

Nordsieck, Niklas; Buer, Tobias; Schönberger, Jörn

Working Paper

A three-phase heuristic for a multi-size container transport problem with partially specified requests

Bremen Computational Logistics Group Working Papers, No. 5

Provided in Cooperation with:

Computational Logistics Junior Research Group, University of Bremen

Suggested Citation: Nordsieck, Niklas; Buer, Tobias; Schönberger, Jörn (2016) : A three-phase heuristic for a multi-size container transport problem with partially specified requests, Bremen Computational Logistics Group Working Papers, No. 5, University of Bremen, Computational Logistics Junior Research Group, Bremen

This Version is available at:

<https://hdl.handle.net/10419/145291>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

A Three-Phase Heuristic for a Multi-Size Container Transport Problem with Partially Specified Requests

Niklas Nordsieck, Tobias Buer and Jörn Schönberger

Bremen Computational Logistics Group Working Papers, No. 5

August 2016

Recommended citation:

Original paper: Nordsieck, N.; Buer, T. and Schönberger, J.: A Three-Phase Heuristic for a Multi-Size Container Transport Problem with Partially Specified Requests, In: Nissen, V.; Stelzer, D. Straßburger, S. and Fischer, D. (eds.): Multikonferenz Wirtschaftsinformatik (MKWI) 2016, Ilmenau 2016, pp. 217-228. URN: urn:nbn:de:gbv:ilm1-2016100012

This postprint: Nordsieck, N.; Buer, T.; Schönberger, J. (2016): A Three-Phase Heuristic for a Multi-Size Container Transport Problem with Partially Specified Requests. In: Bremen Computational Logistics Group Working Papers, No. 5.

Bremen Computational Logistics Group Working Papers provides open access to papers which are (co-)authored by members of the [Computational Logistics](#) group at [University of Bremen](#).

Editor:

Prof. Dr. Tobias Buer

Computational Logistics Junior Research Group

Faculty of Business Studies & Economics | FB7

University of Bremen

Bibliothekstr. 1, 28359 Bremen, Germany

<http://www.cl.uni-bremen.de>

A member of [LogDynamics](#)

This manuscript ...	
...was <i>not</i> peer-reviewed	
...has been peer-reviewed	✓
...is a postprint and has been accepted for publication (see recommended citation)	✓
...updates working paper No.	—

A Three-Phase Heuristic for a Multi-Size Container Transport Problem with Partially Specified Requests

Niklas Nordsieck¹, Tobias Buer^{1,*} and Jörn Schönberger²

¹Computational Logistics, University of Bremen
Bibliothekstr. 1, 28359 Bremen, Germany
{niklas.nordsieck, tobias.buer}@uni-bremen.de

²Chair of Transport Services and Logistics, TU Dresden
Würzburger Str. 35, 01187 Dresden, Germany
joern.schoenberger@tu-dresden.de

Abstract

The present paper studies a generalization of the less-than-truckload pickup and delivery problem. The problem at hand arises in the hinterland of container terminal where empty and loaded containers have to be transported between a container depot and a set of customer locations. However, requests including empty containers are only partially specified. That is, either the pickup location or the delivery location of a request including the transportation of an empty container is a priori known. Customers who demand an empty container do not care which specific empty container is provided, i.e., while the delivery location is given, the pickup location is part of the problem's solution. To solve this problem, an iterated three-phase heuristic is introduced. The first phase completes the partially specified requests, the second phase solves a standard pickup and delivery problem, the third phase changes parameters of the objective function and the process iterates. Computational results on a set of 1,000 test instances are presented.

1 Introduction

The maritime container traffic increased from the year 2000 to the year 2010 by more than 40%. Containerization of transport processes is considered a world-wide success story. Nevertheless, it also leads to new problems. Due to trade imbalances, supply and demand of empty containers may significantly differ in different geographical regions. On the global level, for example, Asia exports significantly more goods to Europe and North America than vice versa. Therefore, a surplus of empty containers exists in Europe and North America while a shortage of empty containers exists in Asia. To balance supply and demand, the problem of repositioning empty containers has to be solved. This repositioning problem does not only arise on the global level between continents, but also on the local level among shippers or consignees in the hinterland

*Corresponding author

of an intermodal container terminal like a seaport or a railway station. More importantly, those drayage operations are responsible for the bigger part of the transport costs of a global supply chain. Increasing the efficiency of drayage operations can therefore significantly contribute to reducing transportation related supply chain costs.

In this paper, we study the problem of truck-based repositioning of empty containers in the hinterland of a container terminal between shippers and consignees of containers. We denote the problem as the *container pickup and deliver problem* (CPDP). From a research perspective, the CPDP is interesting, because it generalizes the well-known less-than-truckload pickup and delivery problem with time windows (PDPTW). In extension of the PDPTW, some of the requests in the CPDP are only partially specified. That is, for some requests either a pickup location or a delivery location has to be determined as part of the problem's solution. Depending on what decision is made to complete these partially specified requests, the total number of pickup and delivery requests is variable. In order to solve the CPDP an iterated three-phase heuristic is presented which solves the CPDP in a sequential way.

The remaining paper is organized as follows. Section 2 describes the CPDP and reviews related literature. Section 3 presents an iterative three-phase heuristic to solve the CPDP. The performance of the heuristic is evaluated in Section 4. Section 5 concludes the paper.

2 Hinterland Transportation of Loaded and Empty 20-foot and 40-foot Containers

A generalization of the well-known less-than-truckload pickup and delivery problem with time windows is presented in Section 2.1. It is denoted as the container pickup and delivery problem. Related literature to the container pickup and delivery problem is discussed in Section 2.2.

2.1 The Container Pickup and Delivery Problem

The CPDP generalizes the well-known PDPTW. For an in-depth discussion of the PDPTW and its variants we refer to literature reviews like, e.g., Savelsbergh and Sol (1995).

We first introduce the PDPTW. We are given a complete graph. The nodes of this graph consists of $i = 1, \dots, n$ customer locations and a vehicle depot $i = 0$. The distance between two nodes i and j is given by d_{ij} . We assume, one distance unit is equal to one unit of driving time. For each customer node i and the vehicle depot 0, a time window $[e_i, l_i]$ is given by the earliest e_i and latest l_i starting time of a service. The service duration required for handling of goods at customer i is given by s_i . The time window $[0_s, 0_e]$ of the single depot 0 defines the planning horizon of the problem which usually covers one working shift.

Let R denote a set of requests. A request $r \in R$ includes a pickup location r^+ , a delivery location r^- , and a load of volume $c(r)$. The task of request r is to pickup the load $c(r)$ at node r^+ and transport it to node r^- . A finite set V of homogeneous vehicles is given. The maximum transport capacity of vehicle $v \in V$ is \bar{v} . The currently loaded capacity of v is denoted as v^c .

where $v^c \leq \bar{v}$ has to hold at every point in time. The arrival time of v at customer i is denoted as v_i^t . By assumption, the cardinality of V is large enough to fulfill all requests in R .

A feasible solution of the PDPTW requires that each request $r \in R$ is serviced by one vehicle. A vehicle $v \in V$ starts and ends its tour at the vehicle depot 0 and visits a set of customers. Vehicle v may leave the depot not earlier than 0_s and return no later than 0_e . A request $r \in R$ is fulfilled by only one vehicle and the pickup location r^+ has to be visited before the corresponding delivery location r^- . Each customer is visited only once, only during its service time window, and the vehicle capacity \bar{v} is never exceeded.

Different objective functions are discussed for the PDPTW. Popular is a lexicographic function where the number of vehicles required to fulfill all requests is minimized first and then the total operation time of all vehicles is minimized second. In this study, the goal is to find a feasible solution that minimizes the total operation time of all vehicles. The operation time of a vehicle v is calculated by the point in time v returns to the vehicle depot minus the point in time v left the vehicle depot.

As distinct from the PDPTW, the CPDP features the following additional characteristics. Only containers are considered as a means of transport. We focus on two container sizes, namely 20-foot and 40-foot containers. Containers may be *empty* or *loaded*. In addition to the vehicle depot we now also include a container depot.

In the CPDP, every request includes the transport of a container. There are three types of requests, which are described from the point of view of a customer (Schönberger et al., 2013):

- i) *Standard*: pickup loaded container at a given customer and deliver it to a given customer or the container depot.
- ii) *Store*: pickup an empty container at a given customer location.
- iii) *Provide*: deliver an empty container to a given customer.

Standard requests are traditional PDPTW requests. In Schönberger et al. (2013) they were differentiated further into export and import request, in case they include the container terminal (at a sea port) as a delivery or a pickup location, respectively. Store and provide requests are, however, only partially specified. That is, either the delivery or the pickup location of such a request is missing and has to be determined while solving the CPDP. In that sense, store and provide requests are *incomplete* requests prior to planning. For a store request, the empty container may be transported to the container depot or to any other customer that has issued a provide request (if the container sizes of both requests are equal). On the other hand, for a provide request, the pickup location of an empty container may be the container depot or any customer that has issued a store request (again, if the container sizes of both requests are equal). We assume, the container depot has a sufficient storage capacity and that it is always possible to fulfill storage and provide requests of empty containers by means of the container depot. An example of a feasible solution of the CPDP is shown in Figure 1. Store requests are issued by customers 2 and 4, provide requests are issued by customers 5 and 6.

Container size is measured in *twenty-foot equivalent unit* (TEU). Each request $r \in R$ requires a load of $c(r) = 1$ or $c(r) = 2$, where $c(r) = 1$ stands for a 20-foot container and $c(r) = 2$

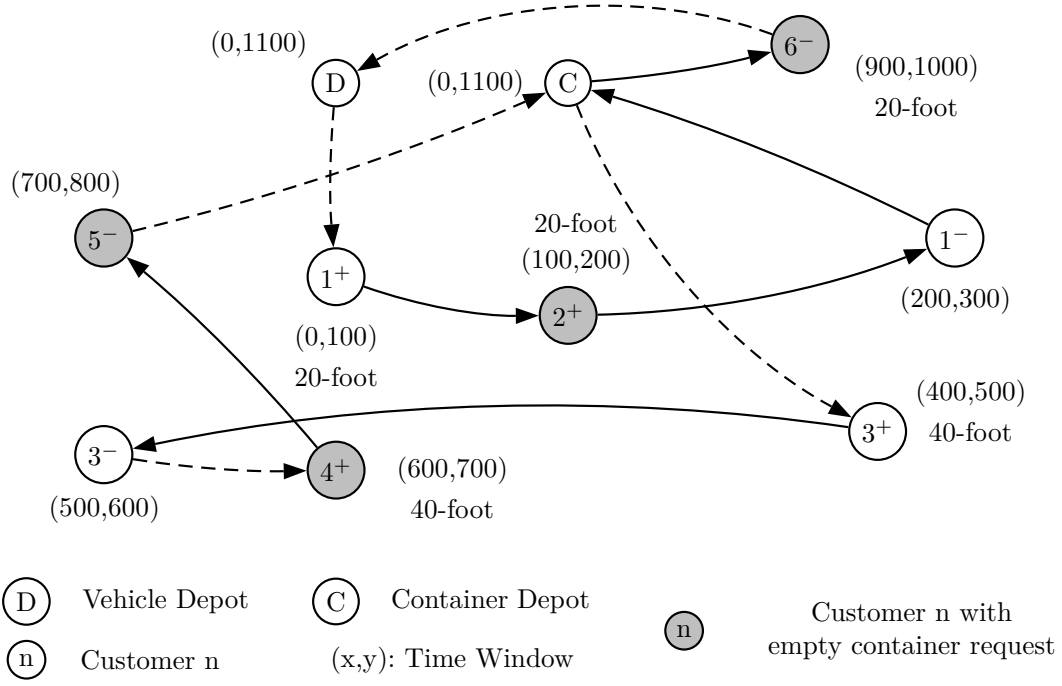


Figure 1: Example instance and solution of the CPDP

represents a 40-foot container. The maximum capacity of a vehicle is $\bar{v} = 2$ and thus can carry a maximum of one 40-foot container or two 20-foot containers simultaneously. Consequently, a full truckload request includes the transportation of a 40-foot container (either empty or loaded) and a less-than-truckload request includes the transportation of a 20-foot container (either empty or loaded).

2.2 Related Literature

The repositioning of empty containers is a problem that arises due to trade imbalances. It is often studied from a global point of view in the maritime transportation, where large trade imbalances between Asia and western countries exist at the moment. However, the problem is also relevant on the local level, where it is studied in the context of the hinterland of a container terminal (for example, a dedicated container depot, a terminal at a seaport or an intermodal railway terminal). In these scenarios, the literature on vehicle routing problems is most relevant to the CPDP.

Jula et al. (2005) were one of the first to investigate container drayage operations. It focuses on the transportation of containers between port terminals, intermodal facilities, and customers. Only full truckload requests are considered and all pickup and delivery requests are known a priori, i.e. incomplete requests are not studied. The objective function minimizes the total travel distance.

Sterzik and Kopfer (2013) formulate the inland container transportation problem. Empty and loaded 40-foot containers are considered as a means of transport, i.e. a full-truckload vehicle routing problem is studied. Incomplete requests are explicitly considered. The goal is

to minimize the total operating time.

Braekers et al. (2013) study a vehicle routing problem for drayage operations. It includes loaded and empty 40-foot container movements, i.e. a full truckload vehicle routing problem is studied. The objective is to lexicographically minimize number of vehicles and the total travel distance. To deal with incomplete requests, an initial request completion phase is used to solve the problem. We borrow this idea of a request completion phase for our solution approach.

All previously mentioned papers consider full truckload transportation. In contrast to that, Vidovic et al. (2011) propose a model with less-than-truckload requests related to the VRP with backhauls. Empty and loaded container movements are considered. However, all empty containers are either picked up or delivered to the container terminal. That is, the possibility of including street-turns by using other customers as source or sink for empty containers is not considered.

Zhang et al. (2015) solve the container truck transportation problem for full and empty container drayage consisting of 20-foot and 40-foot containers. Inbound requests need the movement of a loaded container from a terminal and release an empty container after it is unpacked at the consignee. Outbound requests require an empty container to load the goods. These empty containers are picked up or delivered to a container depot or terminal. The utilization of a street-turn for empty container repositioning is not considered.

Schönberger et al. (2013) present a mixed integer model of the CPDP which takes into account that pickup or delivery locations for empty container requests have to be determined. However, a commercial solver was not able to compute solutions for instances of a practical relevant size in reasonable time. Recently, Funke and Kopfer (2015) propose a matheuristic approach for solving a multi-size container transportation problem. Partially specified requests as well as the use of a container depot and street-turns are considered. The model differs from the CPDP as it assumes that each container (empty or loaded, respectively) is immediately processed after arriving at the customer location and is hereafter available as loaded or empty container, respectively, and ready onward transport by the vehicle.

3 An Iterated Three-Phase Heuristic

To solve the CPDP an iterative three-phase is introduced and referred to as I3PH. An overview is given by Algorithm 1. Sequentially, the heuristic first solves the problem of completing all partially specified requests (Phase 1, see Section 3.1). With all requests defined, a PDPTW is solved in Phase 2 (see Section 3.2). Finally, Phase 3 modifies some model parameters used during Phase 1 and all three phases are repeated (see Section 3.3) until a termination criterion is met.

3.1 Phase 1: Request Completion

In Phase 1, the provide and store requests are completed. Both types of requests are denoted as partially specified requests (PSR). They involve transportation of empty containers. For each

Input: CPDP instance, no. of iterations

while *Weight adjustment not completed* **do**

 Phase 1: Request completion;

 Phase 2: Pickup and delivery problem;

 Phase 3: Weight adjustment;

return feasible tour plan;

Algorithm 1: Overview of the iterated three-phase heuristic I3PH

container size – only 20-foot and 40-foot a considered here – a container assignment problem (CAP) is solved via a MIP-solver. Both instances of the CAP model are linked via weighting parameters related to the travel distance δ_{20}, δ_{40} and temporal closeness τ_{20}, τ_{40} between locations that demand or supply 20-foot or 40-foot containers. The following holds in all phases of I3PH: $\delta_{20} + \tau_{20} \equiv 1$ and $\delta_{40} + \tau_{40} \equiv 1$ with $\delta_{20}, \delta_{40}, \tau_{20}, \tau_{40} \geq 0$.

The CAP is related to linear assignment problems (see e.g. Burkard et al., 2009). CAP is defined by formulas (1) to (4) and assumes *homogenous* container sizes. Therefore, the CAP is solve twice in Phase 1, once for all customers with 20-foot empty container requests and once for customers involving 40-foot empty container requests.

$$\min f(x) = \sum_{i=0}^m \sum_{j=0}^n (\delta \cdot d_{ij} + \tau \cdot t_{ij}^w) \cdot x_{ij} \quad (1)$$

$$s.t. \quad \sum_{j=0}^n x_{ij} = 1 \quad i = 1, \dots, m \quad (2)$$

$$\sum_{i=0}^m x_{ij} = 1 \quad j = 1, \dots, n \quad (3)$$

$$x_{ij} \in \{0, 1\} \quad i = 0, \dots, m \quad j = 0, \dots, n \quad (4)$$

Let $1, \dots, m$ denote pickup locations of empty containers (given in the CPDP as part of a store requests) and let $1, \dots, n$ denote delivery locations of empty containers (given in the CPDP as part of a provide requests). Index 0 denotes a container depot. We assume the storage capacity of the container depot is sufficiently large and there are at least as many empty containers available so that each provide request may be serviced from the container depot. The decision variables (4) are binary. If $x_{ij} = 1$ an empty container at customer i (pickup location) shall be transported to customer j (delivery location), $x_{ij} = 0$ otherwise. The constraints (2) ensure that each store request with a given pickup location i is assigned to one delivery location j (either the container depot or a customer with a provide request). On the other hand, constraints (3) ensure that each provide request with a delivery location j is assigned to one pickup location i (either the container depot or a customer with a store request).

The objective function (1) minimizes the total assignment costs. They depend on two criteria: First, on the Euclidean distance d_{ij} between nodes i and j weighted by parameter δ . Second, on the waiting time t_{ij}^w of a vehicle that visits node j after node i weighted by τ . Note, while distances are given parameters in the CPDP, the waiting time has to be extracted from a

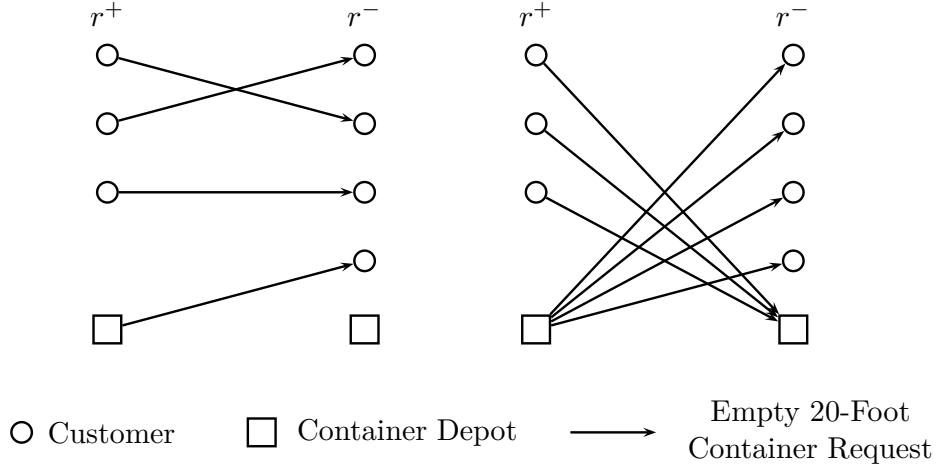


Figure 2: Two CAP solutions which complete empty container requests (left $\delta_{20} = 1$, right $\delta_{20} = 0$)

feasible solution of the CPDP.

Figure 2 shows two possible solutions when solving the problem for empty 20-foot containers with different weights. The purpose of the weights δ and τ is to balance the *geographical closeness* and the *temporal matching* (under time windows) of pickup and deliver locations. Furthermore, by setting other parameters of the CAP during preprocessing, it is easily possible to force or forbid assignments, e.g. because the involved time windows do not match or offer a clear disadvantage. The weights $\delta_{20}, \delta_{40}, \tau_{20}, \tau_{40}$ are updated in Phase 3 of the heuristic which results in a higher solution diversity.

From computational tests, it appears that the completion of 40-foot container requests is significantly more critical with respect to the performance of the heuristic I3PH. This is reasonable, because a 40-foot container request is always a FTL pickup and delivery request. If such a FTL completion decision is bad from the CPDP routing point of view, it is much harder to fix it during Phase 2 than to fix a bad decision involving a 20-foot container which leads to LTL request that can be integrated in more ways into different tours.

3.2 Phase 2: Pickup and Delivery Problem

Phase 2 of the heuristic I3PH solves a standard PDPTW. Many sophisticated approaches are discussed in the literature that can deal very well with this subproblem of the CPDP. The approach used here is straight forward. At first, a set of feasible single-vehicle tours is generated by means of an insertion procedure. Afterwards, a subset of these tours is selected via solving a set covering problem. That is, a subset of tours is selected such that each request is fulfilled and the total operation time of all vehicles is minimized.

In order to generate feasible one-vehicle tours for pickup and delivery requests, three well-known neighborhood moves from Li and Lim (2003) are used. The first move is the *shift* operator. It selects a request and tries to insert it into another tour. The request is shifted to that tour which leads to the highest decrease of the total operation time t^o . The move is applied to

Input: current operation time $f(x)$ of solution x , current minimum operation time $t^{o,best}$

```

( $\delta_{20}^{best}, \tau_{20}^{best}, \delta_{40}^{best}, \tau_{40}^{best}$ )  $\leftarrow$  (1, 0, 1, 0);
foreach  $c \in (40, 20)$  do
  for  $i \leftarrow 0$  to  $i = 4$  do
     $\delta_c \leftarrow 0.25 \cdot i$ ;          /*  $\delta_{20} + \tau_{20} \equiv 1$  and  $\delta_{40} + \tau_{40} \equiv 1$  always holds */
     $R \leftarrow \text{Phase } 1(\delta_c)$ ;      /* Complete requests */
     $x \leftarrow \text{Phase } 2(R)$ ;
    if  $f(x) < t^{o,best}$  then
       $t^{o,best} \leftarrow f(x)$ ;
       $\delta_c^{best} \leftarrow \delta_c$ ;

```

Algorithm 2: Weight adjustment in Phase 3

every request. The second move is the *exchange* operator. It tries to move a request from the current tour to another tour and in return, remove a request from the other tour and insert it into the current tour. It is applied to every request. The exchange move which results in the highest decrease of the total operation time t^o is realized. The third move is *rearrange*. It tries to switch the position of two customer nodes within a tour. The move with the highest decrease of the total operation time t^o is realized. The move continues with the next request in the current tour and continues with requests from other tours.

3.3 Phase 3: Weight Adjustment

As could be noticed from some computational experiments, it appears that the best values for the weighting parameters depend for the most part on the actual instance. Therefore, Phase 3 tries to tune these parameters at runtime of the heuristic I3PH. Phase 3 (see Algorithm 2) updates the weighting parameters δ and τ used in Phase 1 for request completion. In general, the change of these parameters leads to different types of requests and even a different number of requests. This diversity shall be exploited by changing the weighting parameters.

Phase 3 begins adjusting the weights δ_{40} and τ_{40} , if the majority of the PSR include 40-foot containers, else δ_{20} and τ_{20} are adjusted first. The parameter δ_{40} is decremented by one fourth while δ_{20} is constant. The value for δ_{40} which leads to the best solution is fixed and the search modifies the values δ_{20} . Note, due to $\delta_{20} + \tau_{20} \equiv 1$ and $\delta_{40} + \tau_{40} \equiv 1$ the values for τ_{20} and τ_{40} change accordingly. Finally, when the best weight combination has been identified, one final run with a significantly higher number of iterations is performed.

4 Results of Computational Experiments

In order to understand the performance of the heuristic as well as the effects of problem characteristics, a computational study was performed. Section 4.1 describes the setup of this study, Section 4.2 studies the effects of using different weights during Phase 1 of the heuristic, and Section 4.3 analyses the effect of different container sizes on the performance of the heuristic.

4.1 Setup of Computational Study

The heuristic was implemented in Java 8 and tested on an Intel Core I5, 2.5 Ghz CPU with 8 GB system memory. Within Phase 2 2,500 iterations to generate routes were used for test 100 customer instances and 1,000 iterations were used for 200 and 400 customer instances.

There are no benchmark instances in the literature for the CPDP with heterogenous container sizes. However, only recently Funke and Kopfer (2015) introduced instances which are related to CPDP but assume a slightly different model. Therefore, the well-known instances of Li and Lim (2003) for the PDPTW are extended. Six new sets of instances were generated based on the Li and Lim instances that include 100, 200 and 400 customers. From each original instance, for a percentage of *PSR-ratio* of requests either the pickup location or the delivery location has been deleted in order to generate partially specified requests. Also, the load of all requests requests was updated to either 1 TEU or 2 TEU.

To estimate the general performance of Phase 2, a comparison with solutions from the literature is possible. For some of the 100 customer instances, very good solutions with a deviation of less than three percent are computed by Phase 2. However, on average over all 100 customer instances, the performance is inferior compared to specialised state-of-the-art approaches for the PDPTW. These drawbacks appear negligible, as the focus is on the integration of simultaneously routing loaded and empty containers of heterogenous size, rather than developing a new heuristic for the well-known PDPTW.

4.2 Effect of Different Distance Weights on the Solution

The first test studies the effects of using different values for the weights δ_{20} and δ_{40} which are used in Phase 1. Table 1 shows the results for the test instance *plc205* with 50 percent partially specified requests and 20 percent 20-foot containers. The planning horizon of this instance is rather long, i.e., the choice of the values for δ_{20} and δ_{40} have a high impact on the solution quality as more waiting times and thus longer operational times are possible.

The first column of Table 1 shows the current iteration, the second and third column present the used weights δ_{20} and δ_{40} used in function (1). Note, $\delta_{20} + \tau_{20} \equiv 1$ and $\delta_{40} + \tau_{40} \equiv 1$ always holds. Columns four to seven show performance criteria of a solution: overall operation time t^o , total travel distance d , total waiting time of all vehicles t^w , and the number of required vehicles n . The three rightmost columns show performance criteria which refer *only* to the partially specified requests of this instance. The no. of requests states the number of fully defined requests which are generated from the given partially defined requests during Phase 1. Column d^{PSR} gives the total travel distance for the set of the partially defined requests, column t^{wPSR} states the total waiting time when fulfilling the partially specified requests.

The presented weight combinations were chosen as follows: The initial values are $\delta_{20} = 1.0$ and $\delta_{40} = 0.0$. Giving those customers with 40-foot containers the highest priority, because 80 percent of the customers require a 40-foot container, the value for δ_{40} was increased in 0.25 steps. Among these five tests, the minimum overall weighting time t^o was achieved for

Table 1: Effect of distance weights for the test instance plc205 (50% PSR, 20% 20-foot containers)

iteration	Weighting		Performance				PSR only		
	δ_{20}	δ_{40}	t^o	d	t^w	n	no. of requests	d^{PSR}	t^{wPSR}
1	1.00	0.00	26500	3254	10640	17	45	1361	2540
2	1.00	0.25	23620	2845	10868	13	45	1361	2540
3	1.00	0.50	23848	2665	11276	15	30	504	2360
4	1.00	0.75	24222	2696	11798	13	29	480	2408
5	1.00	1.00	28839	2920	16911	14	25	354	7851
6	0.75	0.25	23074	2712	9914	14	33	574	50
7	0.50	0.25	23103	3037	9619	13	33	574	50
8	0.25	0.25	23330	3040	9663	14	34	634	0
9	0.00	0.25	23314	2836	9669	15	35	668	0

$\delta_{40} = 0.25$. Now, the parameter δ_{40} is fixed to $\delta_{40} = 0.25$ while all remaining values for δ_{20} are tested in equidistant steps of 0.25. In terms of t^o the weighting $\delta_{20} = 0.75$ and $\delta_{40} = 0.25$ provided the best results.

As Table 1 reveals, setting adequate weights is very important for the performance of the heuristic. The worst solution in terms of overall operation time and total travel time is up to 30 percent and 22 percent worse, respectively. The impact of changing δ_{20} appears to be smaller than changing δ_{40} , especially with respect to the waiting time t^w . This effect may be due to the low quota of 20-foot containers in this instance. Additionally, an inappropriate assignment of 20-foot containers has only a minor effect on the total waiting time t^w , because instead of waiting a vehicle transporting a 20-foot container can perform a second request at the same time. Therefore, minimizing the total travel distance becomes more important with an increasing number of 20-foot containers and the distance weight δ_{20} should be set to a higher value.

With respect to the distance weight of 40-foot containers, lower values are usually better, since a good performance for the transport of 40-foot containers depends on short waiting times. When transporting a 40-foot container, there is no other option but to deliver it to the assigned customer. If the time window of said customer is very different from the time window of the pickup node, the vehicle has no other option but to wait until the customer starts its service. Therefore, temporal closeness is very important for 40-foot container movements. However, if the value of the distance weight for 40-foot containers is $\delta_{40} = 0$, every provide and store request for a 40-foot container is served by the container depot. This leads to a significant increase in total distance d as shown Table 1 because every empty container request is served by the container depot.

4.3 Effect of Different Mixtures of Container Sizes

The influence of an increasing ratio of 20-foot containers among all serviced containers is studied. Everything equal, more 20-foot containers implies more comprehensive options for routing

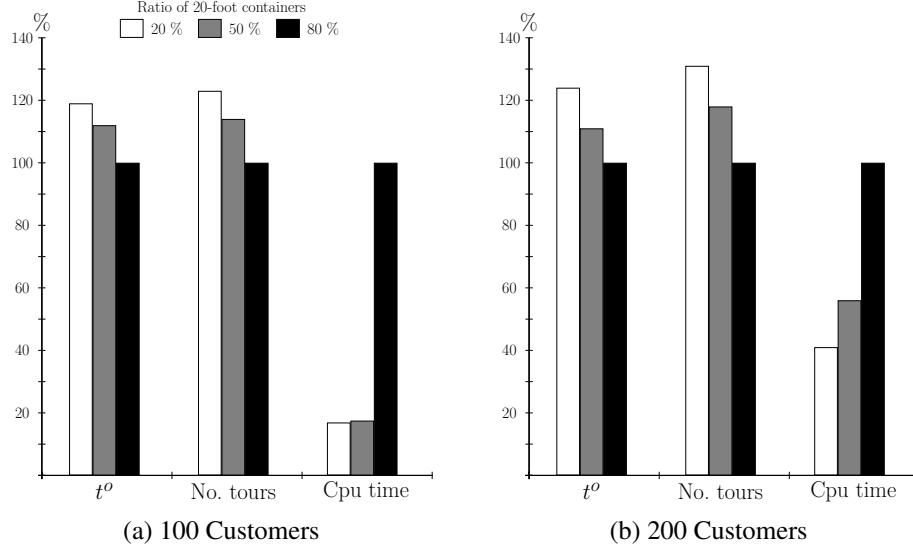


Figure 3: Influence of different percentages of container sizes on solution and performance

which should increase the difficulty of an instance of the CPDP. For this test, the results from all test instances were aggregated over each subset of instances. Table 2 shows the median, mean and standard deviation for the total operation time t^o , the number of tours n and the required computing times in seconds.

The results reveal that the heuristic works as expected. An increasing ratio of 20-foot containers leads to a significant decrease of operation time t^o and in the number of required tours n . However, the computational time increases due to more options for routing. Figure 3 underlines this effect. It shows the averaged results over all 100 and all 200 customer instances at which the averaged results for the group of instances with 80 percent 20-foot containers was defined as 100 percent. Figure 3 emphasizes the significant influence of the percentage of 20-foot containers on the solution and performance of the algorithm. Again, the inability of carrying multiple 40-foot containers simultaneously is probably the main reason for higher operation times which are caused by higher waiting times. Two findings may be inferred from Table 2. First, considering the transport of heterogenous container sizes like 20-foot and 40-foot containers simultaneously might increase the efficiency of planning significantly and lead to lower costs solutions. Second, in instances with heterogenous container sizes, partially specified requests with a higher load (i.e., 40-foot containers) appear to have a significantly higher effect on the total performance of the algorithm than those with a lower load (i.e., 20-foot containers).

5 Conclusion and Outlook

A iterated three-phase heuristic for the container pickup and delivery problem (CPDP) has been introduced. The CPDP generalizes the PDPTW and arises in the hinterland of an intermodal container terminal. It deals with the transportation of 20-foot and 40-foot containers which may be empty or loaded. To solve the CPDP, the heuristic has to deal with partially specified requests. The performance of the heuristic has been discussed on the basis of computational

experiments for over 1,000 test instances. To improve the quality of the heuristic, future research should try to analyse solutions in more comprehensive way in Phase 3 and use this data to find better solutions to the request completion problem of Phase 1 in the sense, that a more efficient routing of the vehicles and containers is possible.

Acknowledgements

The cooperative junior research group on Computational Logistics is funded by the University of Bremen in line with the Excellence Initiative of German federal and state governments.

References

- Funke, J. and H. Kopfer (2015). “A Neighborhood Search for a Multi-size Container Transportation Problem”. In: *IFAC-PapersOnLine* 48.3. 15th {IFAC} Symposium on Information Control Problems in Manufacturing (INCOM 2015), pp. 2041–2046. DOI: [10.1016/j.ifacol.2015.06.389](https://doi.org/10.1016/j.ifacol.2015.06.389).
- Zhang, R., W. Y. Yun, and H. Kopfer (2015). “Multi-size container transportation by truck: modeling and optimization”. In: *Flexible Services and Manufacturing Journal* 27.2-3, pp. 403–430. DOI: [10.1007/s10696-013-9184-5](https://doi.org/10.1007/s10696-013-9184-5).
- Braekers, K., A. Caris, and G. Janssens (2013). “Integrated planning of loaded and empty container movements”. English. In: *OR Spectrum* 35.2, pp. 457–478. DOI: [10.1007/s00291-012-0284-5](https://doi.org/10.1007/s00291-012-0284-5).
- Schönberger, J., T. Buer, and H. Kopfer (2013). “A Model for the Coordination of 20-foot and 40-foot Container Movements in the Hinterland of a Container Terminal”. In: *Proceedings of the 4th International Conference on Computational Logistics (ICCL'13)*. Ed. by D. Pacino and S. V. R. Jensen. Vol. 8197. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 113–127. DOI: [10.1007/978-3-642-41019-2_9](https://doi.org/10.1007/978-3-642-41019-2_9).
- Sterzik, S. and H. Kopfer (2013). “A Tabu Search Heuristic for the Inland Container Transportation Problem”. In: *Computers & Operations Research* 40.4, pp. 953–962. DOI: [10.1016/j.cor.2012.11.015](https://doi.org/10.1016/j.cor.2012.11.015).
- Vidovic, M., G. Radivojevic, and B. Rakovic (2011). “Vehicle routing in containers pickup up and delivery processes”. In: *Procedia - Social and Behavioral Sciences* 20. 14th Euro Working Group on Transportation, pp. 335–343. DOI: [10.1016/j.sbspro.2011.08.039](https://doi.org/10.1016/j.sbspro.2011.08.039).
- Burkard, R. E., M. Dell’Amico, and S. Martello (2009). *Assignment Problems*. Siam. Philadelphia, PA, USA.
- Jula, H., M. Dessouky, P. Ioannou, and A. Chassiakos (2005). “Container movement by trucks in metropolitan networks: modeling and optimization”. In: *Transportation Research Part E: Logistics and Transportation Review* 41.3, pp. 235–259. DOI: [10.1016/j.tre.2004.03.003](https://doi.org/10.1016/j.tre.2004.03.003).
- Li, H. and A. Lim (2003). “A Metaheuristic for the Pickup and Delivery Problem with Time Windows”. In: *International Journal on Artificial Intelligence Tools* 12.02, pp. 173–186. DOI: [10.1142/s0218213003001186](https://doi.org/10.1142/s0218213003001186).
- Savelsbergh, M. W. P. and M. Sol (1995). “The General Pickup and Delivery Problem”. In: *Transportation Science* 29.1, pp. 17–29. DOI: [10.1287/trsc.29.1.17](https://doi.org/10.1287/trsc.29.1.17).

Table 2: Aggregated results for different sets of test instances

instance set			t^o			n			computing time (s)		
customer	PSR ratio [%]	20ft-ratio [%]	median	mean	deviation	median	mean	deviation	median	mean	deviation
100	25	20	4370,5	8002,7	7142,3	14,5	15,4	6,6	17,5	40,4	69,5
		50	4070,5	7517,7	6654,8	14,0	14,2	5,7	22,5	44,8	76,4
		80	3585,0	6666,2	5464,5	12,0	12,4	4,7	36,0	117,6	251,8
	50	20	3973,5	7517,8	6422,3	14,0	14,8	6,4	18,0	50,2	97,1
		50	3878,0	7056,6	5774,9	14,0	13,7	5,7	26,0	48,4	66,9
		80	3452,0	6398,4	4943,6	12,0	12,1	5,0	57,5	416,6	1256,9
200	25	20	22760,0	25008,3	15055,7	30,5	30,7	10,9	21,5	59,0	84,7
		50	20166,5	22199,3	12882,3	27,0	27,6	9,4	82,6	82,6	82,6
		80	16671,0	19641,0	11284,3	23,5	23,3	7,3	101,0	145,7	149,1
	50	20	20796,0	22489,9	12550,7	28,0	29,2	10,9	33,5	69,0	98,3
		50	17356,0	20474,0	11314,1	26,0	26,4	9,3	56,0	94,3	115,2
		80	16382,5	18670,4	9853,7	21,0	22,3	7,9	141,5	168,0	158,7
400	25	20	50989,0	56365,1	29759,1	56,0	59,5	22,1	41,0	209,2	450,6
		50	47493,0	51203,7	24784,4	52,5	53,1	19,6	106,5	269,6	496,5
		80	43280,0	44692,3	20861,7	44,0	45,7	16,3	222,0	559,7	1263,4
	50	20	48974,5	54931,9	29423,6	57,5	57,4	24,6	80,5	270,5	608,0
		50	49449,0	49923,9	22973,6	49,0	50,5	20,6	119,0	328,0	615,4
		80	39478,5	44121,4	21717,3	45,0	45,6	18,2	203,5	430,4	658,5