

DEPARTMENT OF ENVIRONMENT,
TECHNOLOGY AND TECHNOLOGY MANAGEMENT

A metaheuristic for a teaching assistant assignment-routing problem

Pablo Maya, Kenneth Sörensen & Peter Goos

UNIVERSITY OF ANTWERP
Faculty of Applied Economics



Stadscampus
Prinsstraat 13, B.213
BE-2000 Antwerpen
Tel. +32 (0)3 265 40 32
Fax +32 (0)3 265 47 99
<http://www.ua.ac.be/tew>

FACULTY OF APPLIED ECONOMICS

DEPARTMENT OF ENVIRONMENT,
TECHNOLOGY AND TECHNOLOGY MANAGEMENT

A metaheuristic for a teaching assistant assignment-routing problem

Pablo Maya, Kenneth Sörensen & Peter Goos

RESEARCH PAPER 2009-011
NOVEMBER 2009

University of Antwerp, City Campus, Prinsstraat 13, B-2000 Antwerp, Belgium
Research Administration – room B.213
phone: (32) 3 265 40 32
fax: (32) 3 265 47 99
e-mail: joeri.nys@ua.ac.be

The papers can be also found at our website:
www.ua.ac.be/tew
(research > working papers)

D/2009/1169/011

A metaheuristic for a teaching assistant assignment–routing problem

Pablo Maya Kenneth Sörensen
Peter Goos

University of Antwerp, Faculty of Applied Economics

Stadscampus S.B.513 Prinsstraat 13, Antwerp 2000

Tel: +32654061, Fax: +032654901

e-mail: pablo.mayaduque@ua.ac.be

The Flemish Ministry of Education promotes the integrated education of disabled children by providing educational opportunities in common schools. In the current system, disabled children receive ambulant help from a teaching assistant (TA) employed at an institute for extra-ordinary education. The compensation that the TAs receive for driving to visit the students is a major cost factor for the institute that provides the assistance, therefore its management desires a schedule that minimizes the accumulated distance traveled by all TAs combined. We call this optimization problem the teaching assistants assignment routing problem (TAARP). It involves three decisions that have to be taken simultaneously: (1) pupils have to be assigned to TAs; (2) pupils assigned to a given TA have to be spread over the TA's different working days; and (3) the order in which to visit the pupils on each day has to be determined. We propose a solution strategy based on an auction algorithm and a variable neighborhood search which exhibit an excellent performance both in simulated and real instances. The total distance traveled in the solution obtained for the real data set improves the current solution by about 22% which represents a saving of around 9% on the annual budget of the institute.

1 Introduction

To promote the integration of disabled children into the general school system, the Flemish Ministry of Education provides opportunities and special support for those pupils in their own schools. In the current system, disabled children in common schools receive ambulant help from a teaching assistant (TA) employed at an institute for extraordinary education, the royal Institute Woluwe. Three main types of disability are supported: hearing disability, autism spectrum disorder and language disability.

Every pupil that receives assistance is assigned to a TA and is visited at his or her own school. Pupils with a moderate disability are visited once a week, whereas pupils with a severe disability receive assistance twice a week. Every morning, the TAs depart from their homes and travel from school to school to visit the pupils that have been assigned to them for that day. At the end of the working day, the TAs return to their homes. In the beginning of the school year, the TAs receive their roster: a schedule that specifies for each day of the week which pupils they should assist and in which order. The TAs drive their private cars, and receive a financial compensation per driven kilometer. This compensation is a major cost factor for the institute that provides the assistance. Its management therefore desires a schedule that minimizes the accumulated distance traveled by all TAs combined.

Determining the schedule involves three decisions that have to be taken simultaneously: (1) pupils have to be assigned to TAs; (2) pupils assigned to a given TA have to be spread over the TA's different working days; and (3) the order in which to visit the pupils on each day has to be determined. We call this optimization problem the *teaching assistants assignment–routing problem* (TAARP).

Several constraints have to be taken into account when solving the TAARP. First, whether or not a certain TA is allowed to assist a certain pupil depends on her educational degree, on the pupil's disability type, and on the school the pupil is in (nursery, primary school or secondary school). Second, the number of working hours per week is different across TAs. This is due to the fact that certain TAs only work part time, and to an oddity in the Belgian law which prescribes that the number of hours a full time TA has to work depends on her exact diploma. The higher the TA's diploma, the fewer hours she has to work during any given week. Third, pupils that receive assistance twice a week do so at two different and non-consecutive days.

This problem is related to the multi-depot multi-period vehicle routing problem, but has several additional constraints. Some special characteristics of the problem, such as the fact that the teachers' "capacity" (maximum working time) is expressed as a multiple of the unit of "demand" (one hour of teaching), have motivated us to develop a new solution approach for this specific problem. In this paper we therefore propose a new metaheuristic that consists of two phases: a heuristic to find an initial feasible solution, followed by an improving phase. The heuristic for the first phase is inspired by the *auction algorithm*, while the improving phase is a *variable neighborhood search* (VNS)

metaheuristic that considers swaps between two pupils with similar requirements and exchanges that involve more than two pupils with different characteristics (number of required sessions and hours per session).

This paper is organized as follows. Section 2 surveys the relevant literature on related routing problems. In section 3 we develop a mixed-integer programming formulation of the problem. Our solution strategy is proposed in section 4 and tested in section 5. Finally, section 6 contains some conclusions and pointers for future research.

2 Literature Review

Because of the fact that several different starting locations for vehicles are used and that the problem data stretches over several days, the TAARP that we consider in this paper is closely related to the multi depot periodic vehicle routing problem (MDPVRP), which generalizes two other well-known routing problems: the multi depot vehicle routing problem (MDVRP) and the periodic vehicle routing problem (PVRP). The PVRP is a generalization of the VRP the objective of which is to determine routes for a fleet of vehicles over a period of days. Like in the VRP, the vehicles' tours start and end at a given depot. An extensive description of PVRP can be found in a survey by Francis et al. [12], where the evolution of the problem in the literature and its solution methods are presented, starting from the initial identification of the problem by Beltrami and Bodin [1] and its subsequent formulation and first dedicated heuristics by Russell and Igo [23] and Christofides and Beasley [8]. A discussion of the different objectives and constraints is presented in Mourgaya and Vanderbeck [19]. Early heuristics to solve the PVRP consist of a cluster-first route-second approach in which customers are first clustered into days. Examples are the heuristics proposed by Beltrami and Bodin [1], Tan and Beasley [24] and Russell and Gribbin [22]. In some cases these heuristics are complemented with improving phases. Recently approaches based on metaheuristics have been implemented, like the tabu search proposed by Cordeau et al. [9] (also suitable for the MDVRP) and the solution strategy based on genetic algorithms by Drummond et al. [10]. Also, mathematical programming approaches have been presented, like the method based on lagrangian relaxation by Francis et al. [11].

The MDVRP is a generalization of the VRP in which the vehicles can be based in different depots and each route followed by a vehicle must depart from one of those depots and return to it after visiting a sequence of customers. In all other aspects, the problem is completely similar to the VRP. The first approaches for this problem were based on construction and improvement procedures and, in general, used adaptations of known heuristics for the VRP. More recently a search procedure was proposed by Chao et al. [7], and tabu search approaches have been proposed by Renaud et al. [20] and Cordeau et al. [9]. In the latter paper a formulation is presented that shows that the MDVRP is a special case of the PVRP by associating depots with days. Some of the solution approaches for the MDVRP also use a cluster-first route-second strategy (see,

e.g., the clustering procedures described by Giosa et al. [13]). Two exact algorithms have been developed for the MDVRP. Both are due to Laporte et al. [16], but these only work well for relatively small instances in the symmetric and asymmetric cases, respectively.

The combination of periodicity and multiples depots into the MDPVRP has not been extensively studied. Francis et al. [12] define this problem as a variant of the PVRP. Hadjiconstantinou and Baldacci [14] consider the resource planning problem of a utility company that provides preventive maintenance services to a set of a geographically dispersed network of customer using a fleet of depot-based vehicles and crews. They formulate this problem as a MDPVRP and a heuristic based on a generalization of the classical VRP is used to solve it. More recently, Mingozzi [18] presented an integer programming formulation of the MDPVRP that is an extension of the set partitioning formulation of the Capacitated VRP.

3 Mathematical Formulation

The TAARP is defined on a graph where nodes correspond either to TAs' homes or schools where pupils attend classes, while arcs represent routes (distances calculated over a road network) that connect every pair of nodes. In this section, we formulate the TAARP as an integer (binary) program. The symbols used are shown in Table 1.

The model is formulated as follows:

$$\min \sum_{t \in \mathcal{T}} \sum_{d \in \mathcal{D}} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}} c_{ij} x_{ijtd} \quad (1)$$

$$\sum_{i \in \mathcal{N}} x_{ijtd} - \sum_{i \in \mathcal{N}} x_{jtd} = 0 \quad \forall j \in \mathcal{S}, \forall t \in \mathcal{T}, \forall d \in \mathcal{D} \quad (2)$$

$$\sum_{d \in \mathcal{D}} \sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{N}} b_{jt} x_{ijtd} = k_j \quad \forall j \in \mathcal{S} \quad (3)$$

$$\sum_{d \in \mathcal{D}} \sum_{i \in \mathcal{N}} b_{jt} x_{ijtd} \geq k_j + M(y_{jt} - 1) \quad \forall j \in \mathcal{S}, \forall t \in \mathcal{T} \quad (4)$$

$$\sum_{d \in \mathcal{D}} \sum_{i \in \mathcal{N}} b_{jt} x_{ijtd} \leq M y_{jt} \quad \forall j \in \mathcal{S}, \forall t \in \mathcal{T} \quad (5)$$

$$\sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{S}} r_j x_{ijtd} \leq a_{td} \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{D} \quad (6)$$

$$\sum_{i \in \mathcal{T}} \sum_{j \in \mathcal{S}} x_{ijtd} \leq 1 \quad \forall t \in \mathcal{T}, \forall d \in \mathcal{D} \quad (7)$$

$$\sum_{j \in \mathcal{N}} x_{ijtd} + \sum_{j \in \mathcal{N}} x_{ijtd+1} \leq 1 \quad \forall i \in \mathcal{S}, \forall t \in \mathcal{T}, \forall d \in \mathcal{D} \setminus \{|\mathcal{D}|\} \quad (8)$$

Table 1: Symbols used in the MIP formulation of the TAARP

Sets	
\mathcal{S}	Set of pupils
\mathcal{T}	Set of teaching assistants
\mathcal{N}	Set of nodes, $\mathcal{N} = \mathcal{S} \cup \mathcal{T}$
\mathcal{D}	Set of days, $\mathcal{D} = \{1, 2, 3, 4, 5\}$
Parameters	
k_j	Number of times that pupil j has to be visited every week
b_{jt}	$\begin{cases} 1, & \text{If teaching assistant } t \text{ is allowed to visit pupil } j \\ 0, & \text{Otherwise} \end{cases}$
a_{td}	Number of hours of assistance available for teaching assistant t at day d (capacity)
r_j	Number of hours per session that pupil j requires (session duration for pupil j)
c_{ij}	Cost (distance) for traveling from node i to node j
Variables	
x_{ijtd}	$\begin{cases} 1, & \text{If node } i \text{ is visited after node } j \text{ by teaching assistant } t \text{ on day } d \\ 0, & \text{Otherwise} \end{cases}$
y_{jt}	$\begin{cases} 1, & \text{If pupil } j \text{ is assisted by teaching assistant } t \\ 0, & \text{Otherwise} \end{cases}$

$$\sum_{i \in \mathcal{T}} \sum_{j \in \mathcal{C}} x_{ijtd} \leq |\mathcal{C}| - 1 \quad \forall \mathcal{C} \subset \mathcal{S}, 2 \leq |\mathcal{C}| \leq a_{td}, \forall t \in \mathcal{T}, \forall d \in \mathcal{D} \quad (9)$$

$$x_{ijtd} \in \{0, 1\} \quad \forall i \in \mathcal{N}, \forall j \in \mathcal{N}, \forall t \in \mathcal{T}, \forall d \in \mathcal{D} \quad (10)$$

$$y_{jt} \in \{0, 1\} \quad \forall j \in \mathcal{S}, \forall t \in \mathcal{T} \quad (11)$$

The objective function (1) minimizes the total distance traveled by all TAs on all days. Constraints (2) impose that if pupil j is assisted by TA t on day d , then an arc should be traversed by TA t on day d ongoing to pupil j and outgoing from pupil j . Constraints (3) ensure that each pupil is visited as many times as required and that visits are performed only by a TA allowed to visit him/her, while constraints (4) and (5) impose that each pupil is assisted by only one teacher when the pupil requires more than one visit (M is an arbitrarily large constant). Constraints (6) and (7) impose each TA's capacity (maximum number of hours worked on each day) and that each route is executed by exactly one TA respectively. Constraints (8) ensure that for each pupil there is at least one day between two consecutive visits. Constraints (9) correspond to the generalized subtour elimination constraints. Finally, equations (10) and (11) are the integer (binary) constraints.

The model presented involves a large number of variables and the number of subtour elimination constraints in (9) is exponential. Using this model to solve medium and large size instances is not computationally tractable. However, the model is useful to solve small instances in order to compare the results obtained with those of the metaheuristic approach developed in section 4. To solve such instances, we have implemented a cut adding procedure, in which the problem is first solved without considering the subtour elimination constraints (9). At each iteration, the set of violated subtour elimination constraints is determined and these constraints are added to the problem before solving it. This procedure is run iteratively until no more subtours are found. The results of this procedure can be found in section 5.

4 A metaheuristic for the TAARP

In this section, we present an efficient metaheuristic for the TAARP formulation discussed in section 3. The metaheuristic solution approach consists of two procedures. First, a feasible solution is constructed using a procedure inspired by the *auction algorithm*. Next, the constructed solution is improved by a procedure based on *variable neighborhood search* (VNS).

4.1 Initial solution heuristics

Finding a feasible solution of the TAARP is difficult because of the special structure of the problem. Three heuristics to construct an initial solution were implemented.

The first heuristic we implemented is a cluster-first route-second heuristic. The clustering phase of this heuristic uses two of the measures described by Giosa et al. [13] to define a precedence relationship between customers (pupils). This precedence relationship determines the order in which pupils are assigned to TAs. The routing phase involves an insertion algorithm (Campbell and Savelsbergh [5]), where the savings are computed similarly to those presented by Campbell et al. [6].

The second heuristic performs the assignment and routing simultaneously. This heuristic is analogous to the application of the routing phase of the first heuristic applied to the complete set of pupils and without clustering the pupils first. In each iteration the insertion algorithm evaluates the insertion of the pupil in each possible route of each teacher and inserts a pupil in its best possible location.

Neither of these two heuristics provided satisfactory results. We have attempted to improve the performance of the two mentioned heuristics by introducing randomness in a GRASP-like fashion (Resende and Ribeiro [21]). However, even then the effectiveness of these two heuristics proposed is low as they typically end without having found a feasible solution. This is due to the combined effect of the periodicity and the requirement that all the sessions for one pupil must be provided by the same TA.

4.2 A heuristic based on the auction algorithm

We propose a third heuristic that focuses on trying to assign the pupils to TAs in such a way that a feasible solution is found. In order to achieve that objective, we first tackle the problem of assigning pupils to teachers ensuring that all constraints except those related with the routing are fulfilled. Note that after a feasible solution for the assignment problem is found, it can be trivially transformed into a feasible solution for the TAARP by determining an arbitrary routing of the pupils assigned to each working day of each teacher. In other words, the routing does not affect the feasibility of the solution once it has been reached during the assignment phase. Our heuristic is inspired by the auction algorithm which was initially proposed by Bertsekas [2] for the assignment problem. We first provide a description of the basic idea behind the auction algorithm. Next, we introduce additional constraints and features that are required to tackle the TAARP.

4.2.1 The auction algorithm

In an assignment problem n objects and n agents are given, together with a value g_{ij} that represents the benefit of object j for agent i . The objective is to find the optimal assignment of objects to agents so that each agent is assigned to one object, each object is assigned to one agent and the total benefit is maximized.

In the auction algorithm, an object j has a *price* p_j that has to be paid by the agent to whom the object is assigned. Under these conditions, the value of object j for agent i is

equal to $g_{ij} - p_j$. Obviously, Each agent prefers the object which generates the maximum value for him/her, that is $\max_{\{j=1\dots n\}} \{g_{ij} - p_j\}$

The auction algorithm starts with an empty assignment and all prices equal to zero, and ends when all objects have been assigned. The algorithm proceeds in iterations, assigning a single agent to an object in each iteration. A typical iteration involves three steps:

- a) **Selection** Select an agent i to assign an object to. This agent can be selected at random, or in lexicographical order.
- b) **Bidding** Determine the object j_i which has maximum value for agent i , using the formula

$$j_i = \arg \max_{\{j=1\dots n\}} \{g_{ij} - p_j\} \quad (12)$$

The value of object j_i for agent i is given by $v_i = g_{ij_i} - p_{j_i}$, while the bid from agent i on object j_i is estimated to be

$$b_{ij_i} = p_{j_i} + v_i - w_i \quad (13)$$

where w_i is the second best value for agent i of all objects, i.e.

$$w_i = \max_{\{j=1\dots n, j \neq j_i\}} \{g_{ij} - p_j\} \quad (14)$$

- c) **Assignment:** Assign object j_i to agent i . If another agent was assigned to object j_i , this agent becomes unassigned again. Raise the price of object j_i to b_{ij_i} .

The algorithm continues with a sequence of iterations until all agents have been assigned to an object. Bertsekas [2] provides additional suggestions to modify the auction algorithm to deal with assignment problems. One of the suggestions is the ϵ -complementary slackness, which we will use below in the construction of a starting solution for the TAARP. Generally, auction algorithms perform well in practice, have excellent computational complexity properties and their running times are competitive, (Bertsekas [3]).

4.2.2 Auction heuristic for the TAARP

With some modifications, the auction algorithm described in the previous paragraphs can be adapted to find a feasible solution for the TAARP, where pupils have to be assigned to TAs. One of the reasons why a modification is required, is that although each pupil is assigned to a single TA, every TA can have several pupils assigned. We therefore define an *object* in the terminology of the auction algorithm to be *one hour of teaching of a given TA*. We define an *agent* to be *an hour of assistance required by a pupil*.

For each object t and each agent s , we define the benefit g_{st} to be a measure of the proximity between the teacher corresponding to the object and the pupil corresponding to the agent (i.e. a large enough constant minus the driving distance). If we assume that the total number of hours of teaching required by all pupils is equal to the total number of hours provided by the TAs, the corresponding problem of assigning objects (hours provided by a TA) to agents (hours demanded by a pupil) is an assignment problem that can be solved by the auction algorithm.

One problem with the auction algorithm is that its performance can be negatively affected by the fact that for each agent all objects associated with a particular TA generate the same benefit. The extension of the algorithm to the transportation problem considering *similar objects* proposed by Bertsekas and Castanon [4] avoids this difficulty. Two objects j and j' are called similar if they can be matched with the same agents at equal values. For each object j , the set of all objects similar to j is called the *similarity class* of j and is denoted by $M_o(j)$.

There are two other issues that must be taken into account in the real instance: (i) a session lasts either one or two hours, and (ii) there are pupils that require more than one session per week and these sessions must be taught by the same teacher and take place on different days.

The first issue can be addressed by defining *similar agents* as Bertsekas and Castanon [4] propose in order to deal with transportation problems. As mentioned, an agent is defined for each hour of assistance that a pupil requires (e.g. for a pupil who requires two sessions of two hours per week, four agents must be defined). Two agents i and i' are called similar if they obtain the same benefit when an object j is assigned to them. The set of all agents similar to an agent i is called the *similarity class* of i and is denoted by $M_a(i)$. In the auction algorithm for the TAARP, all agents corresponding to a specific pupil are similar.

The second issue implies that the set of objects that an agent i can bid for (denoted by $A(i)$) should often be restricted to a subset of the complete set of unassigned objects. Moreover, the set of possible objects that two agents can bid for are related if the agents correspond to the same pupil. Consider, for example, a pupil s who requires two sessions of two hours per week. For this pupil, four agents are created, one for each hour of assistance. Assume that object j ($j \in A(i)$) has already been assigned to agent i , i.e. one of the hours of assistance for this pupil has already been assigned. Consider a second agent k associated with a different hour of assistance required by pupil s :

- If agent k is a different hour in the same session as i (e.g. i is the first teaching hour of the first session of the week and k is the second hour for the same session) then $A(k)$ consists of objects associated with the same teacher and the same day of the week as object j .
- If agent k is associated with a different session than i (e.g. i is the first hour of the first session and k is the first hour of the second session), then $A(k)$ consists of

all objects associated with the same teacher but a different day of the week than object j .

The set $A(i)$ therefore changes depending on the current assignment (partial state of the auction).

Based on the basic auction heuristic for the assignment problem in Section 4.2.1 and the considerations listed above, we have developed a modified auction heuristic for the TAARP. This auction heuristic works as follows.

1. **Initialization.** Unassign all objects i and all agents j , set all prices p_i to zero.
2. **Auctioning.** This phase iteratively considers three steps until a feasible assignment is found.

If all agents have been assigned go to the routing phase (phase 3), otherwise perform the following steps:

- a) **Selection.** Select an unassigned agent i .
- b) **Bidding.** Determine the object j in $A(i)$ which has maximum value for agent i

$$j_i = \arg \max_{j=1,\dots,n} \{g_{ij} - p_j\} \quad (15)$$

The value of object j_i for agent i is given by $v_i = g_{ij_i} - p_{j_i}$. Estimate the bid from agent i on object j_i :

$$b_{ij_i} = p_{j_i} + v_i - w_i + \epsilon \quad (16)$$

where ϵ is a small constant explained below and w_i is the best value $g_{ij} - p_j$ for any object in $A(i)$ that belongs to a different similarity class than j_i :

$$w_i = \max_{\{j=1,\dots,n; j \notin M_o(j_i)\}} \{g_{ij} - p_j\} \quad (17)$$

If such an object does not exist, then w_i is set to $-v_i$.

- c) **Assignment.**
 - If object j_i was assigned to an agent $l \neq i$, then all objects in the similarity class $M_a(l)$ are unassigned.
 - Assign object j_i to agent i .
 - For every agent $k \in M_a(i)$, raise the price p_k to b_{ij_i} .
- d) **Perturbation.** If the number of iterations without decreasing the number of unassigned agents is equal to L , then decrease the price of all unassigned objects to zero. Otherwise, go to step 2a.

3. **Routing.** Apply a routing procedure to find the shortest tour for each teacher on each day.

In phase 2b, ϵ is a small constant value that ensures that prices keep increasing through the different iterations. This is important because it prevents the auction from getting stuck in a cycle of bids without price increments.

If the number of unassigned agents after the assignment step is not decreasing over time any more, the perturbation step tries to force the unassigned agents to bid for a free object when the algorithm has spent L iterations on a *war of prices*. To this end, the algorithm decreases the price of unassigned objects to make them more *attractive* for unassigned agents.

If at the end of the auction phase, a feasible assignment has been reached, it is still necessary to determine the order in which the pupils assigned to a teacher are visited on each day. This is done by an exhaustive enumeration procedure because the maximum number of pupils visited each day is at most eight.

4.3 Improving phase

The improving phase is based on the *variable neighborhood search* metaheuristic. More in particular, it corresponds to the *basic* VNS variant described in Hansen and Mladenovic [15]. A schematic overview of the improving stage is given in Algorithm 1.

Algorithm 1 Improving phase for the TAARP

Initialization: Consider an initial solution x , and select the set of neighborhood structures $N_k(x)$ $k = 1, 2, \dots, k_{\max}$.

repeat

Set $k \leftarrow 1$

repeat

Explore Neighborhood k . Find the best neighbor x' of x ($x' \in N_k(x)$)

Move or not. If the solution x' thus obtained is better than x , set $x \leftarrow x'$ and $k \leftarrow 1$; otherwise, set $k \leftarrow k + 1$;

until $k > k_{\max}$

until no improvement can be found any more

Two move types have been considered, defining two different neighborhood structures ($k_{\max} = 2$). The first move, that defines neighborhood $N_1(x)$, is a *simple exchange* move that attempts to swap pupils with similar characteristics. This move exchanges the positions of two pupils who need the same number of sessions and hours per sessions if no constraints are violated with the move, (e.g. if one of the TAs is not allowed to assist a pupil involved in the swap).

The second type of move, a *combined exchange*, allows exchanges among three pupils with different characteristics. Three cases of this type of move are considered.

- *Case 1*: A pupil who requires one session of two hours per week is exchanged with two pupils that need one session, of one hour per week and that are currently visited on the same day by the same teacher.
- *Case 2*: A pupil who requires two sessions of one hour per week is exchanged with two pupils of one hour per week session who are currently visited on different days by the same teacher.
- *Case 3*: A pupil who requires two sessions of two hours per week is exchanged with two pupils that need one session, of two hours per week and are currently visited on different days by the same teacher.

Figure 1 visualizes the first case for the second move type. Figure 2 corresponds to the cases 2 and 3, as they only differ on the numbers of hours per session required by the pupils involved in the move. In the figures, the squares correspond to the TAs and the bullets represent pupils. In Figure 1, one pupil who needs one session of two hours per week is swapped with two pupils who need one hour of assistance per week only. In Figure 2, a similar swap is performed except that the two pupils who need one hour of assistance are visited on different days.

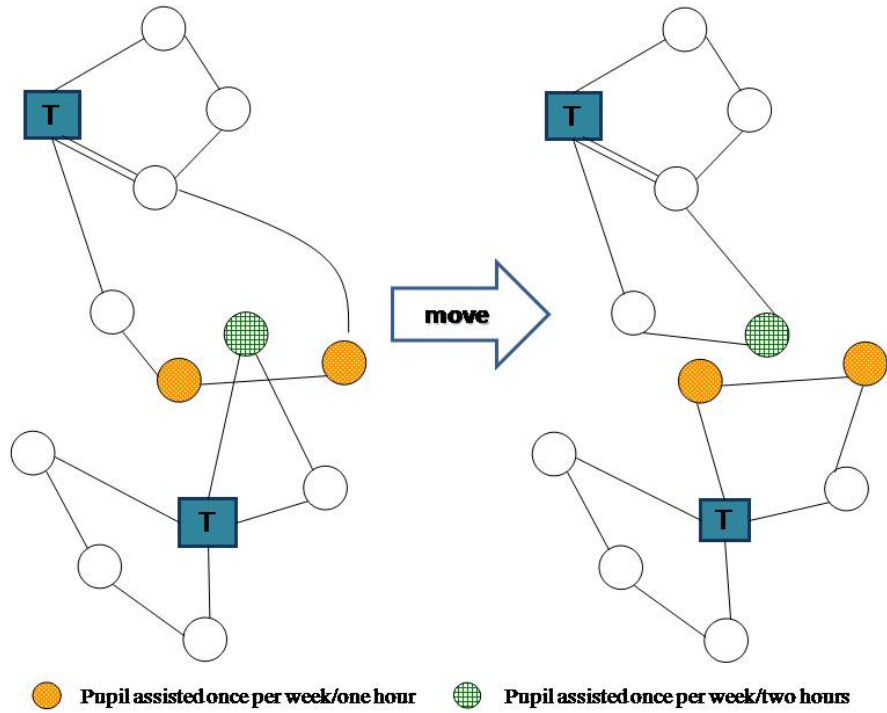


Figure 1: Example of the second type of move: Case 1

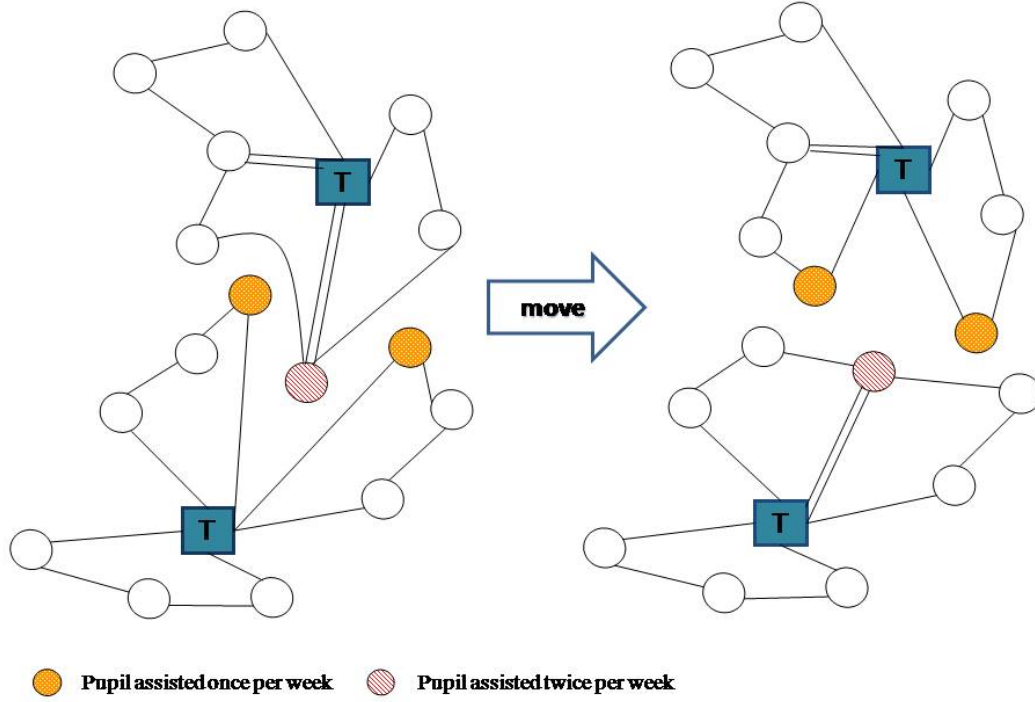


Figure 2: Example of the second type of move: Case 2 and 3

The neighborhood $N_2(x)$ is composed of all solutions x' that can be reached from the current solution x through the moves described in cases 1-3 without violating any constraint. After applying this type of move it is necessary to re-optimize the day-routes obtained because one of the sessions could have been assigned in a suboptimal position into the tour. As in the algorithm for constructing the initial solution, we use a complete enumeration for that purpose.

4.4 Algorithm structure

The solution approach has been described for one iteration, however it is possible to insert it in a multi iterate scheme by considering a different initial solution each time as in the multistart methods described by Martí [17]. In our approach, the auction algorithm is used to generate different starting solutions for the improving phase by randomly selecting agents in step 2a. The complete algorithm is run iteratively until a stopping criterion is reached (e.g. for the number of iterations or running time) and the best solution is kept. A schematic overview of the proposed solution strategy is

presented in Algorithm 2.

Algorithm 2 Solution strategy for the TAARP

Initialization: Consider an initial empty solution
while Stopping condition is not satisfied **do**
 Auction Algorithm: Get initial solution $\rightarrow \mathbf{x}$
 VNS Algorithm: Improvement phase $\rightarrow \mathbf{x}'$
 if \mathbf{x}' improves the incumbent **then**
 Update incumbent
 end if
end while
Return best solution

5 Computational results

In this section, we report the results from a computational study to evaluate the performance of our metaheuristic approach. For the study several small instances were generated randomly so that our approach could be compared to an exact solution method, i.e. the implementation of the mathematical model described in Section 3.

Next, we use our algorithm to solve the real-life instance of the TAARP that motivated this research. The mathematical model was implemented using Xpress-MP while the metaheuristic was coded in Java. All experiments were carried out on a personal computer equipped with an Intel(R)Core(TM)2 Duo(CPU) T9300 processor at 2.5 GHz and with 4 GB of RAM.

5.1 Randomly generated instances

In a first set of experiments, we concentrate on determining the effectiveness and efficiency of the metaheuristic proposed in this paper. Since only small instances can be solved to optimality using the mathematical model, we have generated several small random instances considering 20 pupils, to be assigned to three TAs and three working days. The coordinates of the pupils and TAs were generated using a $U[0, 100]$ distribution and rounded to the nearest integer. For each pupil the number of sessions per week was 1 with probability 0.7 and 2 with probability 0.3, while the number of hours per session is 1 or 2 with probability 0.25 and 0.75, respectively. These values were selected so as they emulate the distribution observed in the real data set. Table 2 shows the results obtained by both the exact method and the metaheuristic for ten of these instances. In the table, the column with heading TTD (total traveled distance) contains the objective function value, the columns with heading MTD contains the maximum traveled distance for any TA, and the columns with heading CPU contains the running time in seconds.

Table 2: Computational results for small simulated instances

Instance	Exact Method			Metaheuristic			
	TTD	MTD	CPU (s)	TTD	MTD	CPU (s)	Gap (%)
1	1166.7	273.3	13101.5	1166.7	273.3	12.0	0.00
2	1082.7	200.6	2095.9	1082.7	200.6	9.4	0.00
3	1025.6	193.4	6079.9	1057.8	225.7	6.4	3.14
4	1297.2	286.5	12246.4	1323.1	207.4	44.5	2.00
5	992.3	263.0	4204.5	1021.8	263.0	3.7	2.97
6	1000.1	174.1	26833.0	1000.1	174.1	5.9	0.00
7	1053.6	207.1	2586.8	1053.6	207.1	14.9	0.00
8	1161.3	210.3	530.0	1161.3	210.3	13.8	0.00
9	1273.4	175.3	11186.5	1273.4	175.3	15.2	0.00
10	1205.0	169.2	452.5	1205.0	169.2	12.8	0.00
Avg.			7931.7			13.9	0.81

The metaheuristic provides the optimal solution for seven out of ten instances. The maximum gap is 3.14%. However, the computing times for the metaheuristic are significantly smaller, even considering that 1000 iterations were performed for all instances and that the optimal solution was usually found long before the end of these iterations.

In the second set of experiments, we concentrate on analyzing the performance of three different elements of our metaheuristic: the heuristic to construct an initial solution, the variable neighborhood search and the effect of randomness. To this end, ten instances with 150 pupils, 16 TAs and 5 working days were generated. For each instance the following experiments were conducted:

- *Observation 0*: An initial solution is constructed using the heuristic described in Section 4.1 without randomization. This means that agents are assigned in the order in which they appear in the input file.
- *Observation 1*: The VNS is applied to the solution produced in Observation 0.
- *Observation i* : i iterations ($i > 1$) of the complete metaheuristic (including randomness in the construction of the initial solution) are carried out and the best solution is reported.

The results of these experiments are presented in Figure 3 where the improvement of each observation (in percent) is computed with respect to the objective value obtained in *observation 0*. The dashed lines are the result for each of the ten problem instances. The solid line is the average result.

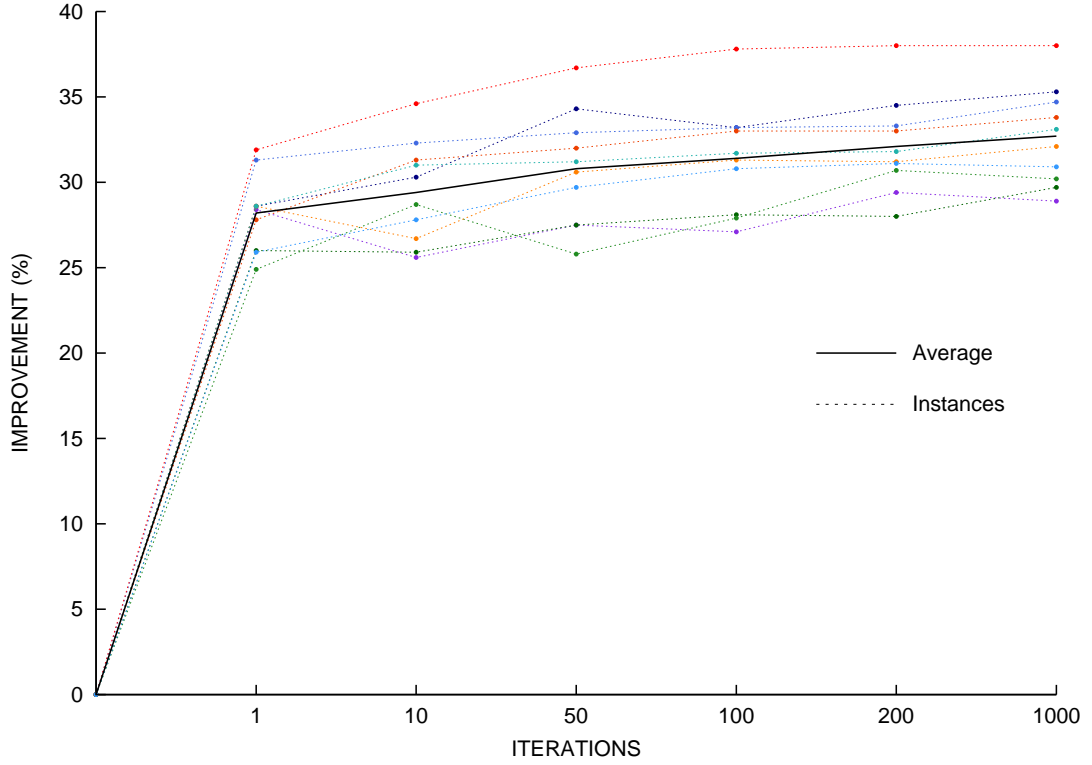


Figure 3: Results of the experiments with 10 artificially generated instances involving 150 pupils, 16 TAs and 5 working days

As can be seen from Figure 3, the heuristic usually provides an initial solution that is quite far from any local optimum, as the VNS generates about 28% improvement over that solution. This also demonstrates that the search in different neighborhoods is working properly. The use of randomness in the construction of the initial solution provides, on average, about 5% of additional improvement.

5.2 Real-life instance

The real-life instance involves 212 disabled children that must be assisted by 24 TAs. The pupils are distributed over 138 schools dispersed in a central area of Belgium. About 25% of the pupils require two visits per week and two of the pupils require three visits per week. For each TA, the number of working days is between one and five and the number of available hours per day is between two and eight.

We have used the *Google Maps API* to develop several scripts that allow us: (1) to geocode the addresses of schools and TAs, (2) to compute an approximation of the real driving distance between each pair of locations, and (3) to draw the routes generated

by our algorithm. Figure 4, showing the geographical distributions of schools and TAs, was generated using one of those scripts.

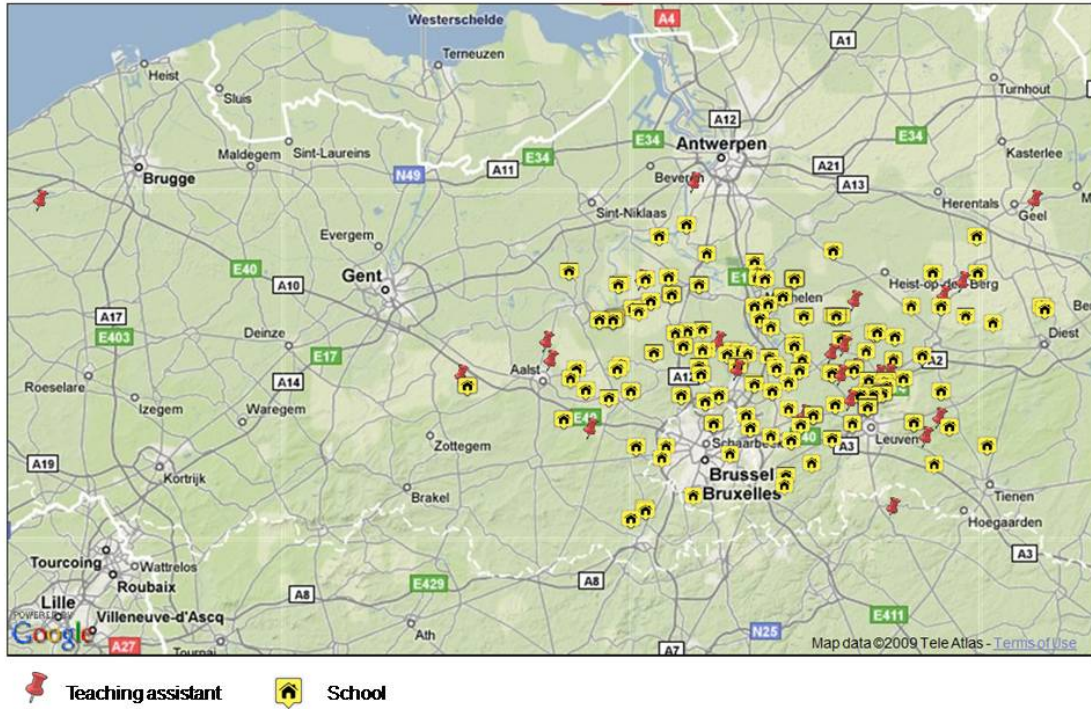


Figure 4: Geographical distribution of schools and TA

Currently, the planning is done by hand using only a spreadsheet. The total distance of the schedule currently in use is 4828.7km. Because of the planning difficulties this poses, the current schedule does not take into account the constraint that enforces the different sessions for a student to take place on non-consecutive days.

To compare our metaheuristic to the current planning technique, we have run it without considering this constraint, using 1000 iterations as a stopping criterion. The best solution was found in iteration 748 and the complete run took 4.6 hours of processing time. Table 3 presents for each TA the total number of pupils assigned, the traveled distance in kilometers per week (TTD) and the number of kilometers traveled per teaching hour (TTD/hour) for both the current solution and for the metaheuristic (MH). The change in TTD compared to the current solution and the number of common students between the two solutions (Comm.) are also showed.

The final solution produced by our algorithm improves the current solution by about 22%, and results in a schedule with 3756.1km of total traveled distance. On average, the total traveled distance per TA is around 45 km less than in the current solution, the number of kilometers traveled per teaching hour is also improved in about 2km and the

Table 3: Comparison of the current and the metaheuristic solution

TA	N° Stud			TTD		TTD/hour		
	Curr.	MH	Comm.	Curr. (km)	MH (km)	Curr.	MH	Differ. (%)
1	6	5	0	72.7	40.2	7.3	4.0	44.7
2	9	13	0	142.8	143.2	5.9	6.0	-0.3
3	7	10	1	75.1	58.8	3.6	2.8	21.7
4	9	11	2	135.2	191.4	5.9	8.3	-41.6
5	7	6	2	103.2	26.0	7.4	1.9	74.8
6	9	8	0	208.7	218.7	10.4	10.9	-4.8
7	7	8	0	248.4	240.8	15.5	15.1	3.1
8	6	5	2	87.0	122.8	7.9	11.2	-41.1
9	4	4	4	4.0	4.0	1.0	1.0	0.0
10	11	11	2	326.4	81.1	12.6	3.1	75.2
11	6	4	0	76.2	45.4	6.3	3.8	40.4
12	14	14	7	270.8	252.9	9.7	9.0	6.6
13	10	13	1	743.1	328.6	23.2	10.3	55.8
14	10	10	5	249.2	160.7	11.9	7.7	35.5
15	7	7	1	71.5	79.6	4.5	5.0	-11.3
16	16	16	13	155.8	147.5	5.6	5.3	5.3
17	9	9	3	186.0	110.9	9.3	5.5	40.4
18	12	11	9	146.6	183.9	5.2	6.6	-25.5
19	6	4	0	28.9	89.7	2.1	6.4	-209.8
20	8	6	0	183.4	26.1	11.5	1.6	85.7
21	11	8	2	151.2	96.8	6.9	4.4	36.0
22	8	9	6	901.2	851.8	41.0	38.7	5.5
23	11	12	5	182.1	131.2	6.5	4.7	28.0
24	9	8	2	79.4	124.2	3.3	5.2	-56.4
Average				201.20	156.50	9.35	7.43	7.00
Total		212	67	4828.71	3756.05			22.21

improvement obtained per TA amounts to 7%.

In our best solution, only 31% of the pupils are assigned to the same TA as in the current solution. This shows that the improvement is not only due to the routing but also to a better assignment of pupils to TAs.

While the travel distance is reduced for most TAs in the final solution, TA19 faces a substantial increase in TTD. However, the total distance that she has to travel per week is still less than 90km, which is less than the average over all the TAs. For TA22, the distances are considerable larger than for the others TA because TA22's home lies at a large distance from the cluster of schools to be visited, as can be seen in figure 4.

The value of our solution of the TAARP at the Royal Institute Woluwe can easily be calculated. The institute receives an annual subsidy of 483.50 euro for 184 of the pupils it assists in the context of integrated education. The total annual budget, which forms the basis for compensating the TAs for the distances they drive with their private cars, is therefore 88964 euro. Our solution to the TAARP problem saves the TAs a total of 1072.6 km per week and thus about 32178 km per annum. The compensation per km driven amounts to 0.25 euro, so that our solution saves the institute 268.15 euro per week and 8044.5 euro per annum, which is 9% of the annual budget and 22% of the current annual expenses for transportation.

Finally, we have also run our algorithm considering the constraint that enforces the different sessions for a student to take place on non-consecutive days. The total distance of the resulting schedule is 3861.5km, which is about 20% better than the current solution and 1% worse than the solution obtained without considering that constraint.

6 Conclusions and future research

In this paper, we have defined the teaching-assistants assignment–routing problem (TAARP) to solve a real-life routing problem faced by several institutes for extraordinary education in Flanders. This real-life problem is closely related to the multi depot periodic vehicle routing problem, but has several additional constraints. We described and implemented a solution approach based on the auction algorithm and concepts from the GRASP and VNS metaheuristics. The approach we proposed was tested using both simulated instances and a real-life data set. On both data sets, our algorithm performed excellently. The solution obtained on the real-life data set improves the total distance traveled by about 22% compared to the current solution used by the institute. The use of our method represents a 9% saving on the available budget to assists the disabled children.

According to our results, the key to improve the current solution in the real-life instance does not lie in the routing itself but in the clustering or distribution of the pupils over the TAs. Indeed, in this case the routing problems usually involve not more than a few nodes and are highly determined for the way that the students are allocated to the different

working days of the TAs. The proposed approach provides an integrated framework for the different decisions involved in the TAARP, where the clustering, assignment and routing decisions are considered simultaneously.

There are two potential topics for future research that arise from this work. First, we could explicitly consider the fairness of the assignment and look for solutions that minimize the total traveled distance while keeping the distance traveled by each teacher as homogeneous as possible. Second, we could attempt to construct a portfolio of alternative low-cost solutions, which can then be ranked using secondary criteria based on human and social considerations that are hard to quantify.

References

- [1] E. Beltrami and L. Bodin. Networks and vehicle routing for municipal waste collection. *Networks*, 4:65–94, 1974.
- [2] D. Bertsekas. Auction algorithms for network flow problems: A tutorial introduction. *Computational Optimization and Applications*, 1:7–66, 1992.
- [3] D. Bertsekas. *Network optimization. Continuous and discrete models*. Athena Scientific, 1998.
- [4] D. Bertsekas and D. Castanon. The auction algorithm for the transportation problem. *Annals of Operations Research*, 20:67–96, 1989.
- [5] A. Campbell and M. Savelsbergh. Efficient insertion heuristics for vehicle routing and scheduling problems. *Transportation Science*, 38:369–378, 2004.
- [6] A. Campbell, D. Vandenbussche, and W. Hermann. Routing for relief effort. *Transportation Science*, 42:127–145, 2008.
- [7] I. Chao, B. Golden, and E. Wasil. A new heuristic for the multi-depot vehicle routing problem that improves upon best-known solutions. *American Journal of Mathematical and Management Sciences*, 13:371–406, 1993.
- [8] N. Christofides and J. E. Beasley. The period routing problem. *Networks*, 14: 237–256., 1984.
- [9] J. Cordeau, M. Gendreau, and G. Laporte. A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks*, 30:105–119, 1997.
- [10] L. Drummond, L. Ochi, and D. Vianna. An asynchronous parallel metaheuristic for the period vehicle routing problem. *Future Generation Computer Systems*, 17: 379–386, 2001.
- [11] P. Francis, K. Smilowitz, and M. Tzur. The period vehicle routing problem with service choice. *Transportation Science*, 40:439–454, 2006.
- [12] P. Francis, K. Smilowitz, and M. Tzur. *The Vehicle Routing Problem: Latest Advances and New Challenges*, chapter The Period Vehicle Routing Problem and its Extensions, pages 73–102. Springer US, 2008.
- [13] I. Giosa, I. Tansini, and I. Viera. New assignment algorithms for the multi-depot vehicle routing problem. *Journal of the Operational Research Society*, 53:977–984, 2002.
- [14] E. Hadjiconstantinou and R. Baldacci. A multi-depot period vehicle routing problem arising in the utilities sector. *Journal of the Operational Research Society*, 49:1239–1248, 1998.

- [15] P. Hansen and N. Mladenovic. *Search Methodologies. Introductory Tutorials in Optimization and Decision Support Techniques*, chapter Variable Neighborhood Search, pages 211–238. Springer US, 2005.
- [16] G. Laporte, Y. Nobert, and D. Arpin. Optimal solutions to capacitated multi-depot vehicle routing problem. congress. *Congressus Numerantium*, 44:283–292, 1984.
- [17] R. Martí. *Handbook on MetaHeuristics*, chapter Multistart Methods, pages 355–368. Springer, 2003.
- [18] A. Mingozzi. The multi-depot periodic vehicle routing problem. In *Abstraction, Reformulation and Approximation. 6th International Symposium, SARA 2005.*, 2005.
- [19] M. Mourgaya and F. Vanderbeck. Periodic routing problem: classification and heuristic for tactical planning. *RAIRO Operations Research*, 40:169–194, 2006.
- [20] J. Renaud, G. Laporte, and F. Boctor. A tabu search heuristic for the multi-depot vehicle routing problem. *Computers & Operations Research*, 23:229–235, 1996.
- [21] M. Resende and C. Ribeiro. *Handbook of Metaheuristics*, chapter Greedy randomized adaptive search procedures, pages 219–249. Kluwer Academic, 2003.
- [22] A. Russell and D. Gribbin. A multiphase approach to the period routing problem. *Networks*, 21:747–765., 1991.
- [23] R. Russell and W. Igo. An assignment routing problem. *Networks*, 9:1–17, 1979.
- [24] C. Tan and J. Beasley. A heuristic algorithm for the period vehicle routing problem. *Omega*, 12:497–504, 1984.