

A DETERMINISTIC ANNEALING ALGORITHM FOR THE PRE- AND END-HAULAGE OF INTERMODAL CONTAINER TERMINALS

An Caris

Gerrit K. Janssens

Transportation Research Institute

Hasselt University - campus Diepenbeek

Wetenschapspark 5 - bus 6, 3590 Diepenbeek, Belgium

e-mail: {an.caris,gerrit.janssens}@uhasselt.be

KEYWORDS

pickup and delivery, intermodal transportation, meta-heuristics, deterministic annealing

ABSTRACT

The drayage of containers in the service area of an intermodal barge terminal is modelled as a full truckload pickup and delivery problem with time windows (FTPDPPTW). Initial solutions are generated with an insertion heuristic and improved with three local search operators. In a post-optimization phase the three search operators are integrated in a deterministic annealing (DA) framework. The mechanism of the heuristic procedures is demonstrated with a numerical example. A sensitivity analysis indicates that the DA algorithm is robust with respect to variations in threshold value and quality of the initial solution.

INTRODUCTION

Intermodal transport has grown into a dynamic transportation research field. Many new intermodal research projects have emerged. Intermodal transport integrates at least two modes of transport in a single transport chain, without a change of container for the goods, with most of the route traveled by rail, inland waterway or ocean-going vessel and with the shortest possible initial and final journeys by road (Macharis and Bontekoning (2004)). An overview of planning issues in intermodal transport and solution methods proposed in scientific literature is given by Caris et al. (2008). Intermodal planning problems are more complex due to the inclusion of multiple transport modes, multiple decision makers and multiple types of load units. Two strategic planning problems, terminal design and infrastructure network configuration, have received an increased attention in recent years. Yet the number of scientific publications on other intermodal planning problems, especially at the operational decision level, remains limited or non-existent. This paper discusses an operational planning problem in intermodal transport. Pre- and end-haulage of intermodal container terminals

involves the pickup or delivery of containers at customer locations. Road transport constitutes a relatively large share of intermodal transport costs. The attractiveness of intermodal transport can be increased by organizing the road segment in the intermodal transport chain more efficiently.

The drayage of containers in the service area of an intermodal terminal may be modelled as a Full Truckload Pickup and Delivery Problem with Time Windows (FTPDPPTW). Savelsbergh and Sol (1995) review the general Pickup and Delivery Problem (PDP). The PDP is an extension to the classical Vehicle Routing Problem (VRP) where customers may both receive and send goods. A fleet of vehicles is required to pickup and/or deliver goods at customer locations. A delivery activity to a consignee starts from the intermodal terminal with a full container and a pickup activity returns a container to the intermodal terminal for shipment by barge. In the Full Truckload Pickup and Delivery Problem (FTPDPPTW) a vehicle carries a single load. In the operational planning problem under investigation, a full truckload is assumed to be a single container. A recent overview of state-of-the-art research on pickup and delivery problems between customers and a depot is presented by Parragh et al. (2008). Only less-than-truckload problems are covered by the authors. Gronalt et al. (2003) study the problem of transporting full truckloads between distribution centres. In their Pickup and Delivery Problem with Time Windows (PDPTW) goods are transported between customer locations, as opposed to our problem definition where all containers either originate or return to the terminal. A full truckload PDPTW is also considered by Currie and Salhi (2003) and Currie and Salhi (2004). The problem studied in these papers also differs from our setting with respect to the definition of customer requests. Goods have to be picked up at works of a construction company and delivered to customers. Wang and Regan (2002) propose a hybrid approach to solve a PDP containing one or more intermodal facilities. Only pickup time windows are considered and the number of vehicles is fixed. The authors apply time window discretization in combination with a

branch and bound method. The closest related article to our research is written by Imai et al. (2007). The authors present a heuristic based on Lagrangian relaxation for the drayage problem of intermodal container terminals, without taking customer time windows into account.

The remainder of this paper is organized as follows. First, the problem formulation is given and a lower bound is proposed. Next, a multi-start local search heuristic is presented to generate an initial solution. This solution is further optimized by a deterministic annealing algorithm. A numerical example demonstrates both heuristic methods. Finally, conclusions are drawn and directions for future research are given.

PROBLEM FORMULATION

The FTPDPTW can be formulated in terms of a Vehicle Routing Problem with full container load. Assuming a homogeneous container type and size, the problem is to find an assignment of delivery and pickup customers to a fleet of vehicles, in order to minimize the total cost of serving all customers, which includes fixed vehicle costs and travelling costs. In accordance with Dumas et al. (1991), a fixed vehicle cost is introduced to minimize the fleet size. Each vehicle used incurs a fixed cost, which may vary with the vehicle. Fixed costs include depreciation of own vehicles or leasing costs if the vehicle is hired, insurance payments and fixed costs for hiring an extra truck driver. Travelling costs are proportional to the total time necessary to serve all customers, which implies travelling time and truck waiting time at customer sites. All orders are assumed to be known in advance, so the problem is studied in a static environment. An intermodal terminal is open during a pre-specified daily time window. All trucks k have to return to the terminal before the end of their depot window $(0, T_k)$. Hard time windows at customer locations are assumed.

The FTPDPTW is defined on a graph $G = (V_0, A)$, where V_0 represents the node set. V is the set of all customers, V^D is the set of delivery customers, V^P is the set of pick-up customers and $\{0\}$ is the singleton representing the depot.

$$\begin{aligned} V_0 &= V \cup \{0\} \\ V &= V^D \cup V^P \\ V^D \cap V^P &= \emptyset \end{aligned}$$

The set of arcs A consists of two types of connections. Arcs either connect the depot with a customer location or provide a connection between two customer locations. Feasible vehicle routes then correspond to paths starting at the depot 0, travelling through arcs connecting customer locations and returning to the depot 0. Only at the beginning and at the end of a route an arc is used to

connect a customer location with the depot. The logic of pickup and delivery customers is incorporated in the definition of travel times d_{ij} of arcs between customer locations. The travel time d_{ij} of arcs between two customer locations depends on the type of customers served. Four combinations of customers are possible: first a delivery then a pickup customer, two delivery customers consecutively, two pickup customers consecutively or first a pickup and then a delivery customer. Only when a pickup customer is served after a delivery customer a truck can drive directly from one customer location to the other. The travel time d_{ij} equals the time necessary to move directly from the delivery customer to the pickup customer t_{ij} .

$$d_{ij} = t_{ij}$$

In the other three customer combinations the truck first has to return to the depot before serving the second customer. In this case the travel time d_{ij} is set equal to the time necessary to travel from the first customer to the depot and then from the depot to the second customer.

$$d_{ij} = t_{i0} + t_{0j}$$

In this way, the problem can be modelled as a vehicle routing problem with time windows, as described by Cordeau et al. (2007). To formulate the problem the following notation is used:

- K = set of trucks
- $x_{ijk} = 1$ if customer i and customer j are served consecutively by truck k , else 0
- $y_k = 1$ if truck k is used, else 0
- C_{ijk} = travelling cost of arc (i, j) by truck k
- FC_k = fixed cost of truck k for a single route
- E_i = earliest start time of customer i
- L_i = latest start time of customer i
- b_i = actual time service at customer i begins
- s_i = service time of delivery i
- d_{ij} = travel time from customer i to customer j
- T_k = time capacity of truck k
- t_{0j} = travel time from terminal 0 to customer j
- t_{ij} = travel time directly from delivery i to pickup j
- t_{i0} = travel time from customer i to terminal 0

$$\text{Min} \sum_{i \in V_0} \sum_{\substack{j \in V_0 \\ i \neq j}} \sum_{k \in K} C_{ijk} x_{ijk} + \sum_{k \in K} FC_k y_k$$

subject to

$$\sum_{j \in V_0} \sum_{k \in K} x_{ijk} = 1 \quad \forall i \in V \quad (1)$$

$$x_{ijk} \leq y_k \quad \forall i, j \in V_0, \\ i \neq j, k \in K \quad (2)$$

$$\sum_{i \in V_0} x_{ijk} - \sum_{i \in V_0} x_{jik} = 0 \quad \forall j \in V, k \in K \quad (3)$$

$$E_i \leq b_i \leq L_i \quad \forall i \in V \quad (4)$$

$$\sum_{k \in K} x_{ijk} \cdot (b_i + s_i + d_{ij} - b_j) \leq 0 \quad \forall i, j \in V \quad (5)$$

$$\sum_{k \in K} x_{0jk} \cdot d_{0j} \leq b_j \quad \forall j \in V \quad (6)$$

$$x_{i0k} \cdot (b_i + s_i + d_{i0} - T_k) \leq 0 \quad \forall i \in V, k \in K \quad (7)$$

$$\sum_{j \in V} x_{0jk} \leq 1 \quad \forall k \in K \quad (8)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i, j \in (V_0), \\ i \neq j, k \in K \quad (9)$$

$$y_k \in \{0, 1\} \quad \forall k \in K \quad (10)$$

$$b_i \geq 0 \quad \forall i \in V \quad (11)$$

The objective function minimizes total costs of serving all customers. A fixed vehicle cost FC_k is incurred for each truck k used. The variable cost C_{ijk} represents the cost of serving customer j immediately after customer i , depending on the travel time and possibly waiting time in case a pickup customer is served directly after a delivery customer. Constraints (1) ensure that each customer is visited exactly once. Constraints (2) avoid to assign customers to unused vehicles. Flow conservation is enforced by constraints (3). Time windows at customer locations are stated in the fourth set of constraints (4). Expressions (5) and (6) enforce the consistency of time variables b_i . Hard time windows are also imposed on the total service time of a route k by constraints (7). Finally, constraints (8) guarantee that each vehicle is used at most once.

LOWER BOUND

The VRP belongs to the class of NP-hard problems. Since exact models are only able to solve relatively small problems, heuristics are used in practice to solve problems of realistic size. A lower bound is proposed to analyze the performance of the heuristics presented next. According to Cordeau et al. (2007) the LP relaxation of the VRPTW provides a weak lower bound. An alternative formulation is given in this section to be able to calculate a better lower bound for the optimal solution. In this formulation delivery customers are always indicated with index

i and pickup customers with index j . Each route k consists of a number of trips executed by a single truck k within the time window $(0, T_k)$. Let a trip be represented as a pair (i, j) where i represents a delivery customer and j a pickup customer. Pickup and delivery customers can be combined or can be served separately. In the case only a delivery customer belongs to a trip, the pair is written as $(i, 0)$. If only a pickup customer belongs to the trip, the pair is written as $(0, j)$. In the latter two cases either the delivery point or the pickup point is represented by the depot 0. This leads to the following alternative notation. All other symbols used, maintain the same interpretation as in the exact problem formulation (formulation (1)-(11)).

V_0^D = set of delivery points including the depot 0
 V_0^P = set of pickup points including the depot 0
 $x_{ijk} = 1$ if delivery i and pickup j are served in one trip by truck k , else 0
 CR_{ijk} = cost of serving pair (i, j) by truck k
 RS_{ij} = time necessary to serve pair (i, j)
 E_i = earliest start time of delivery i
 L_i = latest start time of delivery i
 E_j = earliest start time of pickup j
 L_j = latest start time of pickup j
 b_i = actual time delivery i begins
 b_j = actual time pickup j begins
 t_{0i} = travel time from terminal 0 to delivery i
 t_{ij} = travel time from delivery i to pickup j
 t_{j0} = travel time from pickup j to terminal 0
 s_i = service time of delivery i
 s_j = service time of pickup j

$$\text{Min} \sum_{i \in V_0^D} \sum_{\substack{j \in V_0^P \\ i+j \neq 0}} \sum_{k \in K} CR_{ijk} x_{ijk} + \sum_{k \in K} FC_k y_k$$

subject to

$$\sum_{i \in V_0^D} \sum_{k \in K} x_{ijk} = 1 \quad \forall j \in V^P \quad (12)$$

$$\sum_{j \in V_0^P} \sum_{k \in K} x_{ijk} = 1 \quad \forall i \in V^D \quad (13)$$

$$x_{ijk} \leq y_k \quad \forall i \in V_0^D, j \in V_0^P, \\ i + j \neq 0, k \in K \quad (14)$$

$$E_i \leq b_i \leq L_i \quad \forall i \in V^D \quad (15)$$

$$E_j \leq b_j \leq L_j \quad \forall j \in V^P \quad (16)$$

$$\sum_{k \in K} x_{ijk} \cdot (b_i + s_i + t_{ij} - b_j) \leq 0 \quad \forall i \in V^D, \\ j \in V^P \quad (17)$$

$$\sum_{i \in V_0^D} \sum_{j \in V_0^P} RS_{ij} \cdot x_{ijk} \leq T_k \quad \forall k \in K \quad (18)$$

$$x_{ijk} \in \{0, 1\} \quad \forall i \in V_0^D, j \in V_0^P, k \in K \quad (19)$$

$$y_k \in \{0, 1\} \quad \forall k \in K \quad (20)$$

$$b_i, b_j \geq 0 \quad \forall i \in V^D, j \in V^P \quad (21)$$

In the objective function the variable cost CR_{ijk} represents the cost of performing the complete trip (i, j) by truck k , including the costs incurred by truck k to leave and return to the depot. Equations (12) and (13) guarantee that all pickups and deliveries are visited only once. Constraints (14), (15), (16) and (17) are similar to constraints (2), (4) and (5) in the exact formulation. Time windows for the availability of trucks are expressed by constraints (18). The time necessary to perform trip (i, j) is given by the expression:

$$RS_{ij} = t_{0i} + t_{ij} + t_{j0} + s_i + s_j + MINWAIT_{ij}.$$

The minimum waiting time between delivery customer i and pickup customer j equals:

$$MINWAIT_{ij} = \begin{cases} 0 & \text{if } E_j \leq L_i + s_i + t_{ij} \\ E_j - (L_i + s_i + t_{ij}) & \text{else.} \end{cases}$$

In this formulation the feasibility of the routes is relaxed. If two trips share the same resource (the same vehicle), it is not ensured that the time intervals over which both trips require the resource do not overlap in time. Consequently the lower bound represents the variable costs of optimally combining delivery customers with pickup customers, but underestimates the number of vehicles necessary to perform the selected trips. The lower bound formulation leads to fewer constraints and variables and thus converges more quickly to an integer solution.

MULTI-START LOCAL SEARCH HEURISTIC

In the pre- and end-haulage of intermodal containers substantial cost and time savings may be realized by merging pickup and delivery customers in a single trip, as presented in figure 1. A heuristic procedure based on merging pickup and delivery customers is used to construct initial solutions. The insertion heuristic is briefly presented here to better understand the deterministic annealing algorithm. A detailed description and numerical example can be found in Caris and Janssens (2007). Three local search neighbourhoods are defined to improve initial solutions.

Insertion heuristic

In this section a two-phase insertion heuristic is described to create initial solutions. In a first phase, pickup and delivery customers are combined into pairs of customers. Due to the existence of hard time windows, not every pickup customer and delivery customer can be combined into a feasible pair. A limit is also imposed on the waiting time between delivery i and pickup j . This

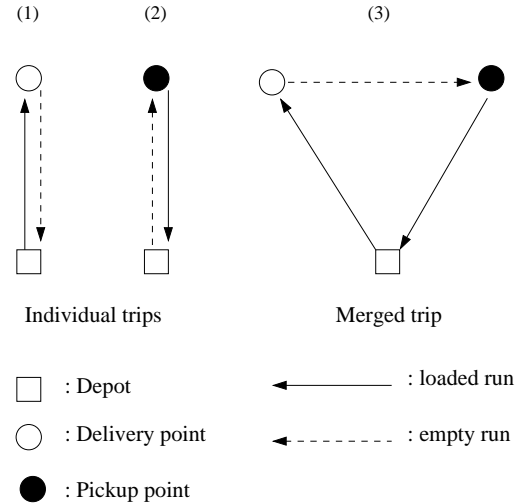


Figure 1: Merging trips

eliminates pairs of customers that are too far away from each other in time. A very large waiting time between the delivery location and pickup location will typically be cost inefficient in road haulage. In forming pairs of pickups and deliveries, both spatial and temporal aspects are to be taken into account. The pairs of pickup and delivery customers are ranked according to four criteria. The time window slack between customers i and j should be as small as possible (criterion 1). Savings in travel time obtained from serving delivery i and pickup j together should be as large as possible (criterion 2). An opportunity cost for not choosing the best combination for a delivery i or pickup j can also be taken into account. Gronalt et al. (2003) argue that this regret approach leads to significant improvements in the best solution. The opportunity cost $OC1_i$ (respectively $OC1_j$) can be defined as the difference in savings in travel time achieved by the best pair for delivery i (pickup j) and the currently selected pair (criterion 3). Finally, the opportunity cost related to the time window slack is incorporated in the selection criterion. This opportunity cost $OC2_i$ (respectively $OC2_j$) is defined as the difference between the time window slack of the current combination and the smallest time window slack of delivery i (pickup j) in any combination (criterion 4). These four criteria are aggregated by making use of weights. The pair of pickup and delivery customers with the lowest value for the following criterion is selected first:

$$w_1 \cdot (L_j - E_i - s_i - t_{ij}) + w_2 \cdot (t_{ij} - t_{i0} - t_{0j}) + w_3 \cdot (OC1_i + OC1_j) + w_4 \cdot (OC2_i + OC2_j). \quad (22)$$

The weights w_1 , w_2 , w_3 and w_4 reflect the importance given to each criterion and serve as parameters of the insertion heuristic. The domain of the weights is not fixed. The ratio between the weights influences the importance of each criterion. Depending on the nature of the problem, more weight should be given to savings

in waiting time or savings in travel time. The weights in the insertion heuristic are only used to construct an initial solution, which will be further improved by the local search procedure described below. The process of pairing customers is repeated until no more feasible combinations exist with respect to the remaining pickup customers and delivery customers. The remaining customers are inserted into individual trips and form an imaginary pair with a dummy customer.

In a second phase routes are constructed sequentially. Vehicles are used in increasing order of their fixed costs FC_k . Pairs of customers are eligible to be inserted into routes in increasing order of their latest start time. A pair of customers can be inserted into an existing route k if it can start later than the time necessary to serve the customers already assigned to the vehicle k and if the vehicle is able to return to the terminal within its depot window. In case insertion into multiple existing routes is feasible, the pair of customers is added to the existing route with the smallest waiting time between the previous pair. If no insertions into existing routes are feasible, the pair of customers is assigned to an unused vehicle to create a new route.

Improvement heuristic

A local search procedure is applied to improve a feasible solution obtained by the insertion heuristic. Three neighborhoods are defined, as presented in figure 2. First, the CROSS operator recombines pairs of customers of different routes. This operator improves the result of the pairing phase in the insertion heuristic. A second operator, COMBINE, joins two routes into one. Finally, customers are removed from a route and inserted into another route by the INSERT operator. The latter two search neighbourhoods affect the result of the route construction phase of the insertion heuristic.

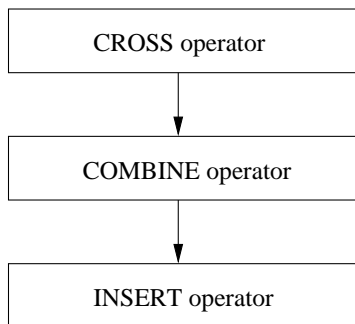


Figure 2: Improvement heuristic

The CROSS operator selects two pairs of pickup and delivery customers, for example (g, h) and (i, j) from two different routes. These pairs are recombined into new pairs of pickup and delivery customers, (g, j) and

(i, h) . First, all feasible CROSS moves are listed. A CROSS move is feasible if the pickup customers and delivery customers can be combined into new pairs, taking into account their time windows. Second, it is checked whether the new pairs of customers can be reinserted into the routes. Either (g, j) is inserted into the first route and (i, h) into the second or the other way round. In the local search heuristic the CROSS move with the largest improvement is selected. If a resulting route only contains dummy customers, this route is removed from the solution and the number of trucks necessary is reduced by one. The COMBINE operator checks whether two routes served by different trucks can be combined into a single route. Whereas the first operator reduces the travelling costs in the objective function, the COMBINE operator is able to reduce the number of trucks. Two routes can be combined if the last pair of the first route can be served before the latest starting time of the second route. The third operator, INSERT, removes pairs of pickup and delivery customers from their routes and reinserts them into another route. The INSERT operator attempts to eliminate routes, by inserting their customers into other routes. Pairs of customers can be inserted in the beginning of a route, between pairs of customers or at the end of a route. Similar to the COMBINE operator, this operator also impacts the number of trucks used and consequently the fixed vehicle costs in the objective function. These neighbourhood mechanisms are subsets of the general λ -interchange mechanism, described by Osman and Wassan (2002). The CROSS operator is an example of a 1-interchange mechanism, which involves only a single customer of each route. Due to the CROSS operator, two routes may exchange either pickup customers or delivery customers of two pairs simultaneously. The INSERT operator represents a 2-consecutive-node interchange mechanism. Two consecutive customers which constitute a pair in a single route are shifted to another route. Similarly, the COMBINE operator may be seen a n -consecutive-node interchange mechanism.

A multistart approach using different values for the weights in selection criterion (22) of the insertion heuristic may be applied to obtain the best overall solution.

DETERMINISTIC ANNEALING

A deterministic annealing algorithm is applied in a post-optimization phase to further improve on solutions found by the multistart local search heuristic. Deterministic annealing (DA), also referred to as 'threshold accepting', is introduced by Dueck and Scheuer (1990) as a deterministic variant of simulated annealing (SA). In each step of an SA algorithm a new solution S' is generated in the neighbourhood of the current solution S . If the new solution has a better objective value, it

is accepted automatically. If it is worse, it is accepted only with a certain probability. The probability of acceptance $e^{-\Delta/T}$ depends on the change in objective value $\Delta = C(S') - C(S)$ and a parameter T , called temperature. The temperature T is updated according to a certain annealing schedule. In the beginning of the search T is set at a level with a high probability of accepting worse solutions. Gradually, the probability of accepting deteriorations is lowered, until only improvements are accepted. A great variety of annealing schedules exist in literature. However, Dueck and Scheuer (1990) state that in most applications the success of SA is very sensitive against the choice of annealing schedule. Deterministic annealing offers a greater simplicity. The difference between SA and DA lies in the different acceptance rules. In DA a neighbouring solution with a worse objective value than the current solution is accepted if the deterioration $\Delta = C(S') - C(S)$ is less than a deterministic threshold value T .

Applications of DA to vehicle routing problems can be found amongst others in Tarantilis et al. (2004) and Bräysy et al. (2008). DA is applied to the problem formulation described in this paper based on the implementation strategy of Bräysy et al. (2008). The final solution of the multistart local search heuristic serves as initial solution for the DA algorithm, presented in Algorithm 1.

Algorithm 1 Deterministic annealing for FTPDPTW

```

Set best solution of multistart local search heuristic
as current best solution  $S_{best}$  of DA
Set  $T = T_{max}$  and  $i_{last} = 0$ 
for  $i=1$  to  $n_{improve}$  do
    Choose two random starting routes
    for All route pair combinations do
        Apply CROSS
        Apply COMBINE
        Apply INSERT
    end for
    if  $C(S') < C(S_{best})$  then
        Set  $S_{best} = S'$  and  $i_{last} = i$ 
    else
        if  $T \leq 0$  and  $i - i_{last} \geq \bar{n}$  then
            Restart from  $S_{best}$ :
            Set  $S' = S_{best}$ ,  $i_{last} = i$  and  $T = r \cdot T_{max}$ 
        else
            if  $T \leq 0$  then
                Set  $T = r \cdot T_{max}$ 
            else
                Set  $T = T - \Delta T$ 
            end if
        end if
    end if
end for

```

The three local search neighbourhoods CROSS, COMBINE and INSERT are integrated in a deterministic annealing framework. Routes are searched in a fixed order, but at the beginning of each iteration the starting point of the search is randomly chosen. Neighbouring solutions with a worse objective value are accepted when $\Delta = C(S') - C(S)$ is less than the threshold value T . For each pair of routes at most one move for each local search operator is accepted in each iteration. In the DA algorithm a first-accept strategy is applied, whereas in the local search heuristic in the previous section the best move was chosen. The threshold value is initially set at a maximum value T_{max} . In each iteration without any improvement in objective function value T is lowered with ΔT units. The threshold value is reset to $r \cdot T_{max}$ whenever it reaches zero, with r a random number between 0 and 1. When after a predefined number of iterations \bar{n} no improvements have been found and T reaches 0 again, the algorithm restarts from the current best solution S_{best} found. The process is repeated for $n_{improve}$ number of iterations.

NUMERICAL EXAMPLE

A numerical example is discussed to demonstrate the mechanism of the heuristic procedures. In this example an intermodal terminal has to pickup or deliver containers to a hundred customer sites. The terminal is open during eight hours per day. Service at customer sites takes eight minutes. The problem is studied in a deterministic environment. Travel times, waiting times and service times are therefore assumed to be constant. Customer locations are randomly selected with x - and y -coordinates between zero and 25. Time windows at customer locations are randomly chosen between 60 and 120 minutes. The terminal cooperates with a single haulier for performing the road segment of intermodal transport requests. Therefore, travelling costs and fixed vehicle costs are assumed equal for all vehicles. A fixed vehicle cost of 10 is charged per vehicle in use.

In the insertion heuristic a maximum waiting time between delivery customers and pickup customers of 30 minutes is allowed. A multistart approach is applied, varying the weights in selection criterion (22). The weights are altered from zero to 100 with increases of five units. The weights always sum up to 100. The best overall solution is obtained with the weights reported in table 1. A large weight is given to the opportunity costs of savings in travel time. No weight is allocated to the time window slack between customers or opportunity costs of time window slack.

Table 2 presents the variable cost (VC) or travelling cost, fixed vehicle cost (FC) and total cost (TC) of the best overall solution after applying the insertion heuristic and the three local search neighbourhoods. The insertion heuristic serves to provide an initial solution

\mathbf{w}_1	\mathbf{w}_2	\mathbf{w}_3	\mathbf{w}_4
0	20	80	0

Table 1: Weights best overall solution multistart local search heuristic

quickly. This initial solution is strongly improved by the three local search operators. The CROSS operator reduces the variable cost, whereas the two other operators are aimed to decrease the fixed vehicle cost. The final total cost differs only 1.94% from the lower bound. In the lower bound solution less vehicles are required, due to the relaxation of route feasibility with respect to the customer time windows.

	VC	FC	TC
Insertion heuristic	2903	90	2993
CROSS	2768	90	2858
COMBINE	2768	90	2858
INSERT	2768	70	2838
Lower bound	2734	50	2784

Table 2: Multistart local search heuristic

Deterministic annealing is applied as a post-optimizer to further reduce the total cost of this solution. In the DA algorithm the number of iterations $n_{improve}$ is fixed at 200. The algorithm is restarted from the current best solution S_{best} after 10 iterations without any improvements \bar{n} with the threshold value at zero. The maximum threshold value T_{max} equals two, with a change in threshold value ΔT of 0.025. Results of three independent runs of the DA algorithm are given in table 3. The DA algorithm finds further reductions in travelling costs. The three runs show similar results with a gap of around 1% between the heuristic solution and the lower bound.

	Run 1	Run 2	Run 3
VC	2737	2740	2748
FC	70	70	70
TC	2807	2810	2818
Gap	0.82%	0.93%	1.22%

Table 3: Deterministic annealing algorithm

Figure 3 presents a sensitivity analysis to determine the maximum threshold value T_{max} . The maximum threshold value is varied from zero to five with intervals of size 0.2. Three independent test runs are performed for each value. The vertical axis shows the percentage deviation of the median objective function value with respect to the lowest total cost over all threshold values and test runs. A maximum threshold value of at least 1.4 is appropriate to obtain the lowest

percentage deviation. Maximum threshold values less than one do not allow enough diversity in the search process. Only minor deviations are found over all threshold values, which indicates that the DA algorithm is robust for changes in maximum threshold value.

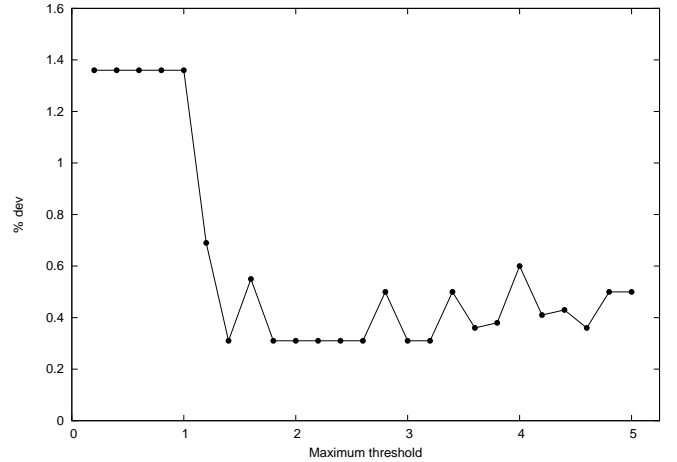


Figure 3: Sensitivity analysis of T_{max}

In figure 4 the influence of parameter ΔT on the solution quality is investigated in a similar way. The maximum threshold value is held constant at $T_{max} = 2$. Only small deviations from the lowest objective function value are measured, showing the robustness of the DA algorithm for changes in ΔT .

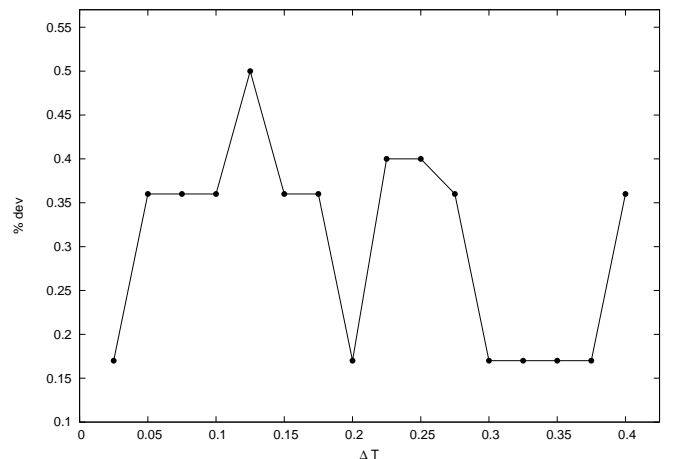


Figure 4: Sensitivity analysis of ΔT

In table 4 multiple initial solutions are tested for the deterministic annealing algorithm. Ten different initial solutions are generated by assigning the values in columns one to four to the weights in selection criterion (22). For each initial solution three independent test runs are performed. The fifth column (TC) mentions the median objective function value. The percentage deviation

from the minimum value is reported in column six (% dev). The total costs differ only slightly. A comparison of table 4 with table 3 shows that the lowest overall cost results from the best solution generated by the multi-start local search heuristic.

w_1	w_2	w_3	w_4	TC	% dev
1	0	4	5	2845	0.23
1	0	8	1	2838	0.00
3	2	1	4	2842	0.14
3	3	2	2	2839	0.05
2	6	3	0	2839	0.05
0	0	6	4	2850	0.42
1	4	4	1	2845	0.23
3	1	3	3	2838	0.00
5	0	1	4	2850	0.42
3	1	4	2	2849	0.38

Table 4: Sensitivity analysis of initial solution

CONCLUSIONS AND FUTURE WORK

In this paper a deterministic annealing algorithm is presented to find near optimal solutions for the drayage of containers in the service area of intermodal terminals. The DA algorithm is based on three local search operators, CROSS, COMBINE and INSERT. A preliminary analysis is performed with a numerical example. The DA algorithm generates good quality solutions independent of the quality of the initial solution. In the future computational experiments will be set up to confirm the robustness of the algorithm with respect to variations in problem characteristics.

REFERENCES

- O. Bräysy, W. Dullaert, G. Hasle, D. Mester, and M. Gendreau. An effective multirestart deterministic annealing metaheuristic for the fleet size and mix vehicle routing problem with time windows. *Transportation Science*, Published online, Articles in Advance, 2008.
- A. Caris and G. K. Janssens. Pre- and end-haulage of intermodal container terminals modelled as a full truckload pickup and delivery problem with time windows. In *Proceedings of the 21st European Conference on Modelling and Simulation*, pages 554–559, Prague, Czech Republic, 2007.
- A. Caris, C. Macharis, and G. K. Janssens. Planning problems in intermodal freight transport: Accomplishments and prospects. *Transportation Planning and Technology*, 31(3):277–302, 2008.
- J.-F. Cordeau, G. Laporte, M. W. Savelsbergh, and D. Vigo. Vehicle routing. In C. Barnhart and G. Laporte, editors, *Transportation*, volume 14 of *Handbooks in operations research and management science*, chapter 6, pages 367–428. North-Holland, 2007.
- R. Currie and S. Salhi. Exact and heuristic methods for a full-load, multi-terminal, vehicle scheduling problem with backhauling and time windows. *Journal of the Operational Research Society*, 54:390–400, 2003.
- R. Currie and S. Salhi. A tabu search heuristic for a full-load, multi-terminal, vehicle scheduling problem with backhauling and time windows. *Journal of Mathematical Modelling and Algorithms*, 3:225–243, 2004.
- G. Dueck and T. Scheuer. Threshold accepting: a general purpose optimization algorithm appearing superior to simulated annealing. *Journal of Computational Physics*, 90:161–175, 1990.
- Y. Dumas, J. Desrosiers, and F. Soumis. The pickup and delivery problem with time windows. *European Journal of Operational Research*, 54:7–22, 1991.
- M. Gronalt, R. F. Hartl, and M. Reimann. New savings based algorithms for time constrained pickup and delivery of full truckloads. *European Journal of Operational Research*, 151:520–535, 2003.
- A. Imai, E. Nishimura, and J. Current. A Lagrangian relaxation-based heuristic for the vehicle routing with full container load. *European Journal of Operational Research*, 176(1):87–105, 2007.
- C. Macharis and Y. M. Bontekoning. Opportunities for OR in intermodal freight transport research: A review. *European Journal of Operational Research*, 153:400–416, 2004.
- I. H. Osman and N. A. Wassan. A reactive tabu search meta-heuristic for the vehicle routing problem with back-hauls. *Journal of Scheduling*, 5:263–285, 2002.
- S. N. Parragh, K. F. Doerner, and R. F. Hartl. A survey on pickup and delivery problems, part i: Transportation between customers and depot. *Journal für Betriebswirtschaft*, 58:21–51, 2008.
- M. Savelsbergh and M. Sol. The general pickup and delivery problem. *Transportation Science*, 29(1):17–29, 1995.
- C. Tarantilis, C. Kiranoudis, and V. Vassiliadis. A threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem. *European Journal of Operational Research*, 152:148–158, 2004.
- X. Wang and A. C. Regan. Local truckload pickup and delivery with hard time window constraints. *Transportation Research Part B*, 36:97–112, 2002.