RETAIL WAREHOUSE LOADING DOCK COORDINATION BY CORE-SELECTING PACKAGE AUCTIONS

Complete Research

Paul Karaenke, TU München, Garching, Germany, karaenke@in.tum.de

Martin Bichler, TU München, Garching, Germany, bichler@in.tum.de

Stefan Minner, TU München, Munich, Germany, stefan.minner@tum.de

Abstract

Congestions at loading docks can cause severe delays in logistics processes and cause increasing bottlenecks for truck routes. For warehouses, uncoordinated arrivals of trucks make appropriate staffing difficult and congestions can interfere with other processes at the facility. To mitigate congestions at loading docks, we propose package auctions to allocate time slots to trucks.

The contribution of this research is the application of core-selecting package auctions to address the loading dock congestion problem. We propose a bidding language and a core-selecting package auction for this setting based on existing literature. Core-selecting payment rules can avoid drawbacks of the Vickrey–Clarke–Groves (VCG) mechanism with Clarke pivot rule, e.g., low perceived fairness of prices. We evaluate our proposal by means of simulation and assess (i) the potential for waiting time reduction compared to uncoordinated arrivals as well as sharing of historical waiting times, (ii) the empirical complexity of the computational problem for scenarios of varying complexity, and (iii) the relation of VCG and bidder-Pareto-optimal core payments. Our findings provide evidence that loading dock auctions can alleviate congestion substantially and that the core-pricing rule is well-suited to address the price fairness and low seller revenue problems in this setting.

Keywords: logistics, package auction, core-selecting auction, market design

1 Introduction

Less stock-keeping at factories and retailers has increased the pressure on carriers to deliver on time. With more trucks, probabilities for congestions rise. These do not only occur on roads, but especially at warehouses. Congestions at loading docks can cause severe delays in logistics processes and cause increasing bottlenecks for truck routes. The congestion incidents result from temporary capacity overload or capacity reductions due to shortage of staff at warehouses.

According to a survey among more than 500 transport companies, 18% of them have an average waiting time of more than two hours and 51% have an average waiting time between one to two hours at each warehouse (Bundesverband Güterkraftverkehr Logistik und Entsorgung e.V., 2013). Another problem is insufficient parking capacity for waiting trucks. The German Federal Office for Goods Transport has provided a special report on problems at loading docks (Bundesamt für Güterverkehr, 2011). The main reasons for waiting times include shortages of resources (staff and infrastructure) and uncoordinated arrivals of trucks, especially at peak times. These problems are interconnected, since uncoordinated arrivals of trucks make appropriate staffing difficult. In addition, congestions can interfere with other

processes due to a high number of trucks at the facility. Proposed remedies for these problems are time slot management, information sharing, and increased infrastructure capacities.
Capacity increases require rather high investments compared to improved coordination by information sharing or time slot management. We have investigated the provision of information about the historical waiting times to the carriers in previous work (Lemke, Bichler, and Minner, 2014). Carriers can utilize this information by changing their routes and schedules accordingly. Since waiting time information is the same for all carriers, there is a risk that they make similar decisions, e.g. delaying the departure to avoid waiting times that occur in the morning, and therefore cause new congestion at another time of the day.
To mitigate congestions at loading docks, we propose the application of package (combinatorial) auctions to allocate time slots to trucks. The contribution of this research is the application of core-selecting package auctions for truck coordination to address the loading dock congestion problem. We propose a bidding language and a core-selecting package auction for this setting based on existing literature. Core-selecting payment rules have been applied in spectrum auctions and can avoid several drawbacks of the Vickrey–Clarke–Groves (VCG) mechanism with Clarke pivot rule, e.g., low perceived fairness of prices (Ausubel and Milgrom, 2002; Day and Milgrom, 2008; Day and Raghavan, 2007).
We use a discrete event simulation to evaluate our proposal and assess (i) the potential for waiting time reduction compared to uncoordinated arrivals as well as sharing of historical waiting times from previous work, (ii) the empirical complexity of the computational problem for scenarios of varying complexity, and (iii) the relation of VCG and bidder-Pareto-optimal core payments. We generate transportation networks based on the CATS algorithm (Leyton-Brown, Pearson, and Shoham, 2000) and generate bids based on the expected round trip times and departure delays. Our findings provide evidence that loading dock auctions can alleviate congestion substantially and that the core-pricing rule is well-suited to address the price fairness and low seller revenue problems in this setting.
The remainder of this paper is structured as follows. In Section 2, we give an overview of related work for both the logistics problem addressed as well as the theories utilized. We describe our research method and the relation to fundamental theory in section 3. Section 4 presents the proposed artifact along with the basic assumption. We present the experimental design and results in section 5 and discuss our findings, implications for practice, as well as limitations and future research in section 6. Section 7 concludes.

2 Literature review

Next, we give an overview of related work for the coordination problem addressed and related approaches. We also provide details of the theories utilized to construct and evaluate the proposed artifact.

2.1 Loading dock coordination

The general problem setting addressed in this paper is comparable with multiple instances of the time-dependent traveling salesman problem (TDTSP). In these problems the dependency of travel times on the time of the day is considered as a generalization of the traditional TSP where constant travel times are assumed. The problem and first integer programming formulations have been introduced by Fox (1973), Fox, Gavish, and Graves (1980), and Picard and Queyranne (1978).
Ichoua, Gendreau, and Potvin (2003) present a model for the related time-dependent vehicle routing problem (where a fixed size fleet has to visit multiple nodes), satisfying the "first-in–first-out" property which ensures that trucks cannot arrive earlier by leaving later, and provide an overview of related TDTSP formulations. Abeledo et al. (2010) study the TDTSP formulation by Picard and Queyranne (1978) and provide computational results with a branch-cut-and-price algorithm for several families of facet-defining cuts of the TDTSP polytope. Kok, Hans, and Schutten (2012) investigate congestion avoidance on the edges of the traveling graph. These congestions are caused by the whole traffic on roads and the number of trucks considered is too small to influence congestions. This does not hold for our scenario where the

performance is influenced by the fraction of carriers winning time slot auctions. Therefore, the approach cannot be applied to the problem addressed.

In out setting, for every truck, a TDTSP has to be solved. In contrast to existing approaches, the problem addressed in the paper also results from the constrained capacity of the *vertices* (as compared to edges) of the traveling graph. That is, additional virtual edges are added to the traveling graph to represent the loading and unloading processes. We also assume that there is no central coordinating entity with perfect information, i.e., solutions interact with those of the other carriers and the resulting round trip times are therefore interdependent; carriers optimize at the same time and their optimizations impact each other.

2.2 Mechanism design and auctions

Mechanism design and auction theory provide mature and rigor methods to build and analyze market designs. In auctions, a mechanism consists of an allocation rule and pricing rule. The former allocates the items among the bidders based on their reported types (or bids). The latter determines the prices the bidders have to pay. The Vickrey–Clarke–Groves (VCG) mechanisms with Clarke pivot rule (Clarke, 1971; Groves, 1973; Vickrey, 1961) constitutes the only strategy-proof mechanism that maximizes social welfare when payments from losing bidders are zero. In package auctions, the representation of bids must be encoded in a bidding language. Since for package auctions with m items there are $2^m - 1$ non-empty subsets, succinct representations of bids are often required for practical applications (Nisan et al., 2007).

Although VCG is the only efficient and strategy-proof mechanism, it can result in unacceptably low perceived fairness of prices, low seller revenue, provide incentives for bidder to use shills, and provide incentive for sellers to exclude qualified bidders. Core-payments can mitigate these drawbacks and a non-empty core always exists in auction problems. An individually rational auction outcome is in the core if there is no group of bidders who would strictly prefer an alternative outcome that is also strictly better for the seller; i.e, there is no group of bidders who "would pay more" than the current payments. Detailed descriptions of the basic concept and examples illustrating the differences to VCG outcomes can be found in (Day and Milgrom, 2008; Day and Raghavan, 2007; Day and Cramton, 2012).

Although core-selecting package auctions provide incentives to untruthful bidding, as Day and Raghavan (2007) show, a payment rule that minimizes total payments in the core also minimizes incentives to deviate from truthful bidding. Whenever the VCG outcome is in the core, it is selected by a bidder-Pareto-optimal core mechanism (Ausubel and Milgrom, 2002).

Day and Milgrom (2008) and Day and Raghavan (2007) show that there exists a complete information Nash equilibrium in deviation strategies for every bidder-Pareto-optimal point in the core, though it is not necessarily an equilibrium when bidders' valuations are private information. In complete information settings several Nash equilibria can exist, i.e., bidders have to coordinate to achieve a specific equilibrium. In real-world settings, the cost of acquiring information required to meaningfully exploit the deviation potential of the proposed mechanism is commonly assumed to exceed the potential benefit of deviation (Day and Raghavan, 2007). In this work we assume that bidders do not deviate from truthful bidding and hence the core-selecting auction will yield efficient results. This assumption does not hold in general and has to be taken into account when interpreting our findings. We will investigate untruthful bidding in our setting in future work.

Another approach to address the issues that can arise from VCG with Clare pivot rule are deferred-acceptance (DA) auctions (Dütting, Gkatzelis, and Roughgarden, 2014; Milgrom and Segal, 2014a,b). These auction maintain strategyproofness, but sacrifice efficiency to address the low revenue/high price problem. In addition, these auctions are weakly group-strategyproof. However, existing analysis for DA is limited to single-minded bidders. We leave the investigations of an application of DA auctions in our setting for future work.

3 Method

We follow the design science research (DSR) approach proposed by Hevner et al. (2004). The DSR paradigm – based on engineering and the sciences of the artificial (Simon, 1996) – is a problem-solving paradigm. It targets the *construction and evaluation* of IT artifacts, enabling organizations to address information-related tasks (Gregor and Jones, 2007; Hevner et al., 2004). IT artifacts are defined as constructs (vocabulary and symbols), models (abstractions and representations), methods (algorithms and practices), and instantiations (implemented and prototype systems). The artifacts proposed in this research are a domain-specific adaption and application of core-selecting package auctions to address the loading dock congestion problem. The conceptual loading dock auction and bidding language constitute a method artifact, while the concrete implementation to gain insight about an application of core-selecting package auctions to the problem addressed in domain-specific settings constitutes an instantiation artifact.

DSR relies upon the application of rigorous methods in both the construction and evaluation of the design artifact. We therefore describe the theories that inform the construction and evaluation of the proposed artifact in the following. The utility, quality, and efficacy of a design artifact must be rigorously demonstrated via well-executed evaluation methods (Hevner et al., 2004). Here, we apply simulation experiments and provide evidence that loading dock auctions can alleviate congestion and that the core-pricing rule is well-suited to address the price fairness and low seller revenue problems in this setting. We follow the structural guidelines for presenting DSR proposed by Gregor and Hevner (2013).

The rationale for selecting game theory to inform the construction of the artifact is as follows. Truck waiting times at loading docks have economic impacts for the carriers. Therefore, reservations for loading docks can be priced and the coordination can be addressed by auctions. Mechanism design and auction theory provide mature and rigor methods to build and analyze market designs (Nisan et al., 2007).

The evaluation of the proposed artifact is informed by simulation literature. The simulation of economic system is a well-established method to evaluate artifacts and can be used to numerically analyze the artifact to estimate the true systems characteristics (Law and Kelton, 1999). We evaluate both the estimated implications for loading dock coordination by the proposed artifact as well as the empirical complexity of the required computations.

4 Artifact description

We describe the loading dock auction by first introducing the basic assumptions and the proposed bidding language. We then describe the winner determination problem and the core-selecting payment rule.

4.1 Basic assumptions

There is a set of carriers and warehouses, each having a specific location. The roads between these locations are defined in a transportation network which also constrains the routes. Each carrier is given a tour that is a random subset of the set of warehouses.

We assume that each *carrier* has one truck only and that the truck has a sufficient capacity to fulfill the orders. The truck starts at the depot and returns to the deport again after completion of the tasks.

Warehouses are facilities where trucks have to be loaded or unloaded. This can for example be a retailer that is supplied with goods. A warehouse can serve a pre-defined number of carriers at the same time. Warehouse (un)loading capacity may vary for both different warehouses and different time slots. When a warehouse is fully occupied and/or reserved, other carriers have to line up and wait until the warehouse employees can serve them.

A *tour* denotes a set of warehouses of a carrier to visit. Carriers have to deliver freight to a warehouse, pick it up there or both. A tour does not define an order, in which the warehouses are to be visited.

A *route* is an ordered tour. Routes define orders in which tours are processed. Routes thus define paths in the transportation network graph. To form a suitable route for a tour, carriers have to order the locations on the tour. This can be done for example by solving a Traveling Salesman Problem (TSP).
The auction allocates warehouse loading dock time slots to bidders. That is, routes are not considered directly but encoded in bids and the TSPs for the trucks are solved before the auction starts. Bidders are allowed to bid on alternative routes implicitly. We assume that the valuations for route reservations are inversely proportional to the round trip times and, with a reduced factor, to the departure delays.
The *payments* for the auction are collected by a central entity that is carrying out the auction on behalf of the warehouses. That is, a service provider for time slot management acts on behalf of the warehouses to improve coordination. The payments might be partly distributed to the warehouses to provide incentives to participate, though concrete distributions of payments are beyond the scope of this work.

4.2 Bidding language

The service capacity of warehouses (loading docks) is modelled as a multi-knapsack problem. In each time slot $t \in T = \{1,2,\dots,\mathscr{T}\}$, each warehouse $k \in K = \{1,2,\dots,\mathscr{K}\}$ has a capacity of $c_k = (c_{k1},\dots,c_{k\mathscr{T}})$. That is, warehouse k can service up to c_{kt} trucks in time slot t. The carriers (bidders) $i \in I = \{1,2,\dots,\mathscr{I}\}$ submit zero or more tuples (b_j, R^j), where R^j denotes a $\mathscr{T} \times \mathscr{K}$ binary reservation matrix ($R^j = \{0,1\}^{\mathscr{T}\times\mathscr{K}} : r^j_{tk} \in \{0,1\} \forall t \in T, k \in K$) and b_j the monetary bid on the reservations described by this matrix. An example reservation matrix is shown in figure 1. The set of bids $J = \{1,2,\dots,\mathscr{J}\}$ is partitioned into subsets of bid indices of the bidders i such that $\forall i \in I \exists! J_i$ with $\bigcup_{i\in I} J_i = J$ and $\bigcap_{i \in I} J_i = \emptyset$.

$$R^1 = \left.\begin{pmatrix} \overbrace{\begin{matrix} 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 & 0 \end{matrix}}^{\text{warehouse}\,k} \\ \begin{matrix} 0 & 0 & 1 & \cdot & \cdot & \cdot & 0 & 0 \end{matrix} \\ \cdot \\ \cdot \\ \cdot \\ \begin{matrix} 0 & 0 & 0 & \cdot & \cdot & \cdot & 1 & 0 \end{matrix} \end{pmatrix}\right\} \text{time slot}\,t$$

Figure 1. Example reservation matrix

4.3 Winner determination

Let r^j_k denote the kth column of R^j. Then, the winner determination problem can be formulated as follows.

$$\begin{aligned} WD(I) = \max \; & \sum_{j\in J} b_j x_j && && \text{(WD)} \\ \text{s. t.} \; & \sum_{j\in J} r^j_k x_j \le c_k^\intercal && \forall k \in K, && \text{(1a)} \\ & \sum_{j \in J_i} x_j \le 1 && \forall i \in I, && \text{(1b)} \\ & x_j \in \{0,1\} && \forall j \in J. && \text{(1c)} \end{aligned}$$

To maximize the social welfare, the objective is to maximize the sum of accepted bids in WD. Constraint (1a) ensures that the warehouse capacities are not exceeded for accepted bids for every time slot. Constraint (1b) models the XOR relation of the bids and ensures that at most one bid can be accepted per bidder. (1c) restricts the bid acceptance decision to binary values.

4.4 Payment rule

Following Day and Raghavan (2007), we define the core separation problem, which yields the most violated core constraint, if any. Let b_i^* denote the bid of winning bidder i from the set of winners $W \subseteq I$ and let $p^\tau = (p_1^\tau, \dots, p_{\mathscr{I}}^\tau)$ denote the payment vector in iteration τ. To calculate equitable bidder-Pareto optimal (EBPO) core payments iteratively, the procedure is as follows. Solve the core separation problem (SEP$^\tau$):

$$
\begin{aligned}
z(p^\tau) = &\max \sum_{j \in J} b_j x_j^\tau - \sum_{i \in W} (b_i^* - p_i^\tau)\gamma_i^\tau && && \text{(SEP}^\tau\text{)}\\
&\text{s. t.} \sum_{j \in J} r_k^j x_j^\tau \leq c_i^\intercal && \forall k \in K, && \text{(2a)}\\
&\sum_{j \in J_i} x_j^\tau \leq 1 && \forall i \in I \setminus W, && \text{(2b)}\\
&\sum_{j \in J_i} x_j^\tau \leq \gamma_i^\tau && \forall i \in W, && \text{(2c)}\\
&x_j^\tau \in \{0,1\} && \forall j \in J. && \text{(2d)}\\
&\gamma_i^\tau \in \{0,1\} && \forall i \in W. && \text{(2e)}
\end{aligned}
$$

(SEP$^\tau$) yields the most violated core constraint, if any. That is, it finds coalitions of bidders C^τ that block the current outcome (who "would pay more" than the current payments). Let $p^{core,\tau} = (p_1^{core,\tau}, \dots, p_{\mathscr{I}}^{core,\tau})$ denote the (temporary) core-payment vector in iteration τ. Then, the minimal payments in the core satisfying the core constraints found (CORE) are calculated (EBPO$^\tau$):

$$
\begin{aligned}
\theta^\tau(\varepsilon) = &\min \sum_{i \in W} p_i^{core,\tau} + \varepsilon m^\tau && && \text{(EBPO}^\tau\text{)}\\
&\text{s. t.} \sum_{i \in W \setminus C^{\tau'}} p_i^{core,\tau} \geq z(p^{\tau'}) - \sum_{i \in W \cap C^{\tau'}} p_i^{\tau'} && \forall \tau' \leq \tau, && \text{(CORE)}\\
&p_i^{core,\tau} - m^\tau \leq p_i^{vcg} && \forall i \in W, && \text{(3a)}\\
&p_i^{core,\tau} \leq b_i^* && \forall i \in W, && \text{(3b)}\\
&p_i^{core,\tau} \geq p_i^{vcg} && \forall i \in W. && \text{(3c)}
\end{aligned}
$$

The minimal payments minimize potential gains from deviation and EBPO minimizes the maximum deviation from VCG payments as a secondary objective. This procedure is repeat until no further core constraint violation is found using (SEP$^\tau$), i.e., it is repeated while $z(p^\tau) > \theta^{\tau-1}(\varepsilon)$ with $\theta^0(\varepsilon) := \sum_i p_i^{\text{vcg}}$. The procedure is described in pseudo code in algorithm 1 in appendix A.1.

5 Evaluation

In this section, we provide details of our experimental evaluation. We use a discrete event simulation and study (i) the potential for waiting time reduction compared to uncoordinated arrivals as well as sharing of historical waiting times from previous work (Lemke, Bichler, and Minner, 2014), (ii) the empirical complexity of the computational problem for scenarios of varying complexity, and (iii) the relation of VCG and bidder-Pareto-optimal core payments. We next describe the setup of the simulation experiments, followed by a presentation of the results.

5.1 Simulation setup

We describe the three different coordination scenarios in the following. Then, we provide details about transportation network, tour and route generation. Finally, we give details about the concrete process and parameters used in the simulations.

Scenario 1: No coordination In this scenario there is no information available to the carriers. Each carrier does his own planning and then starts to process the plan. Arriving at a warehouse, the carrier has to queue up if the warehouse is occupied. This happens when many carriers arrive within a short period. In the uncoordinated case, the carriers are assumed to plan their route optimally with respect to the distance they have to cover while visiting all warehouses on the tour. However, they cannot consider waiting times or reservations because there is no information or mechanism available to them. In the field, carriers might accumulate some information and an estimate of the waiting times over time, but we ignore this in our model for simplicity.

Minimizing the distance essentially means to solve a TSP. For our study it is not important that the carriers take the shortest route, because in this study the waiting times at the warehouses are central, not the optimization algorithms. Also in the field carriers might not always take the optimal route, but one that is close to optimality. We use an exact algorithm based on a branch and bound strategy (Dantzig, Fulkerson, and Johnson, 1954; Held and Karp, 1970) to solve the TSPs for the carriers, which simulates the routing decisions of the carriers.

The departure time, at which the carriers depart at their depots and start to process their routes, is assumed to be the same for all carriers in the morning, but their depot locations and routes may be different. Since we assume higher valuations for earlier departure times (when the shifts start) and no information is available to minimize waiting times there is no flexibility of departure times in this scenario.

Scenario 2: Coordination by information In the second scenario, as we have initially investigated in previous work, we assume that the warehouses publish the average waiting time that carriers had for each hour after the day. Based on this information, carriers can adapt their plans the next day in order to minimize the waiting time. Optimizing the route and departure time with respect to expected waiting times that vary within the day can be formulated as a time-dependent TSP (TDTSP), which is a generalization of the traditional TSP (Abeledo et al., 2010; Ichoua, Gendreau, and Potvin, 2003). As well as the TSP, the TDTSP is also NP-hard and difficult to compute optimally even for small instances.

Therefore, in practice carriers often split the problem into route selection and determination of the departure time (Kok, Hans, and Schutten, 2012). We also follow this approach, because we expect this to be a realistic and computationally simple procedure that carriers might follow.

The route selection is done the same way as in scenario 1 by solving the TSPs. After that, a departure time has to be defined. We have proposed an algorithm that computes the optimal departure time for a given route by using the published waiting time information in previous work (Lemke, Bichler, and Minner, 2014). The algorithm is as follows. For every departure time between the start and end of the day, the expected round trip time (RTT) is calculated, and the earliest possible departure with the minimum expected RTT is selected. If the departure time plus the expected RTT exceeds the end of the day value, the route will be discarded because in practice the RTT will be even greater and warehouses might already be closed after the arrival. The expected RTT is calculated as the sum of travel time, service times, and expected waiting times according to the information provided publicly. Algorithm 2 in appendix A.2 provides pseudo code on how the departure time is determined. How the expected RTT for a given route and departure time is calculated is described in algorithm 3 in appendix A.3.

Scenario 3: Coordination by core-selecting loading dock package auctions The third scenario is based on the artifact proposed in this research. Routes are not considered directly in the loading dock auctions but are encoded in the bids. The carriers therefore have to solve TSPs for the trucks before the auctions start.

Since bidders are implicitly allowed to bid on alternative routes, we generate all possible permutations for each route and discard those alternatives that have an expected RTT beyond a certain threshold in terms of a multiple of the minimal RTT. We denote this factor by route discard factor ρ. For example, a factor of $\rho = 1.1$ for this threshold means we discard all alternative routes that exceed the minimal route by more than 10% of additional RTT.

We then generate bids on all remaining alternative routes. The rationale for creating alternative routes is that in expectation not every minimal route can be reserved for all bidders, since these routes would require conflicting reservations at the same warehouses. We assume that the valuations for route reservations are inversely proportional to the RTTs. That is, bids on alternative routes that have a larger expected RTT will contain a lower monetary bid on the reservations required. We calculate the expected RTT, $E(rtt)$, by algorithm 3 in appendix A.3 with expected waiting times of zero. Then, we use the number of time slots determining the resolution for the auctions, $\mathscr{T}$, and subtract the expected RTT for the route of bid j, $E(rtt_j)$, i.e., $b_j = \mathscr{T} - E(rtt_j)$. The resulting value is then inversely proportional to the RTT, non-negative, and can be scaled to realistic monetary values. For example, the German Federal Office for Goods Transport has identified a willingness to pay of about EUR 2.5 to 3.5 per loading dock reservation in a survey (Bundesamt für Güterverkehr, 2011, p. 25).

In addition to bids on alternative routes, we consider bids on alternative departure times. The different departure times considered in bids are determined by the bid slot length factor λ. We then generate bids for alternative departure times

$$d_j \in \{d_a : d_a = a \cdot \lambda \cdot \mathscr{T} \text{ with } d_a + E(rtt_j) \leq \mathscr{T}, a \in \mathbb{N}_0\},$$

i.e., for all departure times that are integral multiples of the bid slot length factor and the number of auction time slots for which the route ends before the defined end of the day in expectation.

Here, we assume that the valuations for route reservations are inversely proportional to departure delays, though with a reduced factor as compared to higher RTTs. We denote this factor by bid delay factor δ. For example, using a factor of $\delta = 0.1$ of RTTs to include the departure delays in the bidding model means that one time unit of longer RTT is 10 times as bad as a delayed departure by one time unit. That is, we finally calculate the monetary bids, scaled by functions $s_j(\cdot)$, as

$$b_j = s_j(\mathscr{T} - E(rtt_j) - \delta \cdot d_j).$$

Transportation networks The transportation network is a central part of our analysis and the networks need to be realistic for external validity of the simulation results. This graph defines the distances and the connectivity of the different locations like carrier depots and warehouses. The distance is relevant to determine travel times between two locations. Based on the graph the carriers determine their routes in order to visit all warehouses on their tour starting and ending at their depot. For the simulation we do not use a real-world network but generated ones, based on the CATS generator (Leyton-Brown, Pearson, and Shoham, 2000).

Although, in contrast to the generated networks, many real-world networks exhibit scale-free properties, we argue that the CATS graphs are realistic because they feature many properties that are special to transportation networks. Further details about the the network generator can be found in previous work (Lemke, Bichler, and Minner, 2014).

Tours and routes Before we generate tours, we randomly place carriers and warehouses in the network. Each carrier and warehouse has a location defined by a node in the transportation network. The nodes are used exclusively, meaning that a node which already defines the location of a carrier or warehouse may not be used for another one. The locations for the carriers and warehouses are drawn uniformly without replacement from the set of nodes of the transportation network. After the carriers and warehouse are placed in the transportation network, the carriers are given complete tours before the first simulated time period begins.

Each tour is generated by taking a sample from the set of warehouses without replacement. The size of the sample is taken from a distribution which is a parameter of the simulation. Each carrier, having a location for its depot and a tour of warehouses, can start the planning process, which consists of defining a route to visit all warehouses on the tour. The random subsets are generated by shuffling the list of warehouses and then to take the first n entries, where n is taken from the tour size distribution.

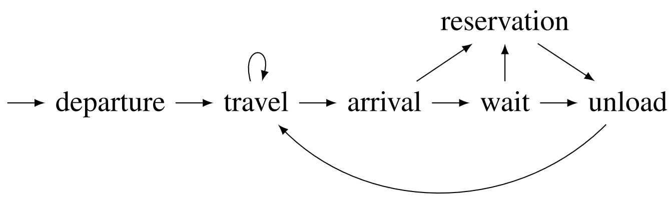


Figure 2. Sequence of events used in the discrete event simulation.

Simulation process and parameters In our simulation six types of events are used. The first event is the *departure event* which happens when a carrier leaves his depot. It results in a *travel event* representing the carrier's travel from his current location to his next one. The travel event is succeeded by an *arrival event*. It represents the carrier arriving at a warehouse and queuing up in order to being loaded or unloaded respectively. The arrival event is either followed by a *wait event*, or a *reservation event*. The latter is directly triggered only if the truck arrives at the exact time the reservation starts. If the truck arrives early or does not have a reservation, the wait event is triggered. If a truck has not arrived when the reservation would become active, the reservation event is discarded and the truck has to enqueue upon arrival.

The *unloading event* is triggered by the warehouse itself, whenever a carrier starts the loading/unloading process. Having completed the unloading process a carrier moves on to the next warehouse or his depot if he already completed his tour. The simulation ends when every carrier has reached his depot again. All possible event sequences are depicted in figure 2.

The transportation network graph is generated based on the number of locations. How many of the locations are carrier depots or warehouses is determined by the *number of carriers* ($\mathscr{I}$) and *number of warehouses* ($\mathscr{K}$). The number of carriers a warehouse can service at a time is denoted as warehouse capacity. To determine the time for the carriers to get from one location to another, the simulation uses the *travel time* parameter to scale the normalized edge weights of the transportation graph. The unloading times are denoted by parameter *unloading time*. The parameter for the number of warehouses a carrier has to visit is called *warehouses per carrier*. For the information sharing scenario, the fraction of carriers who use the information is defined by the parameter *load info user ratio*. The data for load information have a resolution determined by the parameter *slot length*. For each slot the warehouses compute the average waiting time that carriers experienced when arriving within this time slot.

In the auction-based scenario, we use additional parameters to generate the bids. The *route discard factor* (ρ) determines the threshold factor to discard alternative routes. When an alternative route has an expected RTT greater than this factor multiplied by the minimal expected RTT, the alternative route is discarded. The *bid delay factor* (δ) determines the weight of delayed departure in the generation of the monetary bids for routes. That is, a monetary bid is reduced by the relative delay as compared to the length of the day, multiplied by this factor. Finally, we construct bids for alternative departure times based on the *bid slot length factor* (λ). It determines the portions of the working day of which multiples are considered as alternative departure times as long as the expected end time is before the end of the day. For example, $\lambda = 0.25$ means that a departure after zero, one, two, and three quarters of the working day will be considered as departure times. Since four or more quarters obviously lead to an expected route end time beyond the end of the working day, these alternatives are not considered.

The values for each parameter in our simulations are shown in table 1. We used a fully factorial experimental design, so each parameter value was used against every combination of the remaining parameters. There are 140 combinations of basic parameters, for which different simulation experiments are conducted. One experiment with no coordination, nine for the different sharing ratios for the information sharing scenario with ten different settings for the available information (iterations) each, and three for the different bid slot length factors in the auction scenario. In total, we arrive at $140 \cdot (1+9 \cdot 10+3) =$ *13,160* parameter combinations. The simulation system is implemented in the Java programming language

parameter group	parameter	values
basic settings	number of locations	30
	number of carriers ($\mathscr{I}$)	10
	number of warehouses ($\mathscr{K}$)	10
	warehouse capacity	1
	travel time	$\{30,35,40,45,50\}$
	unloading time	$\{20,25,30,35,40,45,50\}$
	warehouses per carrier	$\{2,4,6,8\}$
info sharing	load info user ratio	$\{0.1,\ldots,0.9\}$
	number of iterations	10
	slot length	60
auction	route discard factor (ρ)	1.1
	bid delay factor (δ)	0.1
	bid slot length factor (λ)	$\{0.25,0.5,1.0\}$

Table 1. Simulation parameters

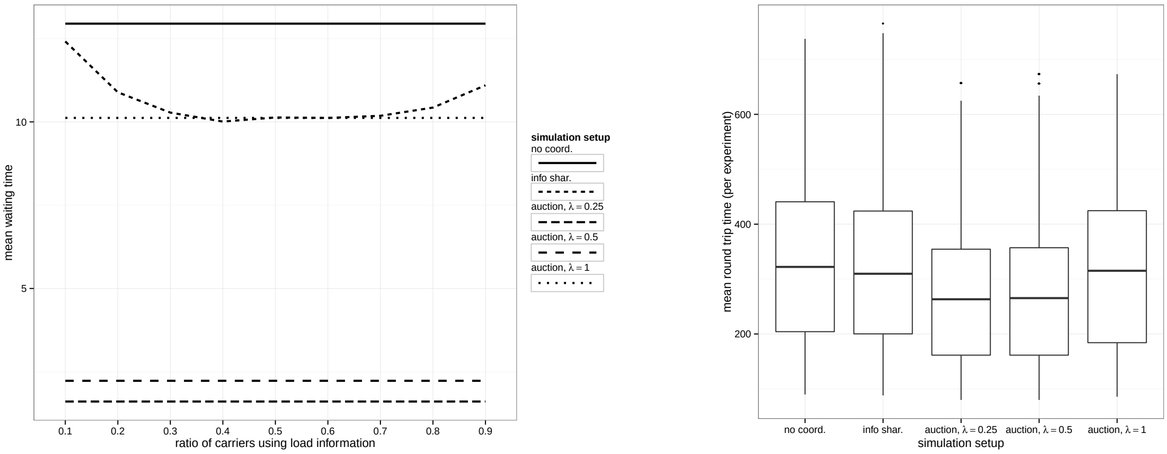
and the commercial mathematical programming solver Gurobi Optimizer v5.6.3 is used to solve WD, SEP, and EBPO. Experiments are executed on a machine with an Intel Core i7-3770 CPU (4 cores, 3.4GHz) and 16GB RAM.

5.2 Results

A sensitivity analysis of the basic parameters and an elaborate comparison of the uncoordinated and information sharing scenarios has been provided in previous work (Lemke, Bichler, and Minner, 2014). We give a brief summary of these results followed by a comparison to those of the auction-based scenario. We have identified a linear relationship between waiting time and unloading time and the effect of the tour size is rather small in comparison. The effect of changing the travel times changes the ratio of travel time and unloading time. If traveling takes longer, the chance of arriving while another truck is being unloaded decreases. Finally, the influence of the graph topology is small compared to other parameters. The results show that providing load information to the carriers reduces the average waiting time by up to 20% as well as the variance of waiting times.
The results of the evaluation show, as can be seen in figure 3 and table 2, that comparable results can already be achieved in the auction-based scenario with only one possible departure time per truck ($\lambda = 1$). Moreover, for additional departure times and therefore additional bids, the average waiting time is further reduced substantially. Compared to no coordination, the mean waiting time is reduced by more than 82.5% for $\lambda = 0.5$ and more than 87.5% for $\lambda = 0.25$. This results from increased possibilities of feasible combination of route reservations with an increased number of alternative bids, though this effect is mitigated by the common factor λ which is the same for all bidders.

		no coord.	info shar.	auction		
				$\lambda = 0.25$	$\lambda = 0.5$	$\lambda = 1.0$
waiting time	μ	12.95	10.62	1.61	2.23	10.12
	σ	22.38	19.94	17.57	14.93	33.42
round trip time	μ	336.02	324.36	279.04	282.96	321.67
	σ	151.08	144.55	134.97	141.29	150.55

Table 2. Waiting and round trip times (minutes)



(a) (b)

Figure 3. Mean waiting time (a) and round trip time (b) of the different scenarios (minutes)

However, the increase of alternative bids to be considered in allocation and pricing leads to a higher complexity and thus exponentially increasing computational times. Table 3 shows the mean, standard deviations, and maximum of the solver runtimes for WD, VCG prices, SEP, and total computational time taken by the solver. In our experiments, we always calculate the optimal results, i.e., we do not provide a time limit for the solver. While the differences in the observed empirical complexity for $\lambda \in \{0.5, 1.0\}$ are not obvious, computational solver time substantially increases for $\lambda = 0.25$. Since solving the SEP may be required for several iterations and calculation of VCG prices requires to solve WD again without each winner, these two problems take the main portion of the observed requirements for computational time. Note that the EBPO solver times are not reported since they are close to zero for all simulation experiments performed.

	$\lambda = 0.25$			$\lambda = 0.5$			$\lambda = 1.0$		
	μ	σ	max	μ	σ	max	μ	σ	max
WD	4.63	14.05	97.51	0.66	1.50	9.30	0.54	1.52	9.29
VCG	22.81	62.31	397.17	2.94	5.81	46.85	2.15	5.82	46.74
SEP	24.07	80.02	621.33	2.29	4.61	27.62	2.12	5.26	34.27
total	51.52	150.25	932.90	5.88	11.35	79.89	4.80	12.02	79.71

Table 3. Solver runtimes (seconds)

The ratios of monetary bids, core, and VCG prices are shown in table 4. Figure 4 and table 5 show the seller revenue for different pricing rules. We have scaled the bids, using functions $s_j(\cdot)$ as described in section 5.1, to a maximal value of 3 monetary units per loading dock reservation to make monetary values comparable to those identified by the German Federal Office for Goods Transport in a survey (Bundesamt für Güterverkehr, 2011, p. 25). These results show that for a high bidder competition ($\lambda = 1$), the core prices are almost 50% above VCG prices in mean. Since for the same unique departure time for all bidders the probability of conflicts increases, the probability of forming a blocking coalition which increases the core price also rises. In contrast, $\lambda = 0.5$ resolves most of these conflicts – the bidders still bid on all alternative routes up to the expected round trip time threshold for every possible starting time – and therefore increases the VCG/core ratio. That is, the prices bidders actually have to pay and therewith the seller revenue, decreases. For a further increase in alternative bids, the complexity increases substantially and the mean waiting time contrarily decreases as mentioned above. However, differences in price ratios are not decreasing obviously anymore. The further alternatives increase the potential for a conflict free allocation for all bidders but losing bidders are unlikely to form a blocking coalition for $\lambda = 0.5$ already.

	$\lambda = 0.25$		$\lambda = 0.5$		$\lambda = 1.0$	
	μ	σ	μ	σ	μ	σ
core/bid	0.11	0.24	0.11	0.23	0.69	0.38
VCG/bid	0.09	0.21	0.09	0.21	0.51	0.45
VCG/core	0.81	0.32	0.81	0.33	0.67	0.42

Table 4. Ratios of bids, core, and VCG prices

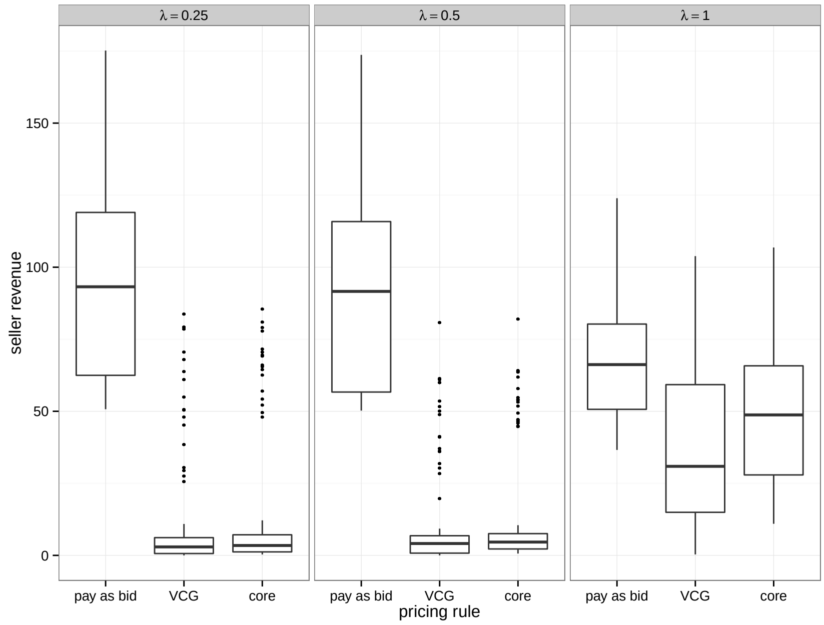


Figure 4. Seller revenue (monetary units)

	$\lambda = 0.25$		$\lambda = 0.5$		$\lambda = 1.0$	
	μ	σ	μ	σ	μ	σ
pay as bid	95.00	32.34	92.35	32.89	67.02	20.92
VCG	9.09	17.81	8.93	15.41	37.18	26.80
core	11.14	20.95	10.73	17.42	49.23	23.39

Table 5. Seller revenue (monetary units)

6 Discussion

Our evaluation has shown the efficacy and utility of the proposed artifact in a realistic logistics setting. Building on previous work, we have proposed a method to further decrease the mean waiting time by more than 80% in our simulations. This improvement, however, comes at the cost of computational complexity. While the differences in the observed empirical complexity for $\lambda \in \{0.5, 1.0\}$ are not obvious, computational solver time substantially increases for $\lambda = 0.25$, even for the rather small setting of ten bidders in our scenario.

While the observed absolute numbers remain feasible in practice, the computational complexity has to be analyzed in detail for practical applications. For example, it may be required to restrict valid bids to a reduced number of time slots since this parameter has an essential impact on the computational complexity. The application of core-selecting payment rules requires WD, SEP, and EBPO to be solved exactly. If this is not feasible in practice since further restrictions on valid bids cannot be applied, approximation of these problems may be required. Then, additional problems can arise that mitigate the acceptance of the approach by participants. For example, when approximating WD, a bidder might win the auction who would not have won in the efficient, exact solution (Goetzendorff et al., 2015). We leave an application of approximation algorithms to our setting for future work.

Note that our results rely on and are limited to the assumption of truthful bidders (and therewith efficiency of the core-selecting auction), which does not hold in general for core-selecting auctions (Day and Milgrom, 2008; Day and Raghavan, 2007). Even if acquiring information to select best responses is infeasible, further research is required on how bidders can benefit with simple deviation strategies. There is a trade-off between potentially higher revenue with fair prices and the possibilities to gain from deviation. For $\lambda = 1.0$, there is a much higher potential to gain from deviation (causing lower efficiency of the auction), though under the assumption of truthful bidding, it produces higher revenue and price fairness. In contrast, for $\lambda \in \{0.25, 0.5\}$ the differences between VCG and core outcomes is much smaller, and the low revenue problem that is addressed by core-payments does not occur to the same extend as for $\lambda = 1.0$. Depending on the application, it might therefore be advantageous to apply VCG-pricing in some settings with higher flexibility and therewith lower competition, though the perceived fairness of prices might be more relevant than full incentive compatibility in others.

We have limited the basic assumptions and simulation parameters in this research to those of our previous work (Lemke, Bichler, and Minner, 2014) to allow a direct comparison of the results. Therefore, the number of bidders and trucks of carriers remain small as compared to practical applications in freight networks. However, in practical applications it is more likely that multiple organizations utilize different time slot allocation schemes, i.e., the setting investigated in this research would then represent only a subset of warehouses, and multiple auctions would be conducted. In our experiments, we assume homogeneous valuations among the carriers regarding waiting and round trip times, which could be relaxed by different valuation models. Another limitation of our approach with regard to practice arises regarding the artificially generated networks. Although the utilized CATS algorithm produces networks with properties comparable to real-worlds transportation networks, demonstration of practical relevance can be improved by using existing networks in future research.

7 Conclusions

The contribution of this research is the application of core-selecting package auctions to address the loading dock congestion problem. We have proposed a bidding language and a core-selecting package auction for this setting based on existing literature. We have provided an analysis of the potential for waiting time reduction and of the empirical computational complexity for settings with varying characteristics.

We have evaluated the artifact by means of simulation. The evaluation has shown the efficacy and usefulness of the artifact. Building on previous work, we have shown that the proposed artifact can further decrease the mean waiting time substantially and that the core-pricing rule is well-suited to address the price fairness and low seller revenue problems in this setting.

References

Abeledo, H., R. Fukasawa, A. Pessoa, and E. Uchoa (2010). "The Time Dependent Traveling Salesman Problem: Polyhedra and Branch-Cut-and-Price Algorithm." In: *Experimental Algorithms*. Ed. by P. Festa. Vol. 6049. Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 202–213.

Ausubel, L. and P. Milgrom (2002). "Ascending Auctions with Package Bidding." *Frontiers of Theoretical Economics* 1, 1–42.

Bundesamt für Güterverkehr (2011). *Sonderbericht zur Situation an der Laderampe*. German. URL: `http://www.bag.bund.de/SharedDocs/Downloads/DE/Marktbeobachtung/Sonderberichte/SB_Rampe.pdf` (visited on 11/28/2014).

Bundesverband Güterkraftverkehr Logistik und Entsorgung e.V. (2013). *Jahresbericht 2012/2013*. German. URL: `http://www.bgl-ev.de/images/downloads/ueber/jahresbericht/bgl-jahresbericht.pdf` (visited on 11/28/2014).

Clarke, E. (1971). "Multipart Pricing of Public Goods." *Public Choice* 11, 17–33.

Dantzig, G., R. Fulkerson, and S. Johnson (1954). "Solution of a large-scale traveling-salesman problem." *Journal of the operations research society of America*, 393–410.

Day, R. and P. Milgrom (2008). "Core-selecting package auctions." *International Journal of Game Theory* 36 (3), 393–407 (3).

Day, R. and S. Raghavan (2007). "Fair payments for efficient allocations in public sector combinatorial auctions." *Management Science* 53 (9), 1389–1406 (9).

Day, R. and P. Cramton (2012). "Quadratic Core-Selecting Payment Rules for Combinatorial Auctions." *Operations Research* 60 (3), 588–603.

Dütting, P., V. Gkatzelis, and T. Roughgarden (2014). "The performance of deferred-acceptance auctions." In: *Proceedings of the Fifteenth ACM Conference on Economics and Computation*. Ed. by M. Babaioff, V. Conitzer, and D. Easley. EC '14. New York, NY, USA: ACM, pp. 187–204.

Fox, K. R. (1973). "Production scheduling on parallel lines with dependencies." PhD thesis. Baltimore: Johns Hopkins University.

Fox, K. R., B. Gavish, and S. C. Graves (1980). "An n-Constraint Formulation of the (Time-Dependent) Traveling Salesman Problem." *Operations Research* 28 (4), 1018–1021.

Goetzendorff, A., M. Bichler, B. Day, and P. Shabalin (2015). "Compact bid languages and core-pricing in large multi-item auctions." *Management Science*. forthcoming.

Gregor, S. and D. Jones (2007). "The Anatomy of a Design Theory." *Journal of the Association for Information Systems* 8 (5).

Gregor, S. and A. R. Hevner (2013). "Positioning and Presenting Design Science Research for Maximum Impact." *MIS Quarterly* 37 (2), 337–355.

Groves, T. (1973). "Incentives in Teams." *Econometrica* 41, 617–631.

Held, M. and R. M. Karp (1970). "The Traveling-Salesman Problem and Minimum Spanning Trees." *Operations Research* 18 (6), 1138–1162.

Hevner, A. R., S. T. March, J. Park, and S. Ram (2004). "Design Science in Information Systems Research." *MIS Quarterly* 28 (1), 75–105.

Ichoua, S., M. Gendreau, and J.-Y. Potvin (2003). "Vehicle dispatching with time-dependent travel times." *European Journal of Operational Research* 144 (2), 379–396.

Kok, A., E. Hans, and J. Schutten (2012). "Vehicle routing under time-dependent travel times: The impact of congestion avoidance." *Computers & Operations Research* 39 (5), 910–918.

Law, A. M. and D. M. Kelton (1999). *Simulation Modeling and Analysis*. 3rd. McGraw-Hill.

Lemke, J., M. Bichler, and S. Minner (2014). "Fighting Waiting Times at the Retail Warehouse: How Information Aids Coordination." In: *7th International Scientific Symposium on Logistics*. Cologne, Germany.

Leyton-Brown, K., M. Pearson, and Y. Shoham (2000). "Towards a universal test suite for combinatorial auction algorithms." In: *Proceedings of the 2nd ACM conference on Electronic commerce*. EC '00. Minneapolis, Minnesota, USA: ACM, pp. 66–76.

Milgrom, P. and I. Segal (2014a). *Deferred-Acceptance Auctions and Radio Spectrum Reallocation*. Working Paper. Stanford University.

— (2014b). "Deferred-acceptance Auctions and Radio Spectrum Reallocation." In: *Proceedings of the Fifteenth ACM Conference on Economics and Computation*. Ed. by M. Babaioff, V. Conitzer, and D. Easley. EC '14. New York, NY, USA: ACM, pp. 185–186.

Nisan, N., T. Roughgarden, E. Tardos, and V. V. Vazirani, eds. (2007). *Algorithmic Game Theory*. Cambridge University Press.

Picard, J.-C. and M. Queyranne (1978). "The Time-Dependent Traveling Salesman Problem and Its Application to the Tardiness Problem in One-Machine Scheduling." *Operations Research* 26 (1), 86–110.

Simon, H. A. (1996). *The Sciences of the Artificial*. 3rd. Cambridge, MA: MIT Press.

Vickrey, W. (1961). "Counterspeculation, Auctions, and Competitive Sealed Tenders." *Journal of Finance* 16 (1), 8–37.

A Algorithms

A.1 Core constraints generation

Let $W \subseteq I$ denote the set of winners, b^* the vector of winning bids, I_{-i} the set of bidders without bidder i, $p^{vcg} = (p_1^{vcg}, \ldots, p_{\mathscr{I}}^{vcg})$ the VCG payment vector, $p^\tau = (p_1^\tau, \ldots, p_{\mathscr{I}}^\tau)$ the payment vector, $p^{core,\tau} = (p_1^{core,\tau}, \ldots, p_{\mathscr{I}}^{core,\tau})$ the (temporary) core-payment vector, and $C^\tau \subseteq I$ the blocking coalition of bidders in iteration τ. Algorithm 1 shows how core-payments are calculated in pseudo code.

$W, b^* \leftarrow$ solve the Winner determination problem $\mathrm{WD}(I)$
foreach $i \in W$ **do**
    $p_i^{\mathrm{vcg}} \leftarrow$ compute the VCG price $b_i^* - \big(\mathrm{WD}(I) - \mathrm{WD}(I_{-i})\big)$
$p^1 \leftarrow p^{\mathrm{vcg}}$
$\theta^0(\varepsilon) \leftarrow \sum_i p_i^{\mathrm{vcg}}$
$\tau \leftarrow 1$
while *true* **do**
    $C^\tau \leftarrow$ solve the separation problem SEP^τ
    if $z(p^\tau) > \theta^{\tau-1}(\varepsilon)$ **then**
        add constraint $\sum_{i \in W \setminus C^\tau} p_i^{core,\tau} \geq z(p^\tau) - \sum_{i \in W \cap C^\tau} p_i^\tau$ to EBPO^τ and solve
        $p^{\tau+1} \leftarrow p^{core,\tau}$ from EBPO^τ
    else
        $p \leftarrow p^\tau$
        break
    $\tau \leftarrow \tau + 1$

Algorithm 1: Core constraints generation (following Day and Raghavan (2007, p. 1398))

A.2 Departure time calculation

Algorithm 2 provides pseudo code on how the departure time is determined.

$minDepartureTime \leftarrow 0$
$minRoundTripTime \leftarrow \infty$
for $d \leftarrow 0$ **to** *endOfDay* **do**
  $rtt \leftarrow expectedRtt(d, route, loadInformation)$
  if $rtt < minRoundTripTime$ *and* $d + rtt < endOfDay$ **then**
    $minDepartureTime \leftarrow d$

Algorithm 2: Finding the departure time for a given route with minimum expected waiting time.

A.3 Expected round trip time calculation

Algorithm 3 shows the pseudo code for calculating the expected round trip time for a given route and departure time. The route is written as a list of locations $\{l_0, l_1, \ldots, l_n\}$ with $l_0 = l_n$ being the depot of the carrier. The function $travelTime : L \times L \rightarrow \mathbb{N}_0$ returns the time needed to travel from one location to another. The historical information is given by average waiting times $waitingTime(k, o)$ at warehouse k in time slot o, which can be computed for time t by $slot(t)$, determining the resolution of the time planning using historic waiting time information. The service time at warehouse k is indicated by $serviceTime(k)$.

$t \leftarrow departure$
for $o \leftarrow 1$ **to** $n - 1$ **do**
  $t \leftarrow t + travelTime(l_{o-1}, l_o)$
  $t \leftarrow t + waitingTime(l_o, slot(t))$
  $t \leftarrow t + serviceTime(l_o)$
$t \leftarrow t + travelTime(l_{n-1}, l_n)$
$expectedRtt = t - departure$

Algorithm 3: Calculating the expected round trip time for a given route and departure time considering waiting time information.