# Non-monetary coordination mechanisms for time slot allocation in warehouse delivery

Paul Karaenke[a], Martin Bichler[a], Soeren Merting[a], Stefan Minner[b]

[a] *Department of Informatics*
*Technical University of Munich*
*85748 Garching, Germany*
[b] *School of Management*
*Technical University of Munich*
*80333 Munich, Germany*

---

**Abstract**

Recent empirical evidence suggests that delivery to retail warehouses suffers from a lack of coordination. While carriers try to optimize their routes, they often experience very long waiting times at loading docks, which renders their individual planning useless. To reduce such inefficiencies, carriers need to coordinate. This problem has received considerable attention in practice, but the design of coordination mechanisms is challenging for several reasons: First, the underlying package assignment problem is NP-hard. Second, efficiency, incentive-compatibility, and fairness are important design desiderata, but in most economic environments they are conflicting. Third, the market for logistics services is competitive and price-based mechanisms where carriers might have to pay for time slots suffer from low acceptance. We draw on recent advances in market design, more specifically randomized matching mechanisms, which set incentives for carriers to share information truthfully such that a central entity can coordinate their plans in a fair and approximately efficient way. We use and adapt the existing maximizing cardinal utilities (MAXCU) framework to a retail logistics problem, which yields a new and powerful approach for coordination. We report numerical experiments based on field data from a real-world logistics network to analyze the average reduction in waiting times and the computation times required and compare to first-come, first-served and an auction mechanism. Our results show that randomized matching mechanisms provide an effective means to reduce waiting times at warehouses without requiring monetary transfers by the carriers. They run in polynomial time and provide a practical solution to wide-spread coordination problems.

*Preprint submitted to European Journal of Operational Research* *April 14, 2020*

*Keywords:* Transportation, electronic markets and matching, coordination

## 1. Introduction

We focus on the loading dock problem in retail logistics that has received much attention by practitioners in the recent years. According to a survey among more than 500 transportation companies in Germany, 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). Such waiting times are a significant problem for carriers and warehouse operators. A recent study among 778 truck drivers by the German Federal Office for Transportation reports that the waiting times even increased in the past years (Bundesamt für Güterverkehr, 2018). In another study, the German Federal Office for Transportation describes the uncoordinated arrivals of trucks as the main reasons for waiting times (Bundesamt für Güterverkehr, 2011). Adding capacity with additional loading docks at warehouse sites requires substantial investments and can also be infeasible in urban areas.

Overall, the lack of coordination causes substantial inefficiencies in retail transportation logistics. The carriers decentrally solve vehicle routing problems and compute optimal routes, but they do this in an uncoordinated manner. The warehouses face capacity planning problems for their loading docks because of the random carrier arrival. If all information about supplier preferences for different routes and warehouse capacities was available, then a central clearing house (potentially organized via a booking platform) could select routes and allocate time slots to carriers such that waiting times are minimized. However, coordination mechanisms to elicit this information from carriers are challenging to design as we will discuss below.

Some retailers use simple first-come, first-served (FCFS) time slot management systems and charge a fixed price for each slot. However, the adoption is low as margins for carriers are low and they are not willing to pay for reservations. We learned in discussions, that charging payments for loading docks reservations is very unpopular for carriers. Moreover, the simple FCFS mechanism collects only little information about carrier preferences (a single package of time slots on a route) such that one cannot expect an efficient allocation of the available capacities at the warehouses. Depending on the random permutation of carriers arriving, the allocation can be arbitrarily bad and there will be many ways to Pareto-improve. In most cases, however, there is not even an FCFS mechanism in place that would alleviate the long waiting times that arise (Bundesverband Güterkraftverkehr Logistik und Entsorgung e.V., 2013). In a previous paper, we analyze various auction mechanisms (Karaenke et al., 2019), but the potentially high payments of carriers for the reservations in auctions again constitute a significant barrier to adoption in the field. So, the question we ask in this paper is, whether efficient coordination can also be facilitated without payments by the carriers.

### *1.1. Matching Mechanisms*

In this paper, we explore new possibilities for the coordination without monetary payments for the allocation of time slots to carriers. Three economic properties of a coordination mechanism are desirable: (allocative) efficiency, incentives for truthful revelation of the carriers' preferences, and envy-freeness.

- *Allocative efficiency* refers to the maximization of the sum of the carriers' utilities. This describes a particular objective function that is widely accepted in economics and also referred to as welfare maximization.

- A mechanism is *incentive-compatible*, if participants reveal their preferences truthfully in equilibrium. Incentives for truthful reporting of preferences are important as strategic manipulation leads to high participation costs, allocative inefficiencies, and it would jeopardize the sustainability of a mechanism in the long run. Ideally, participants should have a dominant equilibrium strategy to reveal their preferences, i.e., their cost savings for a route without waiting times, truthfully. In this case we say that the mechanism is *strategy-proof.*

- *Envy-freeness* means that every carrier prefers his allocation (of time slots) to that of any other carrier. Envy-freeness can be seen as a strong form of *fairness.* Envy-freeness takes into account the preferences of all participants and ensures that none of the agents has justified envy and would want to change his allocation with another participant. This yields also stability of an allocation. Fairness is also considered central in other transportation problems (Lu and Quadrifoglio, 2019; Zhu and Ukkusuri, 2017), and has received a lot of recent attention (Kleinberg et al., 2018) in the context of algorithm design and automated decision making.

Achieving such properties without allowing for monetary payments of the participants and market prices is challenging. However, new developments in the theory of matching with preferences provide approaches to coordination problems. Matching with preferences (but without money) has become a popular subject after the Nobel Prize in Economic Sciences was awarded jointly to Alvin E. Roth and Lloyd S. Shapley by the Swedish Academy of Sciences for the theory of stable allocations and the practice of market design in 2012. Most research assumes simple preferences with unit demand as in school choice or the well-known stable marriage problem (Gale and Shapley, 1962; Diebold and Bichler, 2017; Delorme et al., 2019). Matching problems with preferences for bundles (or packages) of objects have been addressed only recently and are sometimes referred to as *combinatorial assignment problems* (Budish, 2011). The coordination problem in retail logistics is an excellent example: We need to assign bundles of time slots at warehouse loading docks that constitute a tour for a carrier. Unfortunately, *incentive-compatibility*, *allocative efficiency*, and *envy-freeness* are already *conflicting* if one is restricted to agents with unit demand rather than preferences for bundles and deterministic mechanisms (Roth, 1982; Abdulkadiroğlu and Sönmez, 2003).

Randomization potentially allows to circumvent these negative results. Nguyen et al. (2016) proposed the maximizing cardinal utilities (MAXCU) framework for randomized matching mechanisms. They showed that mechanisms in this framework are allocatively efficient, envy-free, and *asymptotically* strategy-proof. This is at the expense of feasibility, and some of the supply constraints on warehouse capacity might be violated. However, these constraint violations are small and limited by the level of complementarity in the preferences, i.e., the size of the packages. In the context of our problem, this means that more carriers are assigned to some loading docks in some time slots, but the level of overbooking is small as we show and can easily be accommodated by warehouse providers. Apart from economic aspects, one of the most important properties of the framework is the computational complexity of the mechanism, however. While the deterministic problem of optimally allocating packages of time slots is NP-hard, the randomized mechanism can be computed in polynomial time. This is of central importance for our problem where problems tend to be large.

While there is some literature on auction mechanisms used in transportation and logistics (Elmaghraby and Keskinocak, 2004; Caplice, 2007; Agrali et al., 2008; Huang and Xu, 2013; Xu and Huang, 2014; Triki et al., 2014; Gansterer et al., 2019) mechanisms without money have received little attention in this literature. MAXCU provides a general framework, but it has never been applied to logistics coordination problems and therefore requires context-specific adjustment and empirical testing.

### *1.2. Contributions*

There are multiple warehouses operated by one or more companies (e.g., retailers). On behalf of the companies' suppliers, carriers need to visit several warehouses and deliver goods. They have preferences for time slots to unload goods based on their routing. Carriers compute tours to visit all warehouses and they are interested in a package of time slots at different warehouses such that they can unload their goods without waiting times on their tour. Similar to the FCFS time slot management systems currently in use, an intermediary provides a platform for coordinating the carriers. Nowadays, this role is assumed by large retailers or third-parties specializing in time-slot management in logistics. Instead of a FCFS time slot management system with fixed reservation prices that is used nowadays, we aim for an economic mechanism in which carriers have simple strategies to reveal their opportunity costs for tours and respective time slots truthfully. This coordination should not lead to extra costs for the carriers.

We introduce a randomized matching mechanism for the assignment of time slots to trucks of a carrier, which draws on the general framework proposed by Nguyen et al. (2016). This randomized mechanism does not require payments by the carriers, but provides incentives for truthful revelation of preferences for the carriers and it yields an approximately efficient and envy-free outcome in expectation. The framework is quite general, but needs to be adapted for a particular allocation problem. More importantly, theory is silent about the efficiency gains (i.e., the waiting time reductions) one can expect for specific problems compared to no coordination, and how severe the capacity violations due to randomized mechanisms are on average. We provide an extensive set of numerical experiments based on data from a retail transportation network in the field to study the efficiency gains one can expect. In particular, we estimate these efficiency gains compared to situations without coordination as well as core-selecting auctions with near-optimal solutions.

We find that the randomized mechanism is very fast to compute, while the optimal deterministic solution typically cannot be computed for realistic problem sizes. Even large problem sizes can be computed in a few seconds or minutes using the randomized mechanism. The violations of the capacity constraints at warehouses are low such that the mechanism can be seen as a viable approach to combat the lack of coordination in retail logistics, while still taking into account preferences of carriers for various routes.

We compared the waiting time reductions to an auction mechanism. Given the hardness of the allocation problem, it is not even clear, which type of auction mechanism could be used. More importantly, the VCG mechanism is not incentive compatible anymore, if the allocation problem cannot be computed exactly (Leyton-Brown et al., 2006). We implemented a combinatorial auction with a core-selecting payment rule based on near-optimal solutions to the allocation problem (Goetzendorff et al., 2015). Interestingly, the waiting time reductions we got from the randomized matching mechanism were close to the ones we achieved with the auction mechanism, although the matching mechanism does not lead to additional costs or monetary transfers for the carriers.

The remainder of this paper is structured as follows. In Section 2, we present the matching mechanisms along with basic assumptions of the model. We present the experimental design and results in Section 3 and discuss our findings, implications for practice, as well as limitations and future research in Section 4.

## 2. Coordination via Matching Mechanisms

We describe the design problem by first introducing the basic assumptions and the fundamental allocation problem if complete information about the carriers' preferences was available. Then we introduce the matching mechanism that satisfies a number of desirable economic properties, and a core-selecting auction mechanism that we compare against.

### *2.1. Economic Environment*

A *matching* or *assignment* describes the allocation of time slots at warehouse loading docks to carriers. Preferences are provided for routes, i.e. bids on packages (bundles) of time slots at different warehouses. Carriers constitute the *agents* in the mechanism. Carriers are allowed to submit preferences on alternative routes on a daily basis, the *package preferences* on bundles of time slots. The value for a shorter route is proportional to the time saved compared to long routes and the time freed up for truck drivers considering the usual waiting times. We assume that a *coordinator* is organizing the matching market on behalf of the warehouses. This coordinator might just provide an information system, which computes allocations once per day for allocations on the next day upon preferences entered by the carriers.

### *2.2. The Time Slot Allocation Problem in Retail Logistics*

We consider a time slot allocation problem with $\mathcal{K}$ warehouses, $\mathcal{I}$ carriers (agents), and $\mathcal{T}$ intra-day time slots. The locations of warehouses and carriers are given within the transportation network with known (average) travel distances and travel times. The service capacity of warehouses (loading docks) is modeled as a multidimensional knapsack problem. In each time slot $t \in T = \{1, 2, \dots, \mathcal{T}\}$, each warehouse $k \in K = \{1, 2, \dots, \mathcal{K}\}$ has a capacity of $c_k = (c_{k1}, \dots, c_{k\mathcal{T}})$. That is, warehouse $k$ can service up to $c_{kt}$ trucks in time slot $t$. We call a pair $o = (k, t)$ of a warehouse and a time slot an *object* with capacity $c(o)$, and define $\mathcal{O}$ as the set of all objects.

Carriers have to deliver freight to a warehouse, pick it up there, or both. We assume that each *carrier* has a truck with sufficient capacity to fulfill the orders. The truck starts at the depot and returns to the depot again after (un)loading its freight at the retailers' warehouses. Within the reserved time slots a carrier can (un)load his freight. In our simulations, we compute a route to visit all warehouses on the tour solving a TSP for each truck independently. Note that every individual truck that can be processed without long waiting times leads to time savings and is beneficial for carriers, i.e., the savings per tour matter primarily. Thus, we consider trucks individually, which also keeps the experimental design simpler. In a design with multiple trucks per carrier we need a number of additional assumptions and treatment variables (e.g., numbers of trucks, costs for using one or multiple trucks). In trial experiments with multiple trucks, we did not see changes in the ranking of coordination formats. Hence, we only report the setting with one truck per carrier. Whether the routes (package bids) are based on the results of a VRP or TSP is not important for the evaluation of waiting times. We only need reasonable tours as input.

The carriers $i \in I = \{1, 2, \ldots, \mathcal{I}\}$ have valuations (cardinal preferences) $v_{iS}$ for bundles $S \in \{0,1\}^{|\mathcal{O}|}$ of objects represented as vectors where $S_o = 1$ if object $o$ is in the bundle. We define the size of a bundle $S$ as the number of nonzero entries in $S$, $size(S) = \sum_{o \in \mathcal{O}} \mathbf{1}_{\{S_o > 0\}}$. A bundle $S$ encodes the sequence of visited warehouses and the respective time slots. Carriers are allowed to submit preferences for as many bundles they want, i.e., they can express preferences for alternative routes and corresponding time slots.

Let $x_{iS}$ denote binary decision variables indicating whether carrier $i$ gets bundle $S$ or not. The winner determination problem (WDP) of the coordinator can be formulated as follows.

$$\begin{aligned}
w(I) = \max \sum_{i,S} v_{iS} x_{iS} & & \text{(WDP)}\\
\text{s.t.} \sum_{i,S: o \in S} x_{iS} \leq c(o) & \quad \forall o \in \mathcal{O} & \text{(supply)}\\
\sum_{S} x_{iS} \leq 1 & \quad \forall i \in I & \text{(demand)}\\
x_{iS} \in \{0,1\} & \quad \forall i \in I, S \in \{0,1\}^{|\mathcal{O}|} & \text{(binary)}
\end{aligned}$$

The objective is to maximize the sum of valuations of the accepted bundles in WDP, i.e., to maximize the social welfare. The (supply) constraint ensures that the warehouse capacities are not exceeded for allocated bundles for each time slot. Constraint (demand) ensures that each carrier wins at most one bundle. Carriers who won one of their submitted tours have reservations for the respective time slots at the loading docks, while losing carriers have to queue for service with lower priority. This is a weighted set packing problem, which is known to be NP-hard (Garey and Johnson, 1979).

### *2.3. The Randomized Matching Mechanism*

Let us now describe a randomized matching mechanism that solves WDP and yields an approximately efficient and envy-free solution in polynomial time. The mechanism is based on the MAXCU (maximizing cardinal utilities) framework introduced by Nguyen et al. (2016). MAXCU is a general approach for solving allocation problems with cardinal preferences over bundles of objects without monetary transfers. First, MAXCU computes an optimal (fractional) solution for the LP-relaxation of WDP with additional envy-constraints. In a second step, MAXCU creates a lottery over approximately feasible integral solutions of WDP, such that the *expected* solution of this lottery equals the fractional solution. As a consequence, also the economic properties envy-freeness and efficiency are in expectation. Approximate feasibility refers to integer solutions that might in some instances violate supply constraints as we will discuss next.

Let us discuss the implementation of this mechanism in the context of our retail logistics domain. To achieve *envy-freeness in expectation* in MAXCU, we need to introduce additional constraints for the computation of the fractional optimum of WDP. An agent $i \in I$ envies another agent $j \in I$ if $i$ prefers the assignment of $j$ over his own assignment. We formalize this as a linear inequality: agent $i$ envies agent $j$ iff

$$\sum_S v_{iS} x_{iS} < \sum_S v_{iS} x_{jS}. \tag{envy}$$

With this we can introduce the no-envy-constraint for every pair of agents $i, j \in I,\ i \neq j$:

$$\sum_S v_{iS}(x_{iS} - x_{jS}) \geq 0 \quad \forall i, j \in I. \tag{no-envy}$$

The first step of MAXCU is to solve the problem

$$x^* = argmax \left\{ \sum_{i,S} v_{iS} x_{iS} \mid supply, demand, no-envy, x_{iS} \in [0,1] \right\}.$$

Since every variable $x_{iS}$ is in $[0,1]$ and the sum over all variables referring to the same agent is not greater than 1, we can interpret the single variables as probabilities and the fractional solution $x^*$ as a random matching. That is, for every agent we do not have an assignment, we only have a probability distribution over assignments. Next, we have to decompose this random matching into a lottery over deterministic matchings.

### 2.3.1. The Lottery Algorithm

Unfortunately, in general it is not possible to decompose $x^*$ into a lottery of integral solutions satisfying (*supply*) and (*demand*) if the bundle size $size(S)$ is larger than one. To circumvent this, one can either scale $x^*$ by a factor $\alpha \in (0,1)$ such that the decomposition becomes possible, or one allows for the relaxation of some constraints. Here, we use the second approach. It is possible to decompose $x^*$ into a lottery of integral solutions satisfying (*demand*) and (*relaxed supply*), where

$$\sum_{i,S:o\in S} x_{iS} \leq c(o) + L - 1 \qquad \forall o \in \mathcal{O}. \qquad \text{(relaxed supply)}$$

Here, $L = \max\{size(S) \mid x^*_{i,S} > 0\}$ is the size of the largest bundle that has a nonzero entry in the solution to (WDP). For creating such a lottery, Nguyen et al. (2016) propose Algorithm 1. Let $dim$ denote the dimension of (WDP). With this polynomial time *lottery algorithm*, we find at most $dim + 1$ integral points, the convex hull of which is arbitrarily close to the fractional solution $x^*$. The algorithm then returns a lottery over these $dim + 1$ integral solutions, which is close to $x^*$ in expectation. In this lottery algorithm, we use a subroutine to return an integer point $\bar{x}$ such that $u^t\bar{x} \geq u^t x^*$, while $u$ is arbitrary. This subroutine is called *iterative rounding algorithm* (IRA) described in the next subsection.

**Algorithm 1:** Lottery algorithm.

1. Set $\mathcal{S} = \{\text{IRA}(x^*)\}$, i.e., find an integer solution via IRA.
2. $y = argmin\{|x^* - y| \mid y \in conv(\mathcal{S})\}$; **if**$(|x^* - y| < \varepsilon)$ END
3. Choose $\mathcal{S}' \subseteq \mathcal{S}$ of size $|\mathcal{S}'| \leq dim$, $y \in conv(\mathcal{S}')$ : $z = x^* + \delta\frac{x^*-y}{|x^*-y|}$
4. Find integral $z'$ s.t. (Demand),(relaxed Supply) and $(Z) : (x^* - y)^t z' \geq (x^* - y)^t z$ via IRA
5. $\mathcal{S} = \mathcal{S}' \cup \{z'\}$ and goto 2.

**Output:** Convex combination of final $y$

Figure 1 shows a graphical representation of one algorithm iteration. The algorithm tries to get $x^*$ covered by the convex hull of $\mathcal{S}$ ($conv(\mathcal{S})$). In each iteration the algorithm decreases the distance between $y$ and $x^*$ by adding a new integral solution to the solution set $\mathcal{S}$ and terminates when the distance between $y$ and $x^*$ is smaller than $\varepsilon$. That is, we consider $y$ as a good

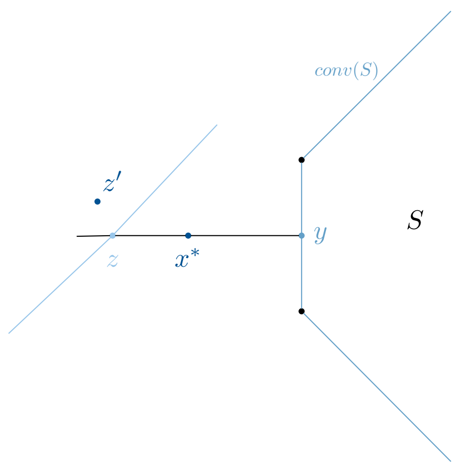


Figure 1: Graphical representation of one iteration of the lottery algorithm.

approximation for $x^*$ and return the support of $y$. All solutions in $\mathcal{S}$ that are not part of the support of $y$, calculated in the quadratic optimization problem (QOP) in step 2, are deleted (step 3). Thus, although we add a new integral solution to $\mathcal{S}$ in each iteration, the size of $\mathcal{S}$ never grows above $dim + 1$, since as long as $y \neq x^*$, $y$ always has to be on a face of $conv(\mathcal{S})$. Hence, the support of $y$ consists of at most $dim$ solutions. Step 4 ensures that we search in the right direction for new integral solutions. As a side product, the QOP also calculates the coefficients $\lambda^{(j)}$ for the convex combination and we have $x^* \approx y = \sum_{j=1}^{|\mathcal{S}|} \lambda^{(j)} x^{(j)}$, for $x^{(j)} \in \mathcal{S}$.

We still have to show how we receive the integral solution satisfying (*demand*) and (*relaxed supply*) from a given fractional solution and a given additional constraint ($Z$) in step 4 of Algorithm 1. We use the iterative rounding algorithm $IRA(x, Z)$ with a (fractional) point $x$ and an additional constraint $Z$ as input.

In step 1a of Algorithm 2, we fix components of $x$ to 0 or 1 if they already have this value, that is, we exclude these components from further optimization steps and consider them as constants. The rounding happens indirectly in step 1b and 2 due to reoptimizing the WDP after deleting constraints. To detect such a constraint, we look at the worst case in each iteration: If we would round up all currently fractional components, would we still fulfill the relaxed supply constraint? If this is the case, we can delete this supply constraint in step 1b, since it cannot happen that we violate its relaxed version in the remaining iterations.

**Algorithm 2:** Iterative rounding algorithm.

1a. Delete all $x_i = 0$, $x_i = 1$, update the constraints and go to 1b.

1b. If there is no $x_i \in \{1, 0\}$, one can find at least one supply-constraint with:
$$\sum_{i \in I} \sum_{S: o \in S} \lceil x_{iS} \rceil \leq c(o) + L - 1.$$
Delete those constraints and goto 2.

2. Solve WDP with $(Z)$; **if**(all $x_i \in \{0, 1\}$) return $x$; **else** goto 1a.

### *2.3.2. Economic Properties*

In the MAXCU mechanism, the LP-relaxation with no envy constraints is solved and the fractional solution is decomposed into a lottery over integral solutions using Algorithm 1 and 2. This yields an efficient outcome in expectation that is *envy-free in expectation* and violates the supply constraints only by at most $L - 1$ (Nguyen et al., 2016). The mechanism is also *asymptotically strategy-proof.* A mechanism is $\epsilon$-strategy-proof if it is an "almost" dominant strategy to report truthfully given any vector of reports by the other carriers. That is, a carrier can gain at most $\epsilon$ by reporting untruthfully. Intuitively, a mechanism is asymptotically strategy-proof if for any $\epsilon > 0$ the mechanism is $\epsilon$-strategy-proof when the number of participants is large enough. So, for large markets truth-telling is almost a dominant strategy (up to $\epsilon$).

## *2.4. Core-Selecting Combinatorial Auctions*

In our experimental analysis, we want to understand the waiting time savings to be expected from of a randomized matching mechanism and compare with an auction mechanism. Payments are a significant barrier to the adoption of a mechanism in this domain, but if an auction mechanism was much more efficient, then this might be justified. Much as in single-item auctions, one can use implement a Vickrey–Clarke–Groves (VCG) mechanism, which is sometimes referred to as a *generalized Vickrey auction.* This auction format exhibits a dominant-strategy equilibrium, but it faces a few problems which do not appear in single-item auctions. Most notably, the outcome of a VCG auction might not be in the core, and losing bidders could make themselves better off together with the auctioneer. Vickrey–

Clarke–Groves solutions outside the core are often seen as undesirable. Core-selecting payment rules avoid such outcomes. While such payment rules are not strategy-proof, it has been argued that, with the uncertainties in large multi-object markets, bidders have sufficient incentives to bid truthfully (Azevedo and Budish, 2019). In particular, one can try to find a vector of core prices which is closest to the VCG payments. This means that such payments are minimal for the bidders, i.e., bidder-optimal. The idea is that bidder-optimal core (BOC) payments minimize the incentive to deviate from truthful bidding (Day and Cramton, 2012). The computation of such core payments is non-trivial. One could add a constraint for each possible losing coalition of carriers to the optimization problem which minimizes payments. The payments must not be less than what a losing coalition is willing to pay. However, the number of constraints grows exponentially with the number of bidders. However, such core constraints can be generated dynamically, which leads to an effective computation of such payments; this has also been used in spectrum auctions. In the computational approach discussed in the literature (Day and Raghavan, 2007) core prices are found by iteratively creating new price vectors $p^t$ and then checking at each iteration $t$ whether there is an alternative outcome which generates strictly more revenue for the seller and which every bidder in this new outcome weakly prefers to the current outcome. This approach has been adopted for spectrum auctions world-wide (Bichler and Goeree, 2017). Non-core outcomes are one issue with the VCG mechanism. More importantly, however, if the allocation problem cannot be computed exactly, as is the case in our logistics problem, then the VCG mechanism is not incentive-compatible anymore and it can lead to payments higher than the stated bids (Bichler, 2017). In our experiments, we leverage a relatively new approach to compute core-selecting payments with near-optimal solutions to the allocation problem (Goetzendorff et al., 2015). This approach avoids negative payments of payments that are higher than the stated bids. We refer the interested reader to related literature for details (Bichler, 2017).

## 3. Experiments

We use experiments to analyze the average case solution quality of our mechanism. We first describe the treatments of our numerical experiments, our simulation system, and the parameters used. Then we present the results including the reductions in waiting times, computation times, the numbers of reported preferences and winning carriers, and the constraint violations resulting from the MAXCU mechanism. Note that real-world applications can differ in various parameters, but we aim for a representative model that provides an estimate of the efficiency gains one can expect. In particular, we want to understand whether we can expect significant reductions in the waiting time with a randomized matching mechanism, and how these savings compare to those achieved with FCFS and a core-selecting auction.

### *3.1. Transportation Network*

We draw on data about the distribution network of a German retailer. The transportation network also provides the distances and respective travel times between the locations and is representative for networks in many other metropolitan areas. Overall, we have 65 locations of retail warehouses with distances and average travel times between all the sites in an adjacency matrix.

The locations of 10 warehouses and the depots of the carriers are determined randomly in each simulation by drawing from the set of 65 locations. Each carrier or truck has 4, 5, or 6 warehouses that he has to visit on a tour. The set of warehouses on a tour is randomly selected from the set of 10 warehouses by sampling without replacement.

### *3.2. Carriers*

We assume that carriers do their own planning independently and then start processing their plans every morning. While we do not consider due dates in our model, goods in retail logistics need to be with the warehouses until a certain time of the day. If carriers start later in the day, this also jeopardizes their due dates in case of congestion or other unforeseen events. Each carrier determines all potential routes without waiting time through the warehouses and reports preferences (i.e., package bids) on those that do not take longer than 110% of the optimal round trip time (RTT). This has several reasons: First, if carriers already have a detour of 10% then they could also go with the shortest route and accept waiting times of that size. Second, with this cutoff carriers submit approximately 5-8 tours on average (see Table 6). We cannot expect carriers to submit hundreds of bids and want to understand the waiting time reduction of the mechanisms with a reasonable number of bids submitted.

After the routes for each carrier have been determined, they compute a valuation (cardinal preference) for each route, which is a valuation for a package in the allocation. The valuation ($v_{iS}$) is proportional to the time saved per day. We compute the difference between the full day and the time needed for the optimal route. For example, if there are 8 hours on a working day and the optimal route takes 6 hours without waiting times, then we use a valuation of 120 minutes for reservations on this route. For an alternative route that takes 7 hours, we use a valuation of 60 minutes, since the time saved by this route is half of the optimal saving.

We compute the bid prices for the auction treatments similarly; the bids are proportional to the time saved per day. We assume a willingness to pay of 3 monetary units per warehouse reservation for the ideal route. This number for the willingness to pay is based on empirical observations (Bundesamt für Güterverkehr, 2011, p. 25). If the time savings are less than for the ideal route, a carrier is also willing to pay less per reservation. For example, for an alternative route that takes 7 hours (i.e., half of the time savings in the optimal route), the willingness-to-pay is only 1.5 monetary units per reservation.

Of course, carriers might determine valuations differently in the field, but we do not attach meaning to the absolute numbers, only to the relative differences of the mechanisms analyzed.

We assume that in all mechanisms, carriers report their opportunity costs to the mechanism and the mechanism then determines an assignment of carriers to time slots on a given day. That is, carriers base their reported preferences on estimated time savings, because they can use their resources for further tasks if waiting time is reduced.

### *3.3. Simulation System*

Our numerical experiments are conducted as a deterministic discrete event simulation with six types of events. In each simulation, the location of the warehouses and carriers is selected randomly. For the purpose of our simulation we assume one carrier has one truck. In the field, a carrier has multiple trucks. However, the waiting time savings can be evaluated on the level of single truck such that the assignment of multiple trucks to a single carrier would not lead to qualitatively different results.

The first event is the *departure event* which takes place when a carrier leaves his depot. It results in a *travel event* describing the carrier's travel from the current location to the 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 be loaded or unloaded respectively. Carriers who won one of their routes and respective reservations are serviced with higher priority at loading docks if they arrive during the time slots, while losing carriers have to queue for service with lower priority (FCFS among carriers without reservations). Thus, 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 time when the reservation is valid. If the truck arrives early or does not have a reservation, the wait event is triggered. Travel times are estimated from the field data. While this simplifies the environment compared to the field, it makes the comparison between the relative solution quality of the mechanisms easier.

The *service event* is triggered by the warehouse, whenever the loading/unloading process is started for a carrier. Having completed the service process, a carrier drives to the next warehouse or his depot if he already completed his tour. The simulation ends when every carrier has reached his depot again.

*3.4. Treatment Variables and Parameters*

The main treatment variable is the coordination mechanism used. In our baseline treatment without coordination ("no coord."), there is no information available to the carriers. In addition, we analyze the results of the MAXCU mechanism described earlier and, where possible, the results of an optimal allocation (OPT) assuming complete information as well as near-optimal allocations computed up to a 20% duality gap (Core0.2). It turns out, that this precision can be achieved quickly, even for large problem sizes. These allocations are used in the core-selecting auctions.

We also compare MAXCU to a simulation of an FCFS system where bidders start making reservations for their favorite route. It can happen that later warehouses on the route are already booked in the meantime, such that the carrier cannot make reservations for the entire route. That is, these carriers may be able to receive reservations for the first parts of their routes only. Therefore, we select a priority order of carriers 100 times per treatment uniform at random and reserve carriers' (partial) routes following a uniform distribution between one reservation and the whole routes' length. This provides a reasonable comparison with real-world practices beyond the case with no coordination.

The values for each parameter and treatment variable are summarized in Table 1. Based on information about problem sizes, we assume different numbers of trucks ($|I| \in \{50, 100, 200, 300, 400, 500\}$) in the experiments. A time slot in our simulation is 15 minutes and we consider 60 time slots per day ($|T|$). The warehouse capacity ranges from 1 to 15 loading docks depending on the number of trucks. In settings with a lower number of trucks, we also assume lower warehouse capacities to get an environment with significant waiting times as they can be observed in the field (warehouse capacities for the different numbers of trucks are are given in Table 2 in section 3.5).

The unloading time is drawn from a distribution based on data about reservations and unloading times from a time slot management system. We assume a "typical" service time of 30 minutes per warehouse and travel times based the historical field data available. We use expected service and travel times for route planning and valuations, but vary these times in the event simulation described. We assume service times and travel times to be normally distributed with a standard deviation of 7.5 minutes for service times and 10% for travel times (see Jenelius and Koutsopoulos (2013) for travel time distributions). The Normal distributions are truncated to $\pm 25\%$ of the mean, in order to avoid extremely short or long service and travel times. We randomly draw locations for warehouses and carriers 5 times each to create different simulation scenarios for each treatment. Subsequently, we draw travel and service times for these scenarios 5 times each resulting in 25 simulations for each treatment.

Apart from this experimental design, we performed sensitivity analyses on our results. To investigate the effect of the distributional assumptions, we conducted additional experiments with a uniform distribution, and we ran the experiments for 100 and 400 carriers each with 8 and 12 warehouses. This explores low- and high-supply scenarios. The additional treatments did not change the overall ranking of mechanisms and the differences in waiting and computational times as well as constraint violations are robust. The results of the experiments for these additional treatments are provided in Appendix  A; detailed results per treatment combination are provided in

| | name | values |
|---|---|---|
| Parameters | number of locations | 65 |
| | number of warehouses ($\|K\|$) | 10 |
| | travel time | distribution from field data |
| | unloading time | distribution from field data |
| | time slot length | 15 min |
| | number of time slots per day ($\|T\|$) | 60 |
| | planning horizon | 900 min (15 min · 60 slots) |
| | number of simulations / treatment | 25 |
| Treatment variables | warehouse capacity | $[1, 15]$ (depending on the number of trucks) |
| | warehouses per truck | $\{4, 5, 6\}$ |
| | number of trucks ($\|I\|$) | $\{50, 100, 200, 300, 400, 500\}$ |
| | matching mechanisms | {no coord., FCFS, MAXCU, OPT, Core0.2} |

Table 1: Parameters and treatment variables

Appendix C (supplementary material).

The simulation was implemented in the Java programming language and the commercial mathematical programming solver Gurobi Optimizer v6.5 was used for all optimization problems. Experiments were executed on computing nodes with two 14-core CPUs (Intel Xeon E5-2697 v3) and 64GB of memory (RAM) each.

### *3.5. Results*

In what follows, we report the savings in the waiting times of trucks due to the use of a coordination mechanisms and the computation times they incur.

**Result 1.** *The MAXCU allocation of tours led to a significant reduction of truck waiting times between 11.54% and 23.86% depending on the number of trucks and warehouse capacities. This reduction was significantly larger than with FCFS reservations, which reduced truck waiting times by 4.52% to 9.4%. Waiting time savings in MAXCU were also close to those with a core-selecting auction, which ranged from 17.44% to 27.08%.*

While the waiting time reduction with the auction mechanism was highest, it is surprising that a polynomial-time randomized mechanism (MAXCU) without payments is almost as efficient. Table 2 shows the average waiting time for the FCFS, MAXCU, and Core0.2 mechanisms as well as the differences to the treatments with no coordination. We analyze competitive situations with low and high waiting times within a 900 minutes (60 slots · 15 min) planning horizon. Waiting times of several hours have been observed in the field (Bundesverband Güterkraftverkehr Logistik und Entsorgung e.V., 2013). The differences in the waiting times increase with larger problem sizes and the coordination turns out to be particularly beneficial for situations where some loading docks are in high demand.

| number of trucks | warehouses per truck | warehouse capacity | mechanism | avg. waiting time | avg. diff. no coord. | |
|---|---|---|---|---|---|---|
| 50 | {4, 5, 6} | {1, 2, 3} | no coord. | 229.41 | | |
| 50 | {4, 5, 6} | {1, 2, 3} | FCFS | 219.04 | 10.37 | (4.52%) |
| 50 | {4, 5, 6} | {1, 2, 3} | MAXCU | 202.93 | 26.48 | (11.54%) |
| 50 | {4, 5, 6} | {1, 2, 3} | Core0.2 | 189.41 | 40.00 | (17.44%) |
| 100 | {4, 5, 6} | {2, 3, 4} | no coord. | 271.80 | | |
| 100 | {4, 5, 6} | {2, 3, 4} | FCFS | 255.37 | 16.43 | (6.04%) |
| 100 | {4, 5, 6} | {2, 3, 4} | MAXCU | 229.67 | 42.13 | (15.50%) |
| 100 | {4, 5, 6} | {2, 3, 4} | Core0.2 | 219.38 | 52.42 | (19.29%) |
| 200 | {4, 5, 6} | {3, 4, 5} | no coord. | 490.55 | | |
| 200 | {4, 5, 6} | {3, 4, 5} | FCFS | 456.57 | 33.98 | (6.93%) |
| 200 | {4, 5, 6} | {3, 4, 5} | MAXCU | 395.63 | 94.92 | (19.35%) |
| 200 | {4, 5, 6} | {3, 4, 5} | Core0.2 | 389.77 | 100.78 | (20.54%) |
| 300 | {4, 5, 6} | {5, 7, 9} | no coord. | 389.31 | | |
| 300 | {4, 5, 6} | {5, 7, 9} | FCFS | 357.88 | 31.43 | (8.07%) |
| 300 | {4, 5, 6} | {5, 7, 9} | MAXCU | 307.31 | 82.00 | (21.06%) |
| 300 | {4, 5, 6} | {5, 7, 9} | Core0.2 | 296.45 | 92.86 | (23.85%) |
| 400 | {4, 5, 6} | {6, 8, 10} | no coord. | 484.70 | | |
| 400 | {4, 5, 6} | {6, 8, 10} | FCFS | 443.54 | 41.16 | (8.49%) |
| 400 | {4, 5, 6} | {6, 8, 10} | MAXCU | 375.68 | 109.02 | (22.49%) |
| 400 | {4, 5, 6} | {6, 8, 10} | Core0.2 | 365.39 | 119.31 | (24.62%) |
| 500 | {4, 5, 6} | {9, 12, 15} | no coord. | 359.42 | | |
| 500 | {4, 5, 6} | {9, 12, 15} | FCFS | 325.65 | 33.77 | (9.40%) |
| 500 | {4, 5, 6} | {9, 12, 15} | MAXCU | 273.67 | 85.75 | (23.86%) |
| 500 | {4, 5, 6} | {9, 12, 15} | Core0.2 | 262.09 | 97.33 | (27.08%) |

Table 2: Waiting times (in minutes) per tour and average saving in times compared to no coordination for all carriers.

The differences in waiting times between the matching mechanism, the auction mechanism, the experiments without coordination, and FCFS were all significant at $p < 0.001$ using Wilcoxon signed rank tests. We also use a linear regression to compare treatments and control for the number of trucks, the warehouses visited per truck, and the warehouse capacity. The differences are also significant (adjusted $R^2 = 0.77$, $p < 0.001$). Table B.1 in Appendix B (supplementary material) provides average waiting times per loading ramp for all treatment combinations. Note that in our simulations we analyze very broad set of demand scenarios, some with very low waiting times, but also some extreme cases with waiting times up to a working day.

We want to study the waiting time savings in all these different scenarios.

Note that the waiting time reductions reported are based on averages for all carriers across scenarios. Winners in the allocation, i.e., carriers who receive one of the bundles of time slots at warehouse loading docks, would ideally have no waiting time at all. Due the fact that there are stochastic travel and service times also winners have some waiting time in the simulation. For example, if the carrier with a reservation has not yet arrived, but another carrier without a reservation is at the loading dock, then the one without reservation would be processed. If the carrier with reservation arrives later, he needs to wait until the one without reservation is finished (no preemption). These waiting times of winners were very low, however.

**Result 2.** *Computing an optimal solution to the deterministic WDP is intractable for larger problem sizes (with more than 200 trucks and 4 warehouses per truck in our experiments). In contrast, MAXCU and the near-optimal solutions for Core0.2 could be computed in less than two minutes on average even for more than 500 trucks and 6 warehouses per truck.*

The computation of optimal solutions assuming complete information via a mixed integer programming approach is time consuming and requires large amounts of RAM. We computed optimal allocations (OPT) only for the environments with 200 trucks and 4 warehouses. While the average computation times were around 45 seconds, some instances took several minutes even for these instances with small size. In contrast, MAXCU and Core0.2 could be computed in less than 10 seconds for these instances (see Table 3). Overall, MAXCU was remarkably fast for the instances used in our experiments.

| mechanism | avg. | max. |
|---|---|---|
| MAXCU | 5.79 | 7.33 |
| Core0.2 | 5.39 | 7.33 |
| OPT | 45.51 | 260.85 |

Table 3: Computation times (in seconds) for 200 trucks and 4 warehouses per truck.

Table B.2 in Appendix B (supplementary material) provides average and maximum computation times per number of trucks and warehouse capacity. There are some small problem sizes, where even the computation of near-optimal solutions (Core0.2) takes much longer than the polynomial time algorithm MAXCU, which is due to the fact that the underlying optimization problem is NP-hard and we use a branch-and-cut algorithm. Even for a duality gap of 20%, the computation times can vary a lot.

For larger problem instances with 300 trucks as well as 200 trucks and 5–6 warehouses per truck, we were not able to compute all optimal solutions and got out-of-memory exceptions, while we could always compute the MAXCU and Core0.2 solutions within about two and five minutes respectively (see Table 4), which is a considerable advantage over an exact optimization approach. On computing nodes with less computational performance (two 10-core CPUs Intel Xeon E5-2660 v2) but 240GB of RAM, we were able to solve some (but not all) of the scenarios for 200 trucks and 5 warehouses per truck to optimality, which took more than 13 hours to compute. That is, from the 5 scenarios with randomly drawn locations for warehouses and carriers, we were able to solve a single one for the optimal allocation mechanism in about 13.25 hours, while others failed due to limited RAM of our computing nodes (240GB).

| number of trucks | warehouses per truck | warehouse capacity | mechanism | avg. | max. |
|---:|---:|---|---|---|---:|
| 50 | {4,5,6} | {1,2,3} | MAXCU | 1.58 | 3.36 |
| 50 | {4,5,6} | {1,2,3} | Core0.2 | 25.58 | 226.60 |
| 100 | {4,5,6} | {2,3,4} | MAXCU | 3.99 | 8.48 |
| 100 | {4,5,6} | {2,3,4} | Core0.2 | 4.34 | 28.30 |
| 200 | {4,5,6} | {3,4,5} | MAXCU | 11.18 | 25.40 |
| 200 | {4,5,6} | {3,4,5} | Core0.2 | 12.54 | 36.00 |
| 300 | {4,5,6} | {5,7,9} | MAXCU | 20.59 | 42.47 |
| 300 | {4,5,6} | {5,7,9} | Core0.2 | 34.82 | 98.02 |
| 400 | {4,5,6} | {6,8,10} | MAXCU | 31.45 | 62.75 |
| 400 | {4,5,6} | {6,8,10} | Core0.2 | 58.01 | 144.41 |
| 500 | {4,5,6} | {9,12,15} | MAXCU | 47.59 | 92.24 |
| 500 | {4,5,6} | {9,12,15} | Core0.2 | 116.95 | 299.85 |

Table 4: Computation times (in seconds) for MAXCU and Core0.2 mechanisms.

At the same time the waiting time savings of MAXCU were almost as high as those computing an optimal allocation. In Table 5 we compared the waiting time savings for those experiments with 200 trucks and 4 warehouses per truck. The average savings for MAXCU were 65.36 minutes (17.48%) per tour, while those for the optimal allocation were 79.64 minutes (21.30%)

with the "no coordination" as a baseline.

| mechanism | avg. waiting time | avg. diff. no coord. | |
|---|---|---|---|
| no coord. | 373.89 | | |
| FCFS | 349.12 | 24.77 | (6.62%) |
| MAXCU | 308.53 | 65.36 | (17.48%) |
| Core0.2 | 300.69 | 73.20 | (19.58%) |
| OPT | 294.25 | 79.64 | (21.30%) |

Table 5: Waiting times (in minutes) per tour for 200 trucks and 4 warehouses per truck and average saving in times compared to no coordination.

To better understand the average problem sizes solved, Table 6 provides the average total number of reported preferences and the average number of preferences per truck in the experiments. We also report the average number of winning carriers across all treatments. Table B.3 (Appendix B, supplementary material) provides details on reported preferences for tours and winning carriers per number of trucks and warehouse capacity.

| number of trucks | warehouses per truck | mechanism | avg. no. pref. | avg. no. winners | avg. no. pref. p. truck | avg. no. win. p. truck |
|---|---|---|---|---|---|---|
| 50–500 | $\{4,5,6\}$ | MAXCU | 2117.31 | 120.70 | 8.19 | 0.50 |
| 50–500 | $\{4,5,6\}$ | Core0.2 | 2117.31 | 111.31 | 8.19 | 0.45 |
| 200 | $\{4\}$ | MAXCU | 1045.67 | 81.67 | 5.23 | 0.41 |
| 200 | $\{4\}$ | Core0.2 | 1045.67 | 74.87 | 5.23 | 0.37 |
| 200 | $\{4\}$ | OPT | 1045.67 | 89.20 | 5.23 | 0.45 |

Table 6: Average numbers of reported preferences for tours and winning carriers.

**Result 3.** *The percentage of time slots with capacity violations in MAXCU ranges from averages of 6.00 to 10.28% depending on the size of the scenario. However, these violations were small relative to the capacity available, ranging from average values of 1.23 to 4.60% for different sizes of scenarios.*

While computable in polynomial time, one of the downsides of MAXCU is that some of the capacity constraints can be violated. Table 7 shows the average number of capacity violations in absolute numbers and relative to the number of time slots available overall. The column "avg. sum viol." takes into account not only the number of time slots where the capacity was violated, but also how much it was violated and sums up these violations.

This column is expressed relative to the total capacity available in column "% of capacity". Overall, the level of violations is small and a planner could take such levels into account when setting the capacity levels in the optimization.

| number of trucks | warehouses per truck | warehouse capacity | avg. no. viol. | % of time slots | avg. sum viol. | % of capacity |
|---|---|---|---|---|---|---|
| 50 | $\{4, 5, 6\}$ | $\{1, 2, 3\}$ | 35.99 | 6.00 | 42.08 | 4.60 |
| 100 | $\{4, 5, 6\}$ | $\{2, 3, 4\}$ | 45.74 | 7.62 | 56.77 | 3.55 |
| 200 | $\{4, 5, 6\}$ | $\{3, 4, 5\}$ | 59.37 | 9.89 | 79.66 | 3.50 |
| 300 | $\{4, 5, 6\}$ | $\{5, 7, 9\}$ | 57.53 | 9.59 | 79.45 | 2.05 |
| 400 | $\{4, 5, 6\}$ | $\{6, 8, 10\}$ | 61.67 | 10.28 | 86.60 | 1.90 |
| 500 | $\{4, 5, 6\}$ | $\{9, 12, 15\}$ | 59.10 | 9.85 | 83.42 | 1.23 |

Table 7: Overview of supply constraint violations.

**Result 4.** *Payments by carriers in a VCG auction are very low. In core-selecting auctions (Core0.2) the payments are almost as high as if carriers had to pay their bid.*

The randomized matching mechanism MAXCU does not incure payments by the carriers. Payments by carriers for reservations can be a significant barrier to adoption in a low-margin business. Table 8 provides an overview of the average auctioneer revenue and the average payments per carrier assuming different payment rules. The pay-as-bid values in the first line in the table denote the sum of the values in the optimal allocation for a pay-as-bid pricing rule using near-optimal allocations. Obviously, there is no reason to believe that carriers would bid truthful in such an auction, but it is a good baseline to compare other payment rules against. We also computed VCG payments, but adapted the payments such that they could not be negative or above the bid. The average VCG revenue and payments are very low, but they show a high standard deviation in comparison to other payment rules. The revenue raised via the core-selecting payment rule is much higher than that in the VCG mechanism and close to the pay-as-bid payments. The competition in our scenarios led to many core constraints and core violations of the VCG mechanism.

## 4. Conclusions

Congestion at loading docks of retail warehouses is a substantial problem in retail transportation logistics and an example of coordination problems as

| | revenue | | payment | |
|---|---|---|---|---|
| mechanism | avg. | std. dev. | avg. | std. dev. |
| Pay-as-bid (Core0.2) | 1024.02 | 621.85 | 11.67 | 1.29 |
| VCG (Core0.2) | 55.09 | 142.70 | 0.63 | 2.63 |
| Core (Core0.2) | 885.65 | 532.60 | 10.16 | 3.88 |

Table 8: Auctioneer revenue and payments by carriers (in monetary units).

they often arise in logistics and beyond. The problem is due to the fact that carriers optimize locally, but there is no coordination among the carriers leading to globally suboptimal allocations of warehouse capacities and long waiting times for carriers at loading docks.

Incentive-compatible mechanisms for hard allocation problems of this sort were long considered illusive. However, recent research has shown a way how randomization can be used to effectively address incentives of carriers to report truthfully, computational costs of the mechanism, and feasibility of the resulting assignment. We draw on a framework for randomized matching mechanisms, MAXCU, recently introduced by Nguyen et al. (2016), which strikes a balance between these design desiderata and adapt it to our retail logistics problem. MAXCU provides a significant contribution to mechanism design theory, but the theoretical results on asymptotic incentive-compatibility and feasibility reveal little about the average case waiting time reduction, capacity violations, and actual computation times we can expect in a realistic environment such as our logistics problem. Therefore, we report the results of numerical experiments in which we draw on data from a real-world logistics network. We analyze environments with several hundred trucks and show that the efficiency gains (waiting time savings) can be substantial. In our results we compare the randomized matching mechanism to an auction mechanism with a core-selecting payment rule that was adapted for near-optimal computations of the assignment problem, and to a first-come, first-served rule. There is a trade-off between the capacity violations incurred by MAXCU and possibly high payments resulting from an auction mechanism. It turns out that the capacity violation with MAXCU are low and the waiting time reductions close to those achieved with the auction mechanism. In contrast, the first-come, first-served policy only leads to a small improvement in waiting times. If the platform provider allows for a large enough duality gap when computing the assignment problem with a branch-and-cut algorithm, the computation times for the auction mechanism in our scenario are as fast as those of MAXCU. In summary, if payments by the carriers were not a concern, then a near-optimal auction provides an alternative. However, in a low-margin logistics business the randomized MAXCU mechanism provides a compelling alternative to auctions, which runs in polynomial time and has surprisingly high time savings and at no cost to the participants. As any experimental research, also this paper has limitations. We analyze characteristic environments with various parameter settings, but for specific applications there will always be differences to the underlying network and in various assumptions and parameters of our simulation. In addition, the behavior of carriers in logistics networks without coordination can differ in the field. For example, carriers often have historical information about waiting times, which they can factor into their tour planning. This is difficult to simulate and the problem that carriers have is akin to that of the Kolkata Paise Restaurant problem (Chakrabarti et al., 2009). If every carrier responds to historical information, waiting times can be at the same level as if no carrier responded with congestion at different nodes, and it is far from obvious which and how many carriers should consider this information without a central coordination mechanism.

Overall, our analysis illustrates that for a realistic environment and a large number of parameter settings the average gains in waiting time by the randomized MAXCU mechanism are substantial, that it achieves high computational and allocative efficiency, is incentive-compatible and envy-free, and does not even require payments by the participants, which is important for the adoption in practice.

**Acknowledgments**

We thank the anonymous reviewers whose comments helped to improve and clarify this manuscript. Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 405008493.

**References**

Abdulkadiroğlu, A. and Sönmez, T. (2003). School choice: A mechanism design approach. *American Economic Review*, 93(3):729–747.

Agrali, S., Tan, B., and Karaesmen, F. (2008). Modeling and analysis of an auction-based logistics market. *European Journal of Operational Research*, 191(1):272–294.

Azevedo, E. M. and Budish, E. (2019). Strategy-proofness in the large. *The Review of Economic Studies*, 86(1):81–116.

Bichler, M. (2017). *Market Design: A Linear Programming Approach to Auctions and Matching*. Cambridge University Press.

Bichler, M. and Goeree, J. K. (2017). *Handbook of spectrum auction design*. Cambridge University Press.

Budish, E. (2011). The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy*, 119(6):1061–1103.

Bundesamt für Güterverkehr (2011). Sonderbericht zur Situation an der Laderampe.

Bundesamt für Güterverkehr (2018). Abläufe an den Laderampen verbessern.

Bundesverband Güterkraftverkehr Logistik und Entsorgung e.V. (2013). Jahresbericht 2012/2013.

Caplice, C. (2007). Electronic markets for truckload transportation. *Production and Operations Management*, 16(4):423–436.

Chakrabarti, A. S., Chakrabarti, B. K., Chatterjee, A., and Mitra, M. (2009). The Kolkata paise restaurant problem and resource utilization. *Physica A: Statistical Mechanics and its Applications*, 388(12):2420–2426.

Day, R. and Cramton, P. (2012). The quadratic core-selecting payment rule for combinatorial auctions. *Operations Research*, 60:588–603.

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

Delorme, M., Garca, S., Gondzio, J., Kalcsics, J., Manlove, D., and Pettersson, W. (2019). Mathematical models for stable matching problems with ties and incomplete lists. *European Journal of Operational Research*, 277(2):426–441.

Diebold, F. and Bichler, M. (2017). Matching with indifferences: A comparison of algorithms in the context of course allocation. *European Journal of Operational Research*, 260(1):268–282.

Elmaghraby, W. and Keskinocak, P. (2004). Combinatorial auctions in procurement. In Billington, C., Harrison, T., Lee, H., and Neale, J., editors, *The Practice of Supply Chain Management: Where Theory and Application Converge*, pages 245–258. Kluwer Academic Publishers.

Gale, D. and Shapley, L. S. (1962). College admissions and the stability of marriage. *American Mathematical Monthly*, 69(1):9–15.

Gansterer, M., Hartl, R. F., and Vetschera, R. (2019). The cost of incentive compatibility in auction-based mechanisms for carrier collaboration. *Networks*, 73(4):490–514.

Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman.

Goetzendorff, A., Bichler, M., Shabalin, P., and Day, R. W. (2015). Compact bid languages and core pricing in large multi-item auctions. *Management Science*, 61(7):1684–1703.

Huang, G. Q. and Xu, S. X. (2013). Truthful multi-unit transportation procurement auctions for logistics e-marketplaces. *Transportation Research Part B: Methodological*, 47:127–148.

Jenelius, E. and Koutsopoulos, H. N. (2013). Travel time estimation for urban road networks using low frequency probe vehicle data. *Transportation Research Part B: Methodological*, 53:64–81.

Karaenke, P., Bichler, M., and Minner, S. (2019). Coordination is hard: Electronic auction mechanisms for increased efficiency in transportation logistics. *Management Science*, 65(12):5449–5956.

Kleinberg, J., Ludwig, J., Mullainathan, S., and Rambachan, A. (2018). Algorithmic fairness. In *AEA Papers and Proceedings*, volume 108, pages 22–27.

Leyton-Brown, K., Nudelman, E., and Shoham, Y. (2006). Empirical hardness models for combinatorial auctions. In Cramton, P., Shoham, Y., and Steinberg, R., editors, *Combinatorial Auctions*. MIT Press, Cambridge, MA.

Lu, W. and Quadrifoglio, L. (2019). Fair cost allocation for ridesharing services  modeling, mathematical programming and an algorithm to find the nucleolus. *Transportation Research Part B: Methodological*, 121:41 – 55.

Nguyen, T., Peivandi, A., and Vohra, R. (2016). Assignment problems with complementarities. *Journal of Economic Theory*, 165:209 – 241.

Roth, A. E. (1982). The economics of matching: Stability and incentives. *Mathematics of Operations Research*, 7(4):617–628.

Triki, C., Oprea, S., Beraldi, P., and Crainic, T. G. (2014). The stochastic bid generation problem in combinatorial transportation auctions. *European Journal of Operational Research*, 236(3):991–999.

Xu, S. X. and Huang, G. Q. (2014). Efficient auctions for distributed transportation procurement. *Transportation Research Part B: Methodological*, 65:47 – 64.

Zhu, F. and Ukkusuri, S. V. (2017). Efficient and fair system states in dynamic transportation networks. *Transportation Research Part B: Methodological*, 104:272 – 289.

## Appendix A. Additional Treatments

### *Appendix A.1. Uniformly distributed travel times*

To investigate the effect of the distributional assumptions, we conducted additional experiments with a uniform distribution and ran the experiments for 100 and 400 carriers. Note that for different travel time distributions, preference and computation times did not change in our experiments, because carriers use expected values that did remained constant across experiments with different distributions. Hence, we only report the waiting times in Table A.9.

| number of trucks | warehouses per truck | warehouse capacity | mechanism | avg. waiting time | avg. diff. no coord. | |
|---|---|---|---|---|---|---|
| 100 | {4,5,6} | {2,3,4} | no coord. | 265.97 | | |
| 100 | {4,5,6} | {2,3,4} | FCFS | 250.45 | 15.52 | (5.84%) |
| 100 | {4,5,6} | {2,3,4} | MAXCU | 226.51 | 39.46 | (14.84%) |
| 100 | {4,5,6} | {2,3,4} | Core0.2 | 215.06 | 50.91 | (19.14%) |
| 400 | {4,5,6} | {6,8,10} | no coord. | 477.53 | | |
| 400 | {4,5,6} | {6,8,10} | FCFS | 437.51 | 40.02 | (8.38%) |
| 400 | {4,5,6} | {6,8,10} | MAXCU | 371.15 | 106.38 | (22.28%) |
| 400 | {4,5,6} | {6,8,10} | Core0.2 | 361.74 | 115.79 | (24.25%) |

Table A.9: Waiting times (in minutes) per tour and average saving in times compared to no coordination for all carriers with uniformly distributed travel times.

*Appendix A.2. 8 and 12 warehouses*

To explore low- and high-supply scenarios, we conducted additional experiments for 100 and 400 carriers each with 8 and 12 warehouses. Table A.10 shows the average waiting time for the FCFS, MAXCU, and Core0.2 mechanisms as well as the differences to the treatments with no coordination for these additional experiments. In contrast to the additional treatments regarding the distributional assumptions, computation times and supply constraint violations are different from our initial experiments; they are shown in Tables A.11 and A.12.

| number of trucks | number of warehouses | warehouses per truck | warehouse capacity | mechanism | avg. waiting time | avg. diff. no coord. | |
|---|---|---|---|---|---|---|---|
| 100 | 8 | {4, 5, 6} | {2, 3, 4} | no coord. | 391.56 | | |
| 100 | 8 | {4, 5, 6} | {2, 3, 4} | FCFS | 368.84 | 22.72 | (5.8%) |
| 100 | 8 | {4, 5, 6} | {2, 3, 4} | MAXCU | 324.81 | 66.75 | (17.05%) |
| 100 | 8 | {4, 5, 6} | {2, 3, 4} | Core0.2 | 319.31 | 72.25 | (18.45%) |
| 100 | 12 | {4, 5, 6} | {2, 3, 4} | no coord. | 204.72 | | |
| 100 | 12 | {4, 5, 6} | {2, 3, 4} | FCFS | 192.89 | 11.83 | (5.78%) |
| 100 | 12 | {4, 5, 6} | {2, 3, 4} | MAXCU | 176.50 | 28.22 | (13.78%) |
| 100 | 12 | {4, 5, 6} | {2, 3, 4} | Core0.2 | 163.29 | 41.43 | (20.24%) |
| 400 | 8 | {4, 5, 6} | {6, 8, 10} | no coord. | 671.82 | | |
| 400 | 8 | {4, 5, 6} | {6, 8, 10} | FCFS | 620.93 | 50.89 | (7.57%) |
| 400 | 8 | {4, 5, 6} | {6, 8, 10} | MAXCU | 533.60 | 138.22 | (20.57%) |
| 400 | 8 | {4, 5, 6} | {6, 8, 10} | Core0.2 | 525.14 | 146.68 | (21.83%) |
| 400 | 12 | {4, 5, 6} | {6, 8, 10} | no coord. | 362.94 | | |
| 400 | 12 | {4, 5, 6} | {6, 8, 10} | FCFS | 330.09 | 32.85 | (9.05%) |
| 400 | 12 | {4, 5, 6} | {6, 8, 10} | MAXCU | 278.98 | 83.96 | (23.13%) |
| 400 | 12 | {4, 5, 6} | {6, 8, 10} | Core0.2 | 269.54 | 93.4 | (25.73%) |

Table A.10: Waiting times (in minutes) per tour and average saving in times compared to no coordination for all carriers with 8 and 12 warehouses.

| number of trucks | number of warehouses | warehouses per truck | warehouse capacity | mechanism | avg. | max. |
|---|---|---|---|---|---|---|
| 100 | 8 | {4,5,6} | {2,3,4} | MAXCU | 2.85 | 5.54 |
| 100 | 8 | {4,5,6} | {2,3,4} | Core0.2 | 3.00 | 12.06 |
| 100 | 12 | {4,5,6} | {2,3,4} | MAXCU | 5.39 | 15.21 |
| 100 | 12 | {4,5,6} | {2,3,4} | Core0.2 | 7.43 | 52.63 |
| 400 | 8 | {4,5,6} | {6,8,10} | MAXCU | 29.23 | 63.63 |
| 400 | 8 | {4,5,6} | {6,8,10} | Core0.2 | 38.30 | 91.52 |
| 400 | 12 | {4,5,6} | {6,8,10} | MAXCU | 38.46 | 75.78 |
| 400 | 12 | {4,5,6} | {6,8,10} | Core0.2 | 72.82 | 181.32 |

Table A.11: Computation times (in seconds) for MAXCU and Core0.2 mechanisms with 8 and 12 warehouses.

| number of trucks | number of warehouses | warehouses per truck | warehouse capacity | avg. no. viol. | % of time slots | avg. sum viol. | % of capacity |
|---|---|---|---|---|---|---|---|
| 100 | 8 | {4,5,6} | {2,3,4} | 41.89 | 8.73 | 53.17 | 4.11 |
| 100 | 12 | {4,5,6} | {2,3,4} | 50.32 | 6.99 | 62.32 | 3.26 |
| 400 | 8 | {4,5,6} | {6,8,10} | 53.71 | 11.19 | 77.39 | 2.11 |
| 400 | 12 | {4,5,6} | {6,8,10} | 68.36 | 9.50 | 94.09 | 1.74 |

Table A.12: Overview of supply constraint violations with 8 and 12 warehouses.