* Springer.

Solomon, M. M. (1987). *Algorithms for the vehicle routing and scheduling problems with time window constraints.* Operations Research, **35(2)**, 254–265.

Hamdy A. Taha. (2003). *Operations Research: an Introduction.* University of Arkansas, Fayetteville: Pearson Education, Inc.

Toth, P.,Vigo, D. (2001). *The vehicle routing problem.* Society for Industrial and Applied Mathematics. Philadelphia, PA, USA.

Woch, M., Lebkowski, P. (2009). *Sequential simulated annealing for the vehicle routing problem with time windows.* Decision Making in Manufacturing and Services, **3(1-2)**, 87–100.