



# An Assignment Heuristic for Time-Dependent Periodic Routing Problems with Complex Constraints

Tomasz Śliwiński\*

*Abstract.* Periodic routing and scheduling is of the utmost importance in many industries with mobile personnel working in the field: sales representatives, service technicians, suppliers, etc. In many cases, the long-term stability of the customer to salesman assignment is required, leading to the decomposition of the major problem into single salesman subproblems. The paper addresses the assignment of customers to salesmen for the future services performed in a periodic fashion. It can be seen as the decomposition phase of the periodic vehicle routing problem PVRP into a number of Periodic Traveling Salesman Problems (PTSP). The proposed algorithm seeks the best assignment by taking into account diverse system requirements, constraints and expected operational costs including time windows, time-dependent travel times and costs, and labor laws.

*Keywords:* periodic vehicle routing, time windows, assignment problem

*Mathematics Subject Classification:* 90B06

*Submitted:* October 12, 2017

*Revised:* January 31, 2019

© 2020 Author. This is an open access publication, which can be used, distributed and reproduced in any medium according to the Creative Commons CC-BY 4.0 License. License requiring that the original work has been properly cited.

## 1. INTRODUCTION

Under consideration is the uncapacitated Periodic Vehicle Routing Problem (PVRP) with multiple time windows, time-dependent travel costs and times, multiple salesmen depot and labor laws. In such a problem each customer has to be visited periodically by a salesman over some planning horizon. Unlike many problems found in the literature, where each customer is visited daily or the required number of times chosen among a set of valid day combinations, here the cycle of visits for the single customer is specified by the time intervals between consecutive visits or/and by setting the range of days when particular visit should take place. While the allowed range of days states a hard constraint, the deviation from the specified time intervals is punished with penalty functions, being effectively a soft constraint.

---

\* Warsaw University of Technology, Institute of Control and Computation Engineering, Warsaw, Poland, e-mail: T.Sliwinski@elka.pw.edu.pl

Each customer can have one or several time windows defined for each day of the planning horizon when the visit should take place. The travel times and costs can be asymmetric and depend on the time of a day and the day of a week. This allows for a better traffic modeling as compared to their symmetric and constant values. All salesmen can operate from different depots. It is assumed, that each salesman can make no more than one trip a day, and each day can have different starting and finishing depots, although the selection of those depots is the input parameter of the problem. The salesmen working time is governed by the relevant labor law, which specifies basic working hours and maximum working hours, daily and weekly. For the basic working hours, the given unit cost is charged. If the basic working hours are exceeded, daily or weekly, overtime is charged. The specified maximum working hours cannot be exceeded.

The paper addresses a particular PVRP in which the long term stability of relations between the salesman and the customer plays an important role. In such a case the initial assignment of customers to salesmen can determine the later efficiency of creating final schedules. The assignment can be seen as a decomposition of the main problem into many subproblems, one for each salesman. The difficulty of such an approach lies in the nondeterministic nature of the future schedules, as it is not possible to predict exact requirements, time schedules and routes in the future. Another stumbling block is the size of the assignment problem – dozens of salesmen, hundreds or even thousands of customers, long planning horizon.

The problem is important to many industries where salesmen (field workers, sales representatives or maintenance teams) are committed to visiting customers on the regular basis in the specific time intervals. Their work schedule must conform to many requirements and regulations, like labor law, time windows of the customers or customer-dependent visit intervals. It should also take into account various goals, with travel and labor costs being the most important ones.

Solving the problem optimally would involve assigning customers to salesmen and solving all the underlying PTSP problems for each salesman over a long scheduling horizon.

Considering all the requirements and constraints, the single underlying PTSP problem can be shortly described as Time Dependent Asymmetric Periodic Traveling Salesman Problem with Multiple Time Windows, Multiple Depots and Labor Laws.

Different extensions to the Traveling Salesman Problem (TSP) were considered in the literature. The Asymmetric TSP (ATSP) with Time Windows have been studied intensively in the literature (Gendreau *et al.*, 1998; Ascheuer *et al.*, 2000, Ascheuer *et al.*, 2001; Focacci *et al.*, 2002) and respective optimization algorithms have been proposed. Extending the problem by adding time-dependent travel times and costs makes it much more difficult. Most approaches focus on using different heuristic procedures Hurkała (2015). The exact solution to the problem proposed in Albiach *et al.* (2008) is based on the graph transformation into an ATSP for which a standard optimization procedure can be used. Unfortunately, adding periodicity to the above problem prevents us from applying this approach. The reason is the size of the problem (for example, 14 days of the scheduling horizon with a moderate 20 visits a day would require data structures with approx. 280 nodes, depending on the periodicity settings

and specific settings on the time intervals between consecutive visits at the same point). On the other hand, PTSP with Multiple Time Windows is recognized in the literature. For example, in Tricoire *et al.* (2010), the authors utilize a heuristic approach with a VNS meta-heuristic to drive the changes of the current solution. Unfortunately, the authors do not take into account the data and rules that are important from the business perspective, namely time dependent travel times, advanced interval definitions and various labor law regulations with impeding costs.

As an approximate solution to the considered single PTSP problem was proposed in Ogryczak *et al.* (2018). It utilizes a multi stage heuristic approach with 4 stages: construction heuristic to find initial solution, Variable Neighborhood Search (VNS) algorithm to improve global solution for the whole horizon, different algorithms to improve solutions separately for each day (VNS, simulated annealing, threshold accepting) and, finally, specialized, greedy heuristics to improve the final solution.

As for the Periodic Vehicle Routing Problems, Francis *et al.* (2008) present an overview of different extensions with synopsis of modeling and solution methods. In the literature, mostly problems with relatively short time horizons are considered. Usually, the specified number of visits should take place on a day from the set of valid day combinations (for examples see Cacchiani *et al.*, 2014; Cordeau *et al.*, 2001; Norouzi *et al.*, 2015; Michallet *et al.* 2014). If, however, the time for the customer to be visited is flexible, the cardinality of the set of possible day combinations grows exponentially. Problems with longer planning horizons, such as maintenance or regular service visits, are considered in Bostel *et al.* (2008). The authors designed plans over a multi-period, rolling horizon, with daily updates. Peng *et al.* (2013) consider periodic inspections in railroad networks with various constraints and requirements. They apply interval flexibility by setting maximum allowed interval between visits. The problem of the long-term assignment of customers to salesmen seems to be absent in the literature of the PVRP problems. However, this requirement is present in many business activities.

In the following section, the procedure of assigning customers to salesmen is presented. It is based on the construction heuristics for the fast building of the hypothetical future schedules combined with the assignment schema. In Section 2 we introduce the approximate assignment heuristic giving details of the applied simplified scheduling algorithms. In Section 3 we present the results of the computational experiments.

## 2. LONG TERM ASSIGNMENT OF CUSTOMERS TO SALESMEN

Due to its enormous size, the specific assignment problem would be hard to solve optimally, that is why we propose an approximate approach based on a random sampling procedure to iteratively restrict the sets of salesmen allowed to visit the customers. The restriction decision is based on the statistical properties of the sampled solutions.

The algorithm's goal is to create an assignment that tries to minimize the expected long-term system operation cost, including time dependent travel costs and labor costs.

## 2.1. Input data

It is crucial that as many different input data and parameters as possible are considered, so that the resulting simulated schedules are a good approximate of the real ones. The major scheduling objects are visits. Each visit  $i \in I$  has a number of direct properties: its preassigned customer, duration, range of days when the visit can take place, and indirect properties, derived from preassigned customer properties, like customer location, time windows allowed for the visit, time-dependent travel costs and times, intervals between consecutive visits to the same customer. Below is the list of all indices and input data utilized by the algorithm.

$d \in \{1, \dots,  D \}$	days of the scheduling horizon
$w \in W$	weeks in the scheduling horizon
$p \in P$	customers, also customer locations
$q \in Q$	salesmen
$i \in I$	visits
$I_p$	set of visits to the customer $p$
$p(i)$	customer served in visit $i$
$\Theta_i^d$	set of time windows for customer $p(i)$ on day $d$
$\Lambda_{ij}^d$	set of traffic time zones for path $(p(i), p(j))$ on day $d$
$b_q^d$	starting visit (depot) for the salesman $q$ on day $d$
$f_q^d$	final visit (depot) for the salesman $q$ on day $d$
$\tau_i^{dk} = [a_i^{dk}, b_i^{dk}]$	$k$ -th time window for customer $p(i)$ on day $d$ ; $k \in \Theta_i^d$
$\sigma_{ij}^{dn} = [c_{ij}^{dn}, d_{ij}^{dn}]$	$n$ -th traffic time zone for path $(p(i), p(j))$ on day $d$
$s_i$	duration visit $i$
$t_{ij}^{dn}$	travel time on path $(p(i), p(j))$ on day $d$ in traffic time zone $n$
$c_{ij}^{dn}$	travel cost on path $(p(i), p(j))$ on day $d$ in traffic time zone $n$
$T_b^d, T_m^d$	basic and maximum working hours on day $d$
$W_b^d, W_o^d$	unit cost for basic working hours and for overtime hours on day $d$
$T_b^w, T_m^w$	basic and maximum working hours in week $w$
$W_o^w$	unit cost for overtime hours in week $w$
$a_p$	required interval between subsequent visits to customer $p$
$P_p^+(\delta)$	piecewise linear penalty function of excess $\delta$ over the interval $a_p$
$P_p^-(\delta)$	piecewise linear penalty function of shortage $\delta$ to the interval $a_p$
$r_i = [d_i^{min}, d_i^{max}]$	possible range of days for visit $i$
$w_p$	point importance/weight
$Q_p^0$	initial set of salesmen allowed to visit customer $p$ ; the set can reflect some salesman skills needed by a particular customer
$L_p^0$	day of the last visit to point $p$

As it can be seen, most parameters can have different values for each day of the scheduling horizon. In particular one can introduce different labor costs depending on the week day and/or season. The same applies to customer time windows – they can be set differently for each day of the week.

Unfortunately, such a detailed parameter setting may lead to a very high memory footprint for the application, such as when the travel times and costs are considered. That is why a single travel cost and times matrix was utilized in the experiments for all days of the scheduling horizon. The matrix was computed as the average of the data for all week days. In future implementations it is possible, however, to utilize the information on the traffic changes over all days of the scheduling horizon.

While the main decision variables of the considered PVRP are related to the actual assignment of customers to salesmen, other variables describe schedules of the single PTSP:

- $b_{pq}$  binary variable = 1 only if customer  $p$  is assigned to salesman  $q$ ;
- $y_{iq}^{dk}$  binary variable = 1 only if visit  $i$  is performed by salesman  $q$  on day  $d$  within window  $k$  of point  $p(i)$ ;
- $x_{ij}^d$  binary variable = 1 only if after visit  $i$  on day  $d$  visit  $j$  is performed;
- $\beta_i^d$  leaving time after completing visit on day  $id$ ;
- $v_{ij}^{dn}$  binary variable = 1 only if travel between visits  $i$  and  $j$  on day  $d$  is performed within traffic time zone  $n$ .

As the number of visits over longer horizon is large, the solution space is tremendous and intractable by using exact approaches.

## 2.2. The approximate assignment

When assigning customers to salesmen for long-term service, not all input parameters can be known in advance. The idea is to sample some of the parameter values and approximately solve the problem for such instances. The results of the simplified scheduling algorithm can differ significantly for each instance. One can anticipate, however, the statistical properties of the schedules created in many runs will depend on the properties of the long term assignment, when customer's location and visiting frequencies are considered. Based on this, the following algorithm for the long term assignment is proposed.

1. Initialize. Prepare input data including initial assignment  $Q_p^0; k := 0$ .
2. Sample of the problem input parameters multiple times and solve simplified assignment and scheduling for a long time horizon. Each time only salesman  $q \in Q_p^k$  can be scheduled for a visit to customer  $p$ . Collecting statistics on the number of visits performed by each salesman to each customer.
3. Select a salesman in the set  $Q_p^k$  with the smallest total number of visits to customer  $p$  over all samples. If  $|Q_p^k| > 1$ , the salesman is removed from the set  $Q_p^k$ .
4. If  $|Q_p^k| = 1 \forall p$ , then STOP, otherwise  $k := k + 1$ , go to 2.

In each iteration, the algorithm restricts the sets of salesmen allowed to visit the customers. The decision is based on the properties of the generated schedule (total

number of the visits to the given customer). The sampling of the input parameters is performed with a different random seed. The total number of iterations equals  $\max_p(|Q_p^0| - 1)$ , as each time only one salesman is removed from each set.

### 2.3. Sampling over the input parameters

Depending on the actual business situation, the values of the most input parameters do not need to be changed frequently, so it can be assumed to be constant over the whole planning horizon. But the schedule calculated for the assignment will not hold for a long time, as any random disturbances will affect its implementation and force to perform new scheduling for single PTSP. Additionally, due to the problem size, we are not even able to create final schedules of good quality. Instead, we want to sample from a number of simplified random schedules created for different random situations and, based on their properties, infer the best assignment minimizing expected cost of future operations.

Seeing each visit as the actual object to be scheduled, we generate a random set of visits and their parameters for each customer to cover the whole planning horizon, based on the problem's input parameters.

Let  $d_p^0$  be a random number in the range  $[0, a_p - 1]$  generated with the uniform distribution, once for each customer. Then, day ranges for all subsequent visits to the same customer (numbered with  $k = \lceil \frac{H}{a_p} \rceil$ ), where  $H$  is the length of the planning horizon) are computed as  $[d_p^0 + ka_p - \frac{a_p}{2}, d_p^0 + ka_p + \frac{a_p}{2}]$ . Certainly, the range size relative to  $a_p$  can be set individually for each customer, but for our computational tests the constant value was chosen. All other visit parameters, like visit duration  $s_p$  or penalty functions  $P_p^+(\delta)$  and  $P_p^-(\delta)$ , are constant over the whole planning horizon.

### 2.4. Simplified scheduling

As we utilize a construction heuristic to create the simplified schedule, an initial ordering of visits is required. We used the following properties to enforce the order of the visits:

- 1) value of  $d_i^{max}$  (latest possible visit day), earliest first;
- 2) penalty for possible delay on day  $d_i^{max}$ , highest first;
- 3) customer importance  $w_p$ , highest first.

The properties are enumerated according to their significance (with (1) being the most important one). The visits are ordered lexicographically, which means that only if the more significant properties are equal would the less significant property be considered.

The scheduling problem can be enormous (dozens of salesmen, thousands of customers), and a fast and effective optimization procedure is required. It has to be sensitive to all the problem properties, with customer locations and frequency of visits being the most important ones.

The applied greedy construction heuristic tries to insert each visit from the ordered list in the best possible place of the route created separately for each salesman. In

detail, let  $Q_p$  be the set of salesmen allowed to visit customer  $p$ . Taking the first visit  $i$  from the ordered list, the algorithm examines all salesmen from the set  $Q_{p(i)}$  and tries to insert the visit in the partially created routes of each salesman. Only days from the day range  $r_i$  are considered.

The visit is inserted temporarily in all possible places of the route without changing the sequence of the previously inserted visits. Each time the optimal schedule for the given route is computed (see Section 2.5). Finally, the visit is inserted in a place (salesman and the point of its partial route) where the total cost increase is the smallest (Fig. 1).

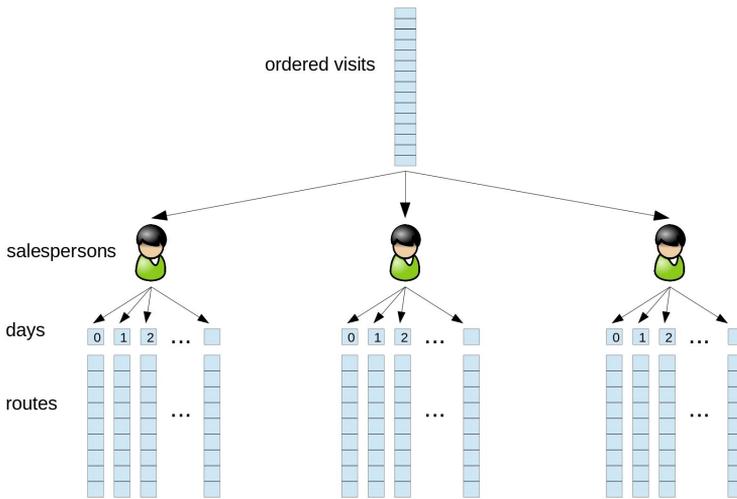


Fig. 1. Illustrative drawing of the scheduling procedure

One should keep in mind that the route of the salesman  $q$  constructed on day  $d$  should begin with designated visit  $b_q^d$  and end with  $f_q^d$ , thus different starting and final depots on each day are possible.

### 2.5. Finding the optimal schedule for the given route

As opposed to classical TSP, problems with time dependent travel costs and times, and time windows require specialized approach to computing the detailed schedule, including the total travel cost and time, for the given route.

The problem has been addressed in the literature (Savelsbergh 1992; Jong *et al.*, 1996). An exact algorithm of finding the minimum route duration for time dependent travel with multiple time windows was presented in Hurkała (2015). After the preprocessing phase which eliminates some of the time windows, the algorithm iteratively evaluates schedules and chooses ones with the shortest duration. The procedure actively eliminates the dominated solutions, substantially reducing the number of the computations.

Approximate approaches are also possible. A simple yet very efficient one probes different travel starting times. It is assumed the total travel duration and cost only depends on the time of the travel start, and the resulting schedule is computed by the simulation of the travel over the successive points on the route. This simple procedure was chosen as the main algorithm for finding an optimal schedule in the assignment procedure. The experiments showed it performs exceptionally well with the real-world data reaching a level of performance two orders of magnitude better than achieved by the exact approach.

## 2.6. Possible extensions/modifications

The algorithm is very flexible and allows application in various real world situations. For example, it can be easily modified to assign customers to a new salesman or rearrange the assignment according to some new requirements. The stopping criterion can also be extended to allow a limited (but greater than 1) number of employees visiting single client.

This should increase the total system performance, but at the cost of weaker customer relations. The algorithm makes possible to include personal travel characteristics – time and costs (in the experiments uniform data were used for all salesmen). The data should be available after a longer period of collecting real travel data.

One of the concerns with the algorithm is the uneven load allocation between salesmen. It is partially limited with maximum daily and weekly working hours. If better load balancing is required, it can easily be incorporated into the simplified scheduling algorithm, extending the cost criterion with some over-/uneven load costs.

## 3. COMPUTATIONAL EXPERIMENTS

### 3.1. Data generation

Three main parameters control the generation of random data: the number of salesmen, horizon length and geographical area (all of Warsaw or only Ochota district). Different combinations of those parameters were tested in the computational experiments.

Labor costs and regulations are set the same for all salesmen (maximum 10 hours of working time a day and 40 hours a week). The number of customers is calculated so that approximate estimated daily travel time should be shorter than 7 hours. This resulted in an average of 24 customers per salesman per day for the district Ochota, and 15 customers per salesman per day for the whole city. The remaining input data were generated as follows.

- Time windows for customers (generated uniformly in the range 7–20 with 30% probability, 7–14 with 25%, 12–20 with 25% probability, 7–11 and 14–20 with 20% probability).
- Customer geographical coordinates were selected from the publicly available data of small businesses to cover selected area (all of Warsaw or only the Ochota district).

- Travel time and costs are based on the real travel data acquired from the commercial provider.
- Intervals between visits for each customer (a number generated uniformly in the range 2–5 with 30% probability, 5–15 with 40% probability, 10–25 with 30% probability).
- Being late penalty: piecewise linear function with 3 segments defined by points (0, separation/4, separation/2, separation) with a gradient 0 for the first segment, gradient 40 PLN/separation for the second segment and gradient 160 PLN/separation for the third segment.
- Being early penalty: piecewise linear function with 3 segments defined by points (separation, –separation/2, –separation/4, 0) with gradient –160 PLN/separation, for the first segment, –40 PLN/separation for the second segment and 0 for the third segment.
- Day ranges for each visit are generated as described in Section 2.3.
- All visits have the same duration: 7.5 minutes.

### 3.2. Simplified assignment algorithm for comparison purposes

We wanted to compare the proposed assignment algorithm to some clustering heuristics available for the VRP problems: Fisher and Jaikumar (1981), Bramel and Simchi-Levi (1995), Ryan *et al.* (1993), Reanaud *et al.* (1996). Unfortunately, most of them were designed for relatively small problem instances (up to 20 vehicles and 200 customers). In the problems concerned here, 5,000 customers must be assigned to 20 salesmen. That is why a simplified approach had to be constructed. The proposed algorithm creates a single long route for each salesman where each customer appears only once. The input data are substantially simplified: no time dependent travel time/costs, no time windows, no visits intervals (although, the visits frequency is used for the customer ordering), no labor costs and regulations, single depot. The algorithm consists of the following steps.

1. The preprocessing phase – the customers are ordered according to their visit frequencies (first the ones visited more often). For each salesman, a single initial customer is selected using standard MacQueen initialization method for the k-means algorithm MacQueen (1967) and initial route from the depot to the single customer is created.
2. The algorithm takes the first customer from the ordered list and tries to insert it in the partially created route of each salesman. The route and the insertion position are chosen with the least cost/time increase. No detailed schedule has to be found for the given route.
3. The algorithm stops when all customers are assigned.

The simplified algorithm is similar to the proposed assignment schema, as also here the route construction is used to assign the customer to the salesman. The major difference is that no visit frequency is used and the assignment is based on a single route.

3.3. The results

The test results cover different numbers of salesmen and different lengths of the scheduling horizon. All the tests were performed on a computer with 2.4GHz i7 series Intel processor.

The values are the average of 10 tests. In Table 1, the computing times are presented. The algorithm performs acceptably, although the exponential complexity as a function of the salesmen number can be seen. The horizon’s length affects the computation times linearly.

**Table 1.** *Computing times [s]*

Area		All of Warsaw				Ochota district			
Horizon length [min.]		30	60	120	180	30	60	120	180
No. of salesmen	5	0.3	0.5	0.8	2.1	3.8	5.2	9.0	15.3
	10	3.2	6.8	13.2	12.3	25.3	38.3	85.3	134.5
	20	40.0	58.0	130.0	252.0	192.3	385.3	815.4	1252.3

To evaluate the assignment algorithm, one needs a quality measure reflecting its desired properties. In the considered situation, the best quality measure seemed to be the total cost of the exact schedules computed separately for each salesman and his assigned customers over the whole scheduling horizon. We utilize the algorithm introduced in Ogryczak *et al.* (2018).

The proposed assignment algorithm (its quality measure) is compared to the procedure where each customer is assigned to a salesman on a random basis, and to the simplified assignment described above. The relative improvements are presented in Tables 2 and 3, respectively.

One may notice that the proposed assignment procedure performs best, showing the advantage of utilizing all the available information. The length of the scheduling horizon improves the quality of the assignment, most probably due to the better (more data) estimation of the expected quality measure of the final schedule.

**Table 2.** *Relative improvement over fully random assignment [%]*

Area		All of Warsaw				Ochota district			
Horizon length [min.]		30	60	120	180	30	60	120	180
No. of salesmen	5	1.3	5.3	2.4	10.0	3.8	4.8	8.3	4.5
	10	5.0	7.3	11.9	16.6	9.3	13.0	19.1	20.0
	20	7.7	14.8	23.0	25.5	10.3	17.8	23.3	24.0

**Table 3.** Relative improvement over simplified random assignment [%]

Area		All of Warsaw				Ochota district			
Horizon length [min.]		30	60	120	180	30	60	120	180
No. of salesmen	5	0.3	3.7	1.8	7.0	2.2	2.3	4.8	3.2
	10	1.2	4.3	3.1	4.3	5.2	8.7	10.9	11.8
	20	4.2	9.2	8.2	9.3	7.8	7.3	12.9	14.2

#### 4. CONCLUSION

The presented long term customer to salesman assignment for periodic routing problems offers a simple yet effective way of optimizing expected overall system costs. The algorithm combines simple construction heuristics for the schedule simulation with the iterative assignment procedure to effectively compute final allocation. The computational experiments show that the algorithm can be applied to large scale problems with various real life constraints and requirements.

#### ACKNOWLEDGMENTS

This research was financed by the European Union through the European Regional Development Fund under the Operational Programme “Innovative Economy” for the years 2007–2013; Priority 1 Research and development of modern technologies under the project POIG.01.03.01-14-076/12: “Decision Support System for Large-Scale Periodic Vehicle Routing and Scheduling Problems with Complex Constraints”.

#### REFERENCES

- Albiach, J., Sanchis, J.,M., Soler, D., 2008. An asymmetric tsp with time windows and with time-dependent travel times and costs: An exact solution through a graph transformation. *European Journal of Operational Research*, 189(3), pp. 789–802.
- Ascheuer, N., Fischetti, M., Groetschel, M., 2000. A polyhedral study of the asymmetric traveling salesman problem with time windows. *Networks*, 36, pp. 69–79.
- Ascheuer, N., Fischetti, M., Groetschel, M., 2001. Solving the asymmetric travelling salesman problem with time windows by branch-and-cut. *Mathematical Programming Series A*, 90, pp. 475–506.
- Bostel, N., Dejax, P., Guez, P., Tricoire, F., 2008. Multiperiod Planning and Routing on a Rolling Horizon for Field Force Optimization Logistics. In: Golden, B., Raghavan, S., Wasil, E. (eds.), *The vehicle routing problem: latest advances and new challenges*. Springer, Berlin, pp. 503–527.
- Bramel, J., Simchi-Levi, D., 1995. A location based heuristic for general routing problems. *Operations Research*, 43(4), pp. 649–660.

- Cacchiani, V., Hemmelmayr, V.C., Tricoire, F., 2014. A set-covering based heuristic algorithm for the periodic vehicle routing problem. *Discrete Applied Mathematics*, 163(1), pp. 53–64.
- Cordeau, J.-F., Laporte, G., Mercier, A., 2001. A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, 52(8), pp. 928–936.
- Fisher, M., Jaikumar, R., 1981. A generalized assignment heuristic for vehicle routing. *Networks*, 11(2), pp. 109–124.
- Focacci, F., Lodi, A., Milano, M., 2002. A hybrid exact algorithm for the tsptw. *INFORMS Journal on Computing*, 14(4), pp. 403–417.
- Francis, P.M., Smilowitz, K.R., Tzur, M., 2008. The Period Vehicle Routing Problem and its Extensions. In: Golden, B., Raghavan, S., Wasil, E. (eds.), *The vehicle routing problem: latest advances and new challenges*. Springer, Berlin, pp. 73–102.
- Gendreau, M., Hertz, A., Laporte, G., Stan, M., 1998. A generalized insertion heuristic for the traveling salesman problem with time windows. *Operations Research*, 46(3), pp. 330–335.
- Hurkała, J., 2015. Time-dependent traveling salesman problem with multiple time windows. *Annals of Computer Science and Information Systems*, 6, 71–78.
- Jong, C., Kant, G., Vliet, A., van, 1996. On finding minimal route duration in the vehicle routing problem with multiple time windows. *Technical Report*, Department of Computer Science, Utrecht University, The Netherlands.
- MacQueen, J.B., 1967. Some methods for classification and analysis of multivariate observation. *Proceedings of Fifth Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1. University of California Press, Berkeley, pp. 281–297.
- Michallet, J., Prins, Ch., Amodeo, L., Yalaoui, F., Vitry, G., 2014. Multi-start iterated local search for the periodic vehicle routing problem with time windows and time spread constraints on services. *Computers & Operations Research*, 41, pp. 196–207.
- Norouzi, N., Sadegh-Amalnick, M., Alinaghiyan, M., 2015. Evaluating of the particle swarm optimization in a periodic vehicle routing problem. *Measurement*, 62, pp. 162–169.
- Ogryczak, W., Śliwiński, T., Hurkała, J., Kaleta, M., Kozłowski, B., Pałka, P., 2018. Large-Scale Periodic Routing Problems for Supporting Planning of Mobile Personnel Tasks. In: Atanassov, K.T., Kacprzyk, J., Kaluszko, A., Krawczak, M., Owsiański, J., Sotirov, S.S., Sotirova, E., Zadrożny, S. (eds.), *Uncertainty and Imprecision in Decision Making and Decision Support: Cross-Fertilization, New Models and Applications*, Springer, pp. 205–216.
- Peng, F., Ouyang, Y., Somani, K., 2013. Optimal routing and scheduling of periodic inspections in large-scale railroad networks. *Journal of Rail Transport Planning & Management*, 3(4), pp. 163–171.
- Reanaud, J., Boctor, F., Laporte, G., 1996. An improved petal heuristic for the vehicle routing problem. *Journal of Operational Research Society*, 47(2), pp. 329–336.
- Ryan, D.M., Hjorring, C., Glover, F., 1993. Extensions of the petal method for vehicle routing. *Journal of Operational Research Society*, 44(3), pp. 289–296.
- Savelsbergh, M., 1992. The vehicle routing problem with time windows: Minimizing route duration. *ORSA Journal on Computing*, 4(2), pp. 146–154.
- Tricoire, F., Romauch, M., Doerner, K.F., Hartl, R.F., 2010. Heuristics for the multiperiod orienteering problem with multiple time windows. *Computers & Operations Research*, 37(2), pp. 351–367.