OR Spectrum (2024) 46:953–985
https://doi.org/10.1007/s00291-024-00762-2

ORIGINAL ARTICLE

The dynamic stochastic container drayage problem with truck appointment scheduling

Kenneth Stoop[1] · Mario Pickavet[1] · Didier Colle[1] · Pieter Audenaert[1]

Received: 10 August 2023 / Accepted: 3 April 2024 / Published online: 26 April 2024

Abstract

In this work, a stochastic dynamic version of the container drayage problem is studied. The presented model incorporates uncertainty in the form of stochastic loading and unloading times at both terminals and customers, as well as stochastic travel times, conditionally dependent upon the departure time, allowing robust planning with respect to varying processing times. Moreover, the presented model is dynamic, allowing flexible orders and having the capability of re-solving the optimization problem in case of last-minute orders. Finally, the model also incorporates a truck appointment system operating at each terminal. First, a description of the general model is given, which amounts to a mixed integer non-linear program. In order to efficiently solve the optimization problem, and linearize both the objective and the conditional chance constraints, it is reformulated based on time window partitioning, yielding a purely integer linear program. As a test case, a large road carrier operating in the port of Antwerp is considered. We demonstrate that the model is efficiently solvable, even for instances of up to 300 orders. Moreover, the impact of incorporating stochastic information is clearly illustrated.

Keywords Container drayage problem · Truck appointment scheduling · Real-time · Stochastic · Routing optimization

✉ Kenneth Stoop
kenneth.stoop@ugent.be

Mario Pickavet
mario.pickavet@ugent.be

Didier Colle
didier.colle@ugent.be

Pieter Audenaert
pieter.audenaert@ugent.be

[1] IDLab, Ghent University - imec, 9000 Ghent, Belgium

1 Introduction

Considering global trades, the short-distance road transport of containers—drayage—is an important link in the associated logistic chain (see Fig. 1) and contributes roughly 30% to the total long-distance transportation cost (Spasovic and Morlok 1993). Drayage operations encompass all short-distance transports of containers over the road, as sea vessels or trains lack door-to-door services. Ever-increasing overseas trades make the problem of optimizing these drayage operations as relevant as ever.

The container drayage problem (CDP) models these drayage operations and aims to optimize them with respect to e.g. traveled distance, number of trucks, operational costs, profits, etc. The CDP is typically handled as a pickup and delivery problem (PDP) (Nossack and Pesch 2013) or a multiple traveling salesman problem (m-TSP) (Wang and Regan 2002), variants of the well-known rich class of vehicle routing problems (VRP) (Toth and Vigo 2014). In the general PDP, transport requests are point-to-point transports, i.e. for each transportation request, goods are picked up at one location, and goods are delivered to another, see Parragh et al. (2008a, b) for an overview. In order to model the CDP however, an extra constraint is necessary, namely, no fractional shipments are allowed, and a container is always transported as a whole. The m-TSP is a variant of the renowned TSP, where multiple salesmen are considered, see Cheikhrouhou and Khoufi (2021) for an overview of the general problem. Just like the related VRP (Toth and Vigo 2002), finding the optimal solution for the CDP is NP-hard (Imai et al. 2007). This limits the size of the problems that can be solved optimally and means approximate methods have to be used for larger problems.

Typically, the problems considered in literature are assumed to be static and deterministic. The assumption of full information is made and a complete planning is computed before operations commence. In practice, however, problems are rarely static or deterministic. For example, orders may be canceled or changed during operations, new high-priority orders might pop up, truck or driver availabilities might change, etc. Moreover, often fixed operating-, loading-, unloading- and travel-times are assumed, but in practice, the uncertainty upon these quantities is non-negligible;

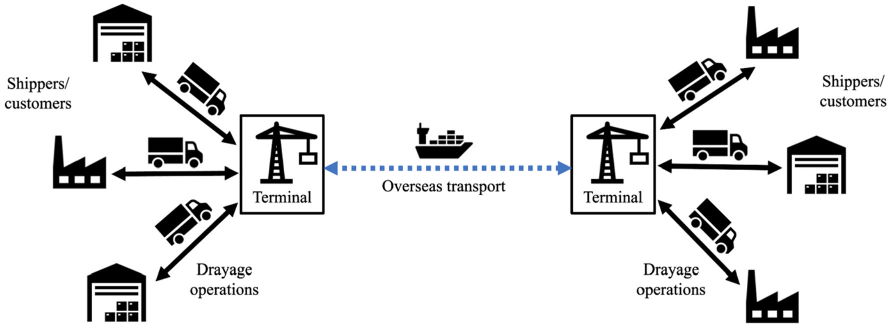

Fig. 1 A simplified schematic of overseas transport

truck turnaround times in the container terminals might vary due to different internal operations, loading or unloading at customers might also experience unforeseen delays, and finally, travel times along the traffic network are dependent on the state of traffic and congestion and will vary considerably.

Taking these facts into consideration, we will formulate the CDP with conditional (i.e. time-dependent) chance constraints based on given general probability distributions which are conditioned upon the time of initiating certain activities. Additionally, the model will be capable of handling flexible orders and recomputing a planning. Finally, a truck appointment system (TAS) operational at multiple terminals will be incorporated into the mathematical model. Such a TAS is employed by the terminals in order to spread the arrivals of trucks and avoid congestion. Each time a container is being transported to or from a terminal, a time slot/window should be booked through the TAS for the respective truck.

The CDP considered here will include both full and empty containers. The problem of transporting empty containers will be incorporated by using requiring and releasing attributes, denoting whether an empty container is needed at the pick-up and if one is released at the drop-off. Empty containers are assumed to be stored at one of the depots of the transport company. It is also assumed that sufficient empty containers are available at these depots. The objective will be minimizing the operation time needed to finish all orders, i.e. the sum of the time spent by each truck during operations.

The full optimization model results in a mixed integer non-linear program (MINLP). However, we will reformulate it based on a time window partitioning, which results in a pure integer linear program (ILP), while still allowing completely general conditional probability distributions to be integrated into the model. Moreover, this linearization will allow the model to be solved and dynamically re-solved by highly efficient existing integer linear program solvers. Our contributions can thus be summarized as follows:

- Incorporating stochasticity (loading times, unloading times, driving times, ...), flexible orders, and a TAS into one container drayage model
- Incorporating chance constraints which are conditionally dependent on departure time
- Linearizing these non-linear conditional chance constraints by time window partitioning
- Testing on large instances based on the real-world case of the port of Antwerp

To the best of our knowledge, we are the first to propose a model for the stochastic CDP with conditionally dependent chance constraints. Moreover, we believe we are the first to apply time window partitioning in this context in order to be able to solve this MINLP, while still using the exact forms for the probability distributions.

The remainder of this article is structured as follows: Sect. 2 will give an overview of existing work that is relevant to the content of the article; Sect. 3 will give a detailed description of the mathematical model that is used; in Sect. 4 some context will be given for the experimental test case used; Sect. 5 contains the obtained experimental results; and finally, Sect. 6 covers the final discussion and conclusions.

2 Literature review

2.1 The container drayage problem

The CDP has been, like the more general class of VRPs, the subject of quite some research. The authors of Wang and Regan (2002) study the container drayage problem as “local truckload pickup and delivery” along with time window constraints and formulate it as a m-TSP with time window constraints (m-TSPTW). Moreover, they propose a time window partitioning scheme to solve the model more efficiently. In Jula et al. (2005) the CDP is considered with time windows and social constraints on the working time of truck drivers, which they formulate as an m-TSPTW. In Imai et al. (2007) the VRP with full container loads is tackled by a heuristic based on the Lagrangian relaxation of the problem containing two subproblems: the classical assignment problem and the generalized assignment problem. More recently, the authors of Escudero-Santana et al. (2021) conduct an in-depth literature review on drayage optimizations. In more recent work, platooning is proposed as a way of reducing human labor and improving fuel efficiency in truck transport. The authors of You et al. (2023) study the CDP with truck platooning and propose an exact algorithm based on Branch-and-Price-and-Cut which allows them to solve instances with up to 50 orders. Similarly, in Yan et al. (2023) the CDP with improved platooning operations is studied, where drivers are not attached to their respective leading trucks but can move by alternative transport modes. The authors employ a heuristic based on simulated annealing to solve the problem. Instances with up to 400 orders are solved.

Some studies consider discretization or partitioning of time windows as a means to handle continuous time variables and time windows, by replacing them with a set of assignment variables. One of the earliest works in which this method is applied is Appelgren (1971), where a ship-scheduling problem is solved based on time discretization, while in Levin (1971) a similar method is used to generate flight assignments. The authors of Wang and Regan (2002) are the first to apply window partitioning in the context of the m-TSPTW such as container drayage optimization. In Zhang et al. (2010) the authors build upon the results of Wang and Regan (2002) and show a clear improvement in the computation time required to solve the CDP.

In the CDP considered in this work, the focus lies on a realistic case of significant size. We will consider a transport company with several depots, fleet sizes ranging up to 120 trucks and a number of orders/tasks ranging up to 300. The generated orders will be distributed over several container terminals and customers distributed over an area covering the port of Antwerp. Our solution method will be based on making a linear approximation to the model by time window partitioning and using existing efficient solvers for this new model.

2.2 Empty containers

The problem of drayage optimization in actuality consists of 2 subproblems, namely a VRP, and an empty container allocation problem. These two problems are sometimes solved in a sequential manner, where the objective of the first is to find the optimal tours, while the objective of the latter is to optimally distribute empty containers among customer locations, depots and terminals, based on supply and demand. The empty container allocation problem in itself has been studied by different authors Chang et al. (2008), Braekers et al. (2011), Kuzmicz and Pesch (2019). The VRP and empty container allocation problem have also been solved together in an integrated manner (Sterzik and Kopfer 2013). The authors of Zhang et al. (2009) for example propose a cluster method and a reactive tabu search heuristic to solve the integrated problem, which they formulate as a m-TSPTW. In Zhang et al. (2011), the authors extend their previous work, considering empty containers as a transportation resource, and assuming a finite number of empty containers available at the depot. More recently, the authors of Song et al. (2023) consider the CDP with a limited number of empty containers at the depot, for which they propose a Branch-and-Price algorithm and solve instances with up to 40 orders. They incorporate a so-called “drop-and-pickup mode”, which means containers can be dropped off by the truck to be packed/unpacked, and can be picked up later. On the other hand, often, it is assumed that enough empty containers are available at the depots and that an unlimited number of empty containers can be stored at the depots, such as in e.g. Zhang et al. (2014), Shiri et al. (2019), Chen et al. (2021). The same assumption will be made in the work presented here.

2.3 Truck appointment system

With the aim of reducing congestion and waiting times and improving their overall efficiency, many terminals have introduced a truck appointment system (TAS) (Huynh et al. 2016). The main idea behind a TAS is to set up time windows which can be booked by truckers who want to pick up or deliver a certain container. The number of times a certain time slot can be booked is limited, which allows the terminal operators to effectively control the truck arrival rates such that the number of visiting trucks can be spread out more evenly throughout the day. Some work has been done in designing a TAS with the impact on drayage tours taken into account by the authors of Torkjazi et al. (2018). The reverse, integrating a TAS into the CDP, has been considered in only a few studies (Namboothiri and Erera 2008; Shiri and Huynh 2016; Shiri et al. 2019), all of which considered a single TAS.

In the optimization model presented in this article, several TAS operating at different container terminals will be included, inspired by the work of Shiri and Huynh (2016).

2.4 Stochastic process times

Integrating uncertainty and stochasticity into VRPs has been the subject of earlier studies as well. Uncertainty can arise in different ways, such as stochastic orders, stochastic service times at customers, or stochastic travel times between customers (Oyola et al. 2018). In order to incorporate stochasticity into the problem, different techniques are discerned. Broadly speaking, these techniques can be divided into: chance-constrained optimization (Li et al. 2008), stochastic programming (Birge and Louveaux 2011), and robust optimization (Yu and Li 2000). When stochasticity pertains solely to the constraints rather than the objective function, a conventional method known as chance-constrained optimization is employed. This framework enables the formulation of an optimization problem with the condition that the probability of violating specific constraints should be limited to a predetermined value. In stochastic programming, the probability distributions of the random variables are assumed to be known, and the objective is to optimize the expected value of the objective function. On the other hand, robust optimization focuses on scenarios where the underlying distributions are often unknown (though not necessarily), and the objective is to find a solution that remains robust (i.e. feasible) against all potential uncertainty scenarios. A recent example of a VRP with uncertainty can be found in the work of Messaoud (2023), where the electric vehicle routing problem with stochastic travel times is considered. They define an integer optimization model with chance constraints on the total service time of each vehicle not exceeding a set threshold, and the battery level of each vehicle not going negative. To solve the problem, a metaheuristic approach based on large neighborhood search is proposed, where the chance constraints are checked by Monte Carlo sampling for each generated solution.

In Marković et al. (2014), a truck dispatching model is studied in the setting of truck-train intermodal transport, where uncertainty in truck roundtrip durations and uncertain train departure times are incorporated. In the context of CDPs, there has been little research on integrating uncertainty into the models. The authors of Shiri et al. (2019) study the CDP with a TAS operating at a single terminal, and incorporate stochastic container packing and unpacking times. Their model is based on chance constraints which are linearized by less strict approximations which make no assumptions about the specific form of the probability distributions, but only assume the mean and either the standard deviation or the lower and upper bounds to be known. In You et al. (2021) the CDP with uncertain packing and unpacking times is studied. The authors tackle this problem by employing a tractor-trailer separable mode and constructing an optimization model with an objective containing a cost minimization term, and a robustness maximization term. A robustness measure is defined by the buffer times available between the end of one order and the start of the next one, weighted by an exponential function. No assumptions about the probability distributions are made.

In the model proposed here, all processing times are assumed to be stochastic (i.e. loading, unloading, travel times, ...). The relevant probability distributions are assumed to be known, from which the composite probability distributions are computed which in turn are used in the chance constraints of the model. Moreover, the chance constraints will be conditionally dependent on the time of commencement of specific activities.

2.5 Dynamic problems

Another way of handling unexpected or uncertain events in VRPs or CDPs is by considering dynamic models. An often considered source of dynamism is the presence of flexible orders, i.e. orders which are changed or revealed during operations. For example, in Escudero et al. (2011, 2013), a framework which allows to re-optimize drayage operations based on real-time GPS data of the trucks is proposed. The authors of Zhang et al. (2014) study the CDP with flexible orders, where during operations, orders can be canceled or new orders can be added and the planning can be re-optimized. In order to handle flexible orders, they introduce a temporal vertex set in the graph formulation which represents the orders that are being handled at the decision epoch. The CDP with uncertain job arrivals has also been studied in the context of e-commerce drayage platforms in Chen et al. (2023). The authors propose a multi-stage stochastic programming model which is solved by a two-phase heuristic. The first phase consists of solving a container drayage service booking subproblem in order to decide on accepting or rejecting incoming orders, and determining the time period for carrying out the accepted orders. In the second phase, a fleet routing subproblem is solved to compute a schedule for the trucks. In Bjelić et al. (2022) a dynamic CDP is studied where the sources of uncertainty are dynamic orders, changing travel times, and changes in order time windows and service times. The authors propose a re-optimization scheme for a given set of decision epochs, where the optimization model at each epoch is solved with a heuristic based on a variable neighborhood search. In order to further increase efficiency, they also consider longer vehicle combinations in their solution, i.e. multiple trailers can be combined onto one truck. The authors of Jia et al. (2022) consider a dynamic CDP where request arrival times are unknown, as well as uncertain time windows for the requests. The problem is tackled by a Markov decision process model, which is solved with an integrated reinforcement learning and integer programming method.

The model we will present in this article will also allow dynamic orders to be included. New orders can be added, or existing ones can be changed and the model can be re-optimized in a small amount of time.

3 Problem formulation

The problem that will be tackled in this work is thus stochastic as well as dynamic with respect to flexible orders. Both time windows and a TAS will be incorporated. The objective considered will be minimizing the total drayage operating time needed to process all orders. In the following subsections, the assumptions, notations and formulation will be discussed in more detail.

3.1 Description and assumptions

For the CDP considered here, a set of depot nodes V_D is considered, each having a location L_i and a certain number of identical trucks present $n_i, \forall i \in V_D$ at each decision epoch. At the end of each operating day, the trucks are assumed to return to one of the depots. We will also assume that there is no limit on the number of empty containers that can be stored or retrieved at one of these depots. It is also assumed that a truck and trailer, with the standard capacity for a 40-foot container, transport only one container (either a 40-foot or 20-foot container) at a time, i.e. no 20-foot containers are combined onto one trailer. In practice, two containers are seldom combined onto one trailer, because of the risk of exceeding the legally allowed weight limits on a single truck.

Next, a set of orders V_O is given, all of which should be handled. Each order has a number of attributes: origin O_i; destination D_i; time window $[A_i, B_i]$; type; and a requiring and releasing attribute ϕ_i^Q and ϕ_i^L. Both the origin and destination of an order can be either a customer location or a container terminal. The time window $[A_i, B_i]$ denotes the time frame in which the activities of order $i \in V_O$ should be commenced. The type of the order can be either:

- *Import* the origin of the order is a container terminal, the destination a customer
- *Export* the origin of the order is a customer, the destination a container terminal
- *Transfer* both the origin and destination of the order are a customer

In the formulation of the optimization model, additionally, the subset of import orders and export orders $V_O^I \subset V_O$ and $V_O^E \subset V_O$ will be used. In order to model the flow of empty containers, the requiring and releasing attributes ϕ_i^Q and ϕ_i^L are used, similar to the work in Zhang et al. (2014).

$$\phi_i^Q = \begin{cases} 0 & \text{No empty container is required at the origin of order } i \\ 1 & \text{An empty container is required at the origin of order } i \end{cases} \tag{1}$$

Table 1 Different types of physical transport processes that can be modeled with attributes ϕ_i^Q and ϕ_i^L

Type of physical transport process	ϕ_i^Q	ϕ_i^L
Import order, truck does not wait at customer to unload the container	0	0
Import order, truck waits at customer to unload the container	0	1
Export order, truck picks up a loaded container	0	0
Export order, truck waits at customer to load the container	1	0
Transport a loaded/unloaded container from a customer to another customer	0	0
Load an empty container at a customer and drop it off at another customer	1	0
Pick up a loaded container at a customer and unload it at another customer	0	1
Load a container at a customer and unload it at another customer	1	1

$$\phi_i^L = \begin{cases} 0 & \text{No empty container is released at the destination of order } i \\ 1 & \text{An empty container is released at the destination of order } i \end{cases} \tag{2}$$

Using these two attributes a number of different types of physical transport processes can be modeled, see Table 1.

In order to handle flexible orders and recompute a planning, a set of temporal vertices V_T is introduced, comparable to Zhang et al. (2014). This set contains all orders being processed (i.e. started but not finished) at the time s of a decision epoch. Each vertex resembles the remaining time of the activities of the order being processed. Similarly to order vertices in V_O, these temporal vertices have the following relevant attributes: a destination D_i, type and releasing attribute ϕ_i^L (equal to the ones of the corresponding order vertex). These vertices have no time windows since the starting times have already been fixed. Such a temporary vertex has to precede an order or depot vertex, but no order or depot can be linked to a temporary vertex [see constraints (10d) and (10c) in the optimization model presented in the next subsections].

Finally, a set of arcs A is defined between order vertices, depots, and temporal vertices, representing the transitions. First of all, arcs from depots to order nodes are defined, as well as arcs from order nodes to depots. Secondly, arcs between different order nodes are defined. Finally, we have arcs going from temporal vertices to order vertices.

$$\begin{aligned} A =\{(i,j) \,:\, & i \in V_D, j \in V_O, \text{ or } i \in V_O, j \in V_O, i \neq j, \text{ or} \\ & i \in V_O, j \in V_D, \text{ or } i \in V_T, j \in V_O, \text{ or } i \in V_T, j \in V_D\} \end{aligned} \tag{3}$$

With each arc, and thus physical displacement, a travel time is associated as follows:

$$\tau_{ij} = \begin{cases} \tau(L_i, O_j), & \forall i \in V_D, j \in V_O \\ \tau(D_i, L_j), & \forall i \in V_O \cup V_T, j \in V_D \\ \tau(D_i, O_j), & \forall i \in V_O \cup V_T, j \in V_O, \phi_i^L = \phi_j^Q \\ \min\limits_{k \in V_D}(\tau(D_i, L_k) + \tau(L_k, O_j)), & \forall i \in V_O \cup V_T, j \in V_O, \phi_i^L \neq \phi_j^Q \end{cases} \tag{4}$$

Here, the fourth case corresponds to the case where either an empty container is released after order i, but order j does not require an empty container, meaning it first has to be dropped off at the depot k which minimizes the distance $i - k - j$; or similarly, if order i does not release an empty container, but j requires one, one should be picked up at the depot k which minimizes the distance $i - k - j$. In this way, the transport of empty containers is modeled.

In the model considered here, a truck appointment system (TAS) is incorporated at each of the container terminals. A distinction is made between import orders $V_O^I \subset V_O$ and export orders $V_O^E \subset V_O$, each having a separate TAS. Each terminal $m \in M$ has a corresponding set of time slots T_m. Each time slot $l \in T_m$ and terminal m has a corresponding number of available places for import Q_m^l, and export R_m^l. The width of the time slots is chosen to be the same for each terminal and is set to 2 h, i.e. the time slots are 0:00–2:00, 2:00–4:00, 4:00–6:00, 6:00–8:00, 8:00–10:00, 10:00–12:00, 12:00–14:00, 14:00–16:00, 16:00–18:00, 18:00–20:00, 20:00–22:00, 22:00–24:00. The number of available places in each time slot, of course, depends on the capacity and the number of places that are booked by other transport companies, which will be elaborated in more detail in Sect. 4.

In the formulation of the optimization model, a few different types of variables are used. The first type $y_i \in \mathbb{R}, \quad \forall i \in V_O$ represents the time at which the activities of order i are commenced. Next, the binary variables $x_{ij} \in \{0,1\}, \quad \forall (i,j) \in A$ denote which vertex j should succeed the activities in i. Note that the fleet is assumed to be homogeneous and there are no vehicle-specific constraints, such that a two-index formulation suffices and no three-index variables x_{ij}^k (with k denoting the specific vehicle) are necessary. Finally, two sets of binary variables are introduced in order to capture the TAS functionality:

$$q_{il} = \begin{cases} 1, \text{ if } l \in T_{m(i)} \text{ is booked in the import TAS of terminal } m(i) \text{ for import order } i \\ 0, \text{ otherwise} \end{cases}$$

$$r_{il} = \begin{cases} 1, \text{ if } l \in T_{m(i)} \text{ is booked in the export TAS of terminal } m(i) \text{ for export order } i \\ 0, \text{ otherwise} \end{cases}$$

where $m(i)$ denotes the terminal corresponding to order i for import and export orders.

3.2 Stochasticity

The model formulated and studied in this work will incorporate uncertainty and stochasticity of both travel times and loading and unloading times. First of all the probability distribution of the loading and unloading time will depend on the attributes ϕ_i^Q and ϕ_i^L and on the location (terminal or customer), $p_i^{O/D}(t_i|\phi_i^{Q/L})$, see Sect. 4. These distributions will however be assumed to be unconditioned on the time of arrival. The travel time between locations will be assumed to be distributed according to a given probability distribution, conditioned upon the time of departure $p_\tau(\tau|T)$. The travel time described in (4) will thus be distributed accordingly

$$\tau_{ij} = \begin{cases} \tau(L_i, O_j) \sim p_1(\tau_{ij}|s), & \forall i \in V_D, j \in V_O \\ \tau(D_i, L_j) \sim p_2(\tau_{ij}|y_i), & \forall i \in V_O \cup V_T, j \in V_D \\ \tau(D_i, O_j) \sim p_2(\tau_{ij}|y_i), & \forall i \in V_O \cup V_T, j \in V_O, \phi_i^L = \phi_j^Q \\ \min\limits_{k \in V_D}(\tau(D_i, L_k) + \tau(L_k, O_j)) \sim p_2(\tau_{ij}|y_i), & \forall i \in V_O \cup V_T, j \in V_O, \phi_i^L \neq \phi_j^Q \end{cases} \tag{5}$$

Here $p_1(\tau_{ij}|s) = p_\tau(\tau_{ij}|s)$ for the first case, since departure from a depot takes place at time s, for a decision epoch at time s. For the second, third, and fourth cases, the time of departure from the destination D_i of order i depends on the time it takes to complete the other activities of order i. Let us therefore first compute the distribution of this departure time, given the activities of order i are initiated at time y_i. The first activity is the loading of a container at the origin O_i, for which the time needed to complete this, t_i^O, is distributed as $p_i(t_i^O|\phi_i^Q)$. The next activity is the trip from O_i to D_i, with departure time $y_i + t_i^O$, with travel time τ_i. The distribution of $t_i^{O\tau} = t_i^O + \tau_i$ is then given by

$$p^{O\tau}(t_i^{O\tau}|y_i) = \int p_i(t_i^O|\phi_i^Q)p_\tau(t_i^{O\tau} - t_i^O|y_i + t_i^O)dt_i^O \tag{6}$$

The following activity is the unloading of the container at the destination D_i, the time for which, t_i^D, is distributed as $p_i(t_i^D|\phi_i^L)$. The probability distribution of the total time of all activities in order i, $t_i = t_i^{O\tau} + t_i^D$ is then given by

$$p^{O\tau D}(t_i|y_i) = \int p^{O\tau}(t_i^{O\tau}|y_i)p_i^D(t_i - t_i^{O\tau}|\phi_i^L)dt_i^{O\tau} \tag{7}$$

Finally, the distribution of the travel time between i and j is given by

$$p_2(\tau_{ij}|y_i) = \int p_\tau(\tau_{ij}|y_i + t_i)p^{O\tau D}(t_i|y_i)dt_i \tag{8}$$

In the optimization model, the probability distribution of the sum $t_{ij} = t_i + \tau_{ij}$ will be needed, which can be computed in the following way

$$p_{ij}(t_{ij}|y_i) = \int p_\tau(t_{ij} - t_i|y_i + t_i)p^{O\tau D}(t_i|y_i)dt_i \tag{9}$$

3.3 Optimization model: MINLP

The objective of the optimization model will be to minimize the expected value of the total operation time needed to complete all orders. The complete optimization problem is given below and is a mixed integer non-linear program (MINLP).

$$\begin{aligned}\min \Bigg[& \sum_{i\in V_O}\sum_{j\in V_D}(y_i + \mathbb{E}[t_i + \tau_{ij}])x_{ij} - \sum_{i\in V_D}\sum_{j\in V_O}(y_j - \mathbb{E}[\tau_{ij}])x_{ij} \\ & + \sum_{i\in V_T}\sum_{j\in V_D}(y_i + \mathbb{E}[t_i + \tau_{ij}])x_{ij} - \sum_{i\in V_T}\sum_{j\in V_O\cup V_D} y_i x_{ij}\Bigg]\end{aligned} \tag{10a}$$

$$\text{Subject to} \sum_{j\in V_O} x_{ij} \le n_i, \quad \forall i \in V_D \tag{10b}$$

$$\sum_{j\in V_O\cup V_D} x_{ij} = 1, \quad \forall i \in V_T \tag{10c}$$

$$\sum_{j \in V_O \cup V_D \cup V_T} x_{ji} = \sum_{j \in V_O \cup V_D} x_{ij} = 1, \quad \forall i \in V_O \tag{10d}$$

$$A_i \leq y_i \leq B_i, \quad \forall i \in V_O \tag{10e}$$

$$P(x_{ij}(y_i + t_i + \tau_{ij}) \leq y_j | y_i) \geq (1 - \alpha), \quad \forall i,j \in V_O \tag{10f}$$

$$P(x_{ij}(y_i + t_i + \tau_{ij}) \leq y_j | y_i) \geq (1 - \alpha), \quad \forall i \in V_T, \quad \forall j \in V_O \tag{10g}$$

$$P(x_{ij}(s + \tau_{ij}) \leq y_j | s) \geq (1 - \beta), \quad \forall i \in V_D, \quad \forall j \in V_O \tag{10h}$$

$$y_i q_{il} \leq U_l^{m(i)}, \quad \forall i \in V_O^I, \quad \forall l \in T_{m(i)} \tag{10i}$$

$$y_i \geq q_{il} L_l^{m(i)}, \quad \forall i \in V_O^I, \quad \forall l \in T_{m(i)} \tag{10j}$$

$$P(L_l^{m(i)} r_{il} \leq r_{il}(y_i + t_i^O + \tau_i) \leq U_l^{m(i)} | y_i) \geq (1 - \epsilon), \quad \forall i \in V_O^E, \quad \forall l \in T_{m(i)} \tag{10k}$$

$$\sum_{l \in T_{m(i)}} q_{il} = 1, \quad \forall i \in V_O^I \tag{10l}$$

$$\sum_{l \in T_{m(i)}} r_{il} = 1, \quad \forall i \in V_O^E \tag{10m}$$

$$\sum_{\substack{i \in V_O^I \\ m(i) = h}} q_{il} \leq Q_l^h, \quad \forall l \in T_h, \quad \forall h \in M \tag{10n}$$

$$\sum_{\substack{i \in V_O^E \\ m(i) = h}} r_{il} \leq R_l^h, \quad \forall l \in T_h, \quad \forall h \in M \tag{10o}$$

$$x_{ij} \in \{0, 1\}, \quad i \neq j, \quad \forall i \in V_O \cup V_D \cup V_T, \quad \forall j \in V_O \cup V_D \tag{10p}$$

$$y_i \in \mathbb{R}, \quad \forall i \in V_O \tag{10q}$$

$$q_{il} \in \{0, 1\}, \quad \forall i \in V_O^I, \quad \forall l \in T_{m(i)} \tag{10r}$$

$$r_{il} \in \{0, 1\}, \quad \forall i \in V_O^E, \quad \forall l \in T_{m(i)} \tag{10s}$$

Let us first discuss the objective (10a) of minimizing the total operating time. The first line consists of two terms, the first of these represents the times at which trucks arrive back at a depot after finishing their orders, i.e. a truck departing from order i to depot j will arrive at time $y_i + \mathbb{E}[t_i + \tau_{ij}]$; the second term in this line denotes the departing times of all trucks, i.e. if a truck start an order j at time y_j, it should depart at time $y_j - \mathbb{E}[\tau_{ij}]$ at depot i. The difference of the time at which trucks return to a depot and the time at which they depart at a depot thus represents the total operation time, including waiting, loading, unloading, driving, etc. The second line in the objective is the total operating time for temporal vertices, i.e. they arrive at time $y_i + \mathbb{E}[t_i + \tau_{ij}]$ at a depot j and depart at time y_i.

The basic constraints (10b), (10c), and (10d) ensure that at most n_i trucks can depart at a given depot i, each temporal vertex must be left, and every order vertex must have exactly one incoming and one outgoing truck, respectively. Next, the starting time should respect the time windows (10e), and the probability of arriving before the starting time of the next order y_j should meet at least some threshold (10f), (10g), (10h), where $\alpha, \beta \in [0, 1]$ are some user-defined parameters. Note that these constraints also eliminate subtours among orders. These probabilities can be easily computed by integrating (9) with the correct corresponding bounds. The next set of constraints implements the TAS functionality. For any import orders, the starting time of an order should meet the time slot booked in the TAS (10i) and (10j), and for any export orders, the probability of meeting the booked time slot should be at least a certain threshold value (10k), with $\epsilon \in [0, 1]$ a user-defined parameter. For every import or export order, exactly one time slot should be booked, (10l) and (10m). And, for every terminal, the number of booked places in each time slot for import and export should not exceed the number of available places in that time slot, (10n) and (10o). Finally, (10p), (10q), (10r), and (10s) define the variable domains.

This optimization problem is a mixed integer problem and is also non-linear due to the objective (10a) and constraints (10f), (10g), (10h), (10i) and (10k). In the next subsection, a framework which fully linearizes this optimization problem will be discussed.

3.4 ILP: Window partitioning

In order to tackle the problem with very efficient existing ILP solvers, it is translated into an easier (but still NP-hard) integer linear program. This will allow us to solve problem instances with a considerable amount of orders, as will be demonstrated in the experiments. The optimization model of the previous subsection will be reformulated and approximated based on the discretization of the continuous time variable and time windows, $y_i \in [A_i, B_i]$. Consider order i and its time window $[A_i, B_i]$, a discretization width δ is chosen and the time window is replaced by a set of smaller time windows

$$[A_i, B_i] \to [A_i, A_i + \delta], [A_i + \delta, A_i + 2\delta], \cdots [A_i + (n-1)\delta, B_i] \tag{11}$$

where $n = \lfloor (B_i - A_i)/\delta \rfloor$. Each order vertex is replaced by a set of suborder vertices. Each suborder has a certain smaller time window $[a_i, b_i]$, which represents the discrete starting time of that suborder by only considering the latest time at which that suborder should be commenced, b_i. Introducing the set of all suborders ω, the respective sets for import and export, ω_I and ω_E, and replacing V_O by ω in the set of arcs A and in the definition of the variables x_{ij}, as well as replacing V_T by ω_T, the stochastic travel time becomes:

$$\tau_{ij} = \begin{cases} \tau(L_i, O_j) \sim p_1(\tau_{ij}|s), & \forall i \in V_D, j \in \omega \\ \tau(D_i, L_j) \sim p_2(\tau_{ij}|b_i), & \forall i \in \omega \cup \omega_T, j \in V_D \\ \tau(D_i, O_j) \sim p_2(\tau_{ij}|b_i), & i \in \omega \cup \omega_T, j \in \omega, \phi_i^L = \phi_j^Q \\ \min_{k \in V_D}(\tau(D_i, L_k) + \tau(L_k, O_j)) \sim p_2(\tau_{ij}|b_i), & i \in \omega \cup \omega_T, j \in \omega, \phi_i^L \neq \phi_j^Q \end{cases} \tag{12}$$

By substituting the decision variable y_i by the parameter b_i, the optimization problem from Sect. 3.3 can thus be reformulated as (where the indicator $o(i) \in V_O, \quad \forall i \in \omega$ denotes the order to which a suborder belongs):

$$\min \Bigg[\sum_{i \in \omega} \sum_{j \in V_D} (b_i + \mathbb{E}[t_i + \tau_{ij}]) x_{ij} - \sum_{i \in V_D} \sum_{j \in \omega} (b_j - \mathbb{E}[\tau_{ij}]) x_{ij} + \sum_{i \in \omega_T} \sum_{j \in V_D} (b_i + \mathbb{E}[t_i + \tau_{ij}]) x_{ij} - \sum_{i \in \omega_T} \sum_{j \in \omega \cup V_D} b_i x_{ij} \Bigg] \tag{13a}$$

$$\text{Subject to} \sum_{j \in \omega} x_{ij} \leq n_i, \quad \forall i \in V_D \tag{13b}$$

$$\sum_{j \in \omega \cup V_D} x_{ij} = 1, \quad \forall i \in \omega_T \tag{13c}$$

$$\sum_{j \in \omega \cup V_D \cup \omega_T} x_{ji} = \sum_{j \in \omega \cup V_D} x_{ij}, \quad \forall i \in \omega \tag{13d}$$

$$\sum_{i \in \omega \cup V_D \cup \omega_T} \sum_{\substack{j \in \omega \\ o(j) = k}} x_{ij} = 1, \quad \forall k \in V_O \tag{13e}$$

$$P(b_i + t_i + \tau_{ij} \leq b_j | b_i) \geq (1 - \alpha) x_{ij}, \quad \forall i,j \in \omega \tag{13f}$$

$$P(b_i + t_i + \tau_{ij} \leq b_j | b_i) \geq (1-\alpha)x_{ij}, \quad \forall i \in \omega_T, \quad \forall j \in \omega \tag{13g}$$

$$P(s + \tau_{ij} \leq b_j | s) \geq (1-\beta)x_{ij}, \quad \forall i \in V_D, \quad \forall j \in \omega \tag{13h}$$

$$b_i q_{il} \leq U_l^{m(i)}, \quad \forall i \in \omega_I, \quad \forall l \in T_{m(i)} \tag{13i}$$

$$b_i \geq q_{il} L_l^{m(i)}, \quad \forall i \in \omega_I, \quad \forall l \in T_{m(i)} \tag{13j}$$

$$P(L_l^{m(i)} \leq b_i + t_i^O + \tau_i \leq U_l^{m(i)} | b_i) \geq (1-\epsilon) r_{il}, \quad \forall i \in \omega_E, \quad \forall l \in T_{m(i)} \tag{13k}$$

$$\sum_{\substack{i \in \omega \\ o(i) = k}} \sum_{l \in T_{m(i)}} q_{il} = 1, \quad \forall k \in V_O^I \tag{13l}$$

$$\sum_{\substack{i \in \omega \\ o(i) = k}} \sum_{l \in T_{m(i)}} r_{il} = 1, \quad \forall k \in V_O^E \tag{13m}$$

$$\sum_{\substack{i \in \omega_I \\ m(i) = h}} q_{il} \leq Q_l^h, \quad \forall l \in T_h, \quad \forall h \in M \tag{13n}$$

$$\sum_{\substack{i \in \omega_E \\ m(i) = h}} r_{il} \leq R_l^h, \quad \forall l \in T_h, \quad \forall h \in M \tag{13o}$$

$$x_{ij} \in \{0,1\}, \quad i \neq j, \quad \forall i \in \omega \cup V_D \cup \omega_T, \quad \forall j \in \omega \cup V_D \tag{13p}$$

$$q_{il} \in \{0,1\}, \quad \forall i \in \omega_I, \quad \forall l \in T_{m(i)} \tag{13q}$$

$$r_{il} \in \{0,1\}, \quad \forall i \in \omega_E, \quad \forall l \in T_{m(i)} \tag{13r}$$

The new objective (13a) is the straightforward linearization of (10a). Constraints (13b), (13c), and (13d) are the equivalents of constraints (10b), (10c), and (10d), while constraint (13e) enforces that for every order, exactly one suborder vertex should be visited. Constraints (13f), (13g) and (13h) are the linear versions of (10f), (10g) and (10h). The TAS constraints (13i), (13j), (13k), (13l), (13m), (13n), and

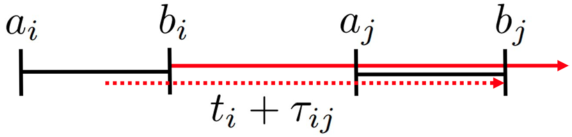

Fig. 2 An example of a feasible movement/arc in the original model (dashed arrow) which becomes infeasible due to the window partitioning (solid arrow)

(13o) are also the direct translation of (10i), (10j), (10k), (10l), (10m), (10n), and (10o). Finally (13p), (13q), and (13r) define the variables.

The obtained model is a pure integer linear program (ILP) which is an over-constrained version of the original problem, since some feasible movements/arcs are eliminated by constraints (13f), (13g), see Fig. 2. The optimal solution of the new model thus results in a suboptimal solution (i.e. an upper bound) of the original MINLP. The approximation will, however, approach the optimal solution for a small enough discretization δ.

4 Experimental setup and test case

As already mentioned earlier, the experiments and test cases will be based on the port of Antwerp, Belgium. The port of Antwerp is a seaport located centrally in Europe, near the city of Antwerp, is Europe's second-largest seaport and is the 14th largest container port worldwide. The port generates an added value of roughly 22 billion euros, i.e. about 4.1% of Belgian GDP (Rubbrecht 2022). In 2022, 13 484 122 TEU (twenty-foot equivalent unit) of containers were handled in the port of Antwerp, closely following the largest port in Europe, namely the port of Rotterdam with 14 455 000 TEU handled in 2022. Below, the test case will be discussed in more detail.

4.1 Terminals, customers and depots

The port of Antwerp has 5 main container terminals, see Fig. 3. In Table 2 the terminals are listed with their respective annual capacities. When generating orders for the experiments, it is assumed that 45% are import, 45% are export, and 10% are transports between customer nodes. Moreover, for import and export orders, the container terminal is chosen in a random fashion, weighted by the respective capacities.

As described earlier, each of the terminals employs a TAS for import and one for export. Each terminal is assumed to have the same time slots. In order to model the number of available places in each time slot at each terminal, the number of available places are weighted by the capacities of the terminals, and reduced by an amount proportional to the typical arrival rate at each time, see

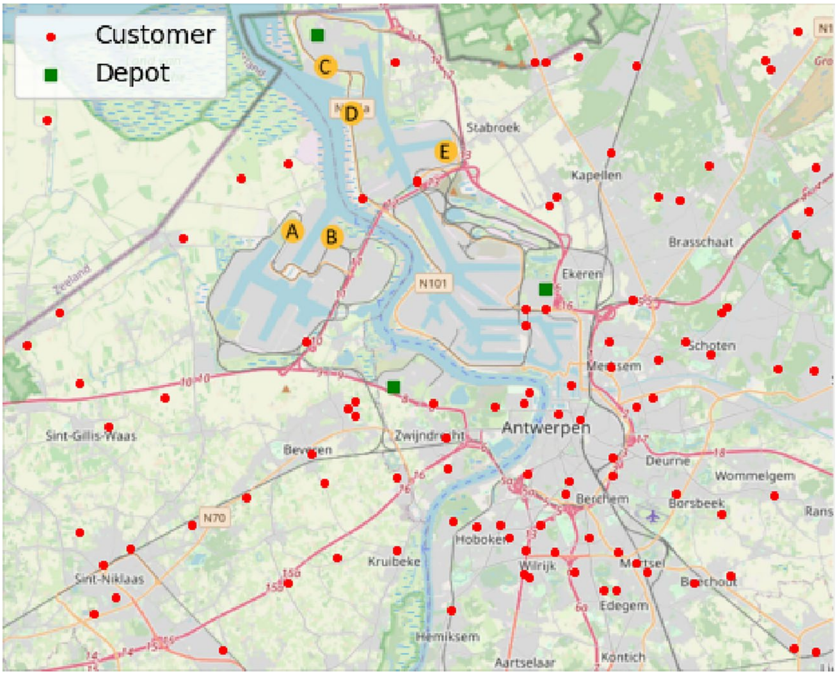

Fig. 3 The port of Antwerp and its hinterland, along with the locations of the 5 main container terminals

Table 2 The 5 container terminals in the port of Antwerp, and their annual capacities

Label in Fig. 3	Terminal	Capacity (TEU)
A	MSC PSA European Terminal (MPET)	9,000,000 (53%)
B	DP World Antwerp Gateway Terminal	2,500,000 (15%)
C	PSA Antwerp Noordzee Terminal	2,600,000 (15%)
D	PSA Antwerp Europa Terminal	1,800,000 (11%)
E	Antwerp Container Terminal	1,000,000 (6%)

Fig. 4. The resulting available places used in the experiments (import + export) are summarized in Table 3.

Customer vertices are sampled randomly in the area of study, 100 such vertices are generated, see Fig. 3. In generating orders, the customers are picked randomly from this set.

The transport company considered in the experiments is assumed to own and manage 3 different depots at which trucks and empty containers can be stored, their locations are depicted in Fig. 3 as well. For all experiments, the number of trucks is assumed to be distributed equally over the different depots.

The travel time matrix between all vertices is computed based on the underlying traffic network. To this end data from OpenStreetMap (2020) was used.

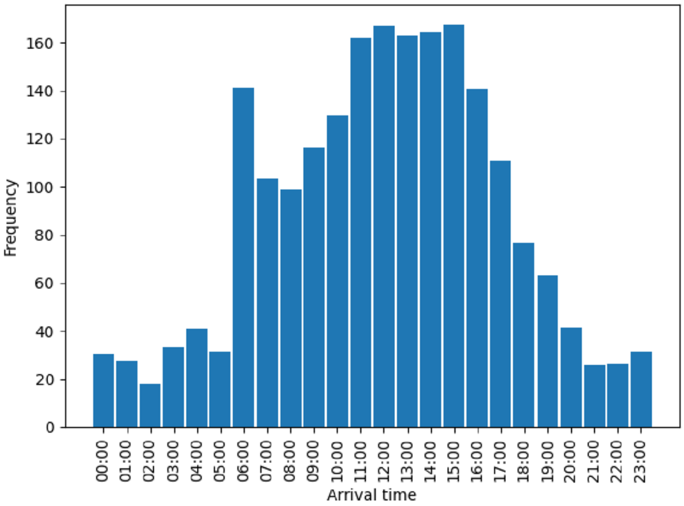

Fig. 4 The typical arrival frequency at a container terminal

Table 3 The number of available places in the TAS for each time slot, for the different container terminals

Terminal	00:00	02:00	04:00	06:00	08:00	10:00	12:00	14:00	16:00	18:00	20:00	22:00
A	171	175	164	78	93	55	35	35	75	131	167	172
B	48	49	46	22	26	16	10	10	21	37	47	48
C	50	51	48	23	27	16	11	10	22	38	49	50
D	35	35	33	16	19	11	7	7	15	27	34	35
E	19	20	19	9	11	7	4	4	9	15	19	20

4.2 Constructing probability distributions

4.2.1 Travel time

In order to estimate the conditional probability distribution of travel times, historic GPS data of trucks in Belgium was used. Since no trip-specific data is available for every possible route, a "global" distribution of the delay on the road network is constructed, averaged over all road segments. To this end a network of origins o and destinations d that appear in the historic data is constructed, and for each pair the travel times are scaled by the minimum value that appears for that specific (o, d) pair, $\theta = t_{od}/t_{od}^{\min}$, as a function of the departure time T. This allows us to consider all (o, d) pairs together, and construct a global distribution $p_\theta(\theta|T)$. To construct this global distribution, for each departure time T, a weighted Gaussian KDE of all data points was used, with the weights given by

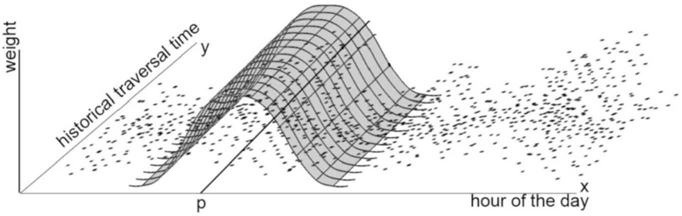

Fig. 5 Illustration of the weighing in the KDE of $p_\theta(\theta|T)$

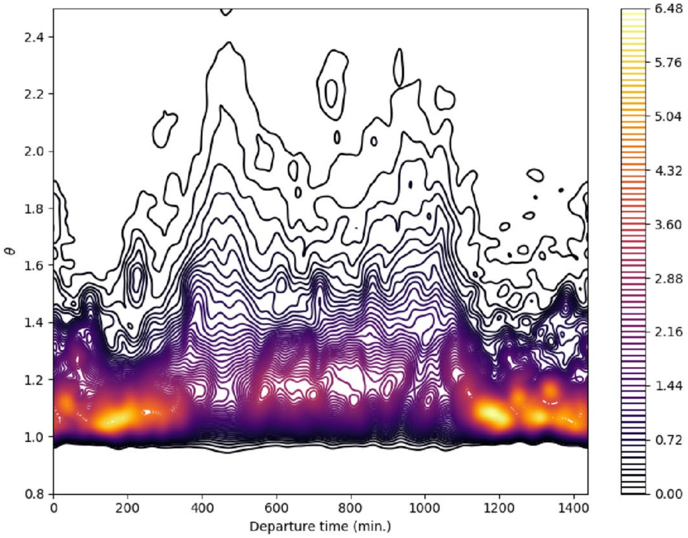

Fig. 6 The obtained estimator for $p_\theta(\theta|T)$

$$w(\theta_i) = e^{-\frac{1}{2}\left(\frac{T^i-T}{\sigma_w}\right)^2} \quad \forall\, (T^i, \theta_i) \tag{14}$$

where (T^i, θ_i) denotes an individual data point i with departure time T^i and delay factor θ_i. The parameter σ_w was set to 30 min. Figure 5 provides an illustration of the KDE weights. Putting this all together an estimator for the distribution $p_\theta(\theta|T)$ is obtained, depicted in Fig. 6. In Fig. 7 the mean of $p_\theta(\theta|T)$ is given for each T, from which one can clearly see 2 distinctive peaks corresponding to peak hours in traffic. Note that the increase in the mean around the morning and evening is not so much due to the peak of $p_\theta(\theta|T)$ shifting for each T, but due to the tail of the distribution reaching much further at these moments, as can be seen in Fig. 6, meaning there is way more variation or uncertainty about the travel time at these moments.

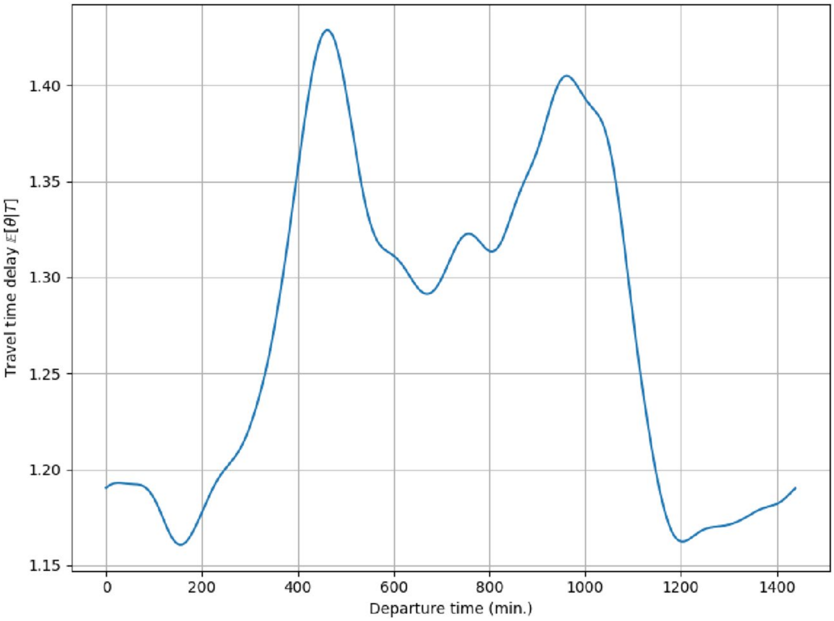

Fig. 7 The mean of $\theta \sim p_\theta(\theta|T)$

Using the global delay distribution $p_\theta(\theta|T)$, the travel time distributions of specific (o, d) pairs within the road network can be approximated by scaling this distribution by the minimal possible travel time between this origin and destination. For example

$$p_\tau(\tau_{ij}|T) = p_\theta(\tau_{ij}/\tau^0_{ij}|T) \tag{15}$$

Note that in general, the delay or travel time distribution should not only be dependent on the departure time, but also on the specific trajectory or the roads within it. Here, however, longer trajectories are considered, for which one can assume that the localized effects of specific roads within a trajectory get averaged out. This in turn justifies approximating the travel time distribution of a given trajectory by rescaling the global averaged delay distribution.

4.2.2 Turnaround time at container terminals

The probability distribution of the turnaround time (both import and export) at container terminals was determined from historical data from a terminal in the port of Antwerp, averaged over a typical day. The resulting distribution is depicted in Fig. 8.

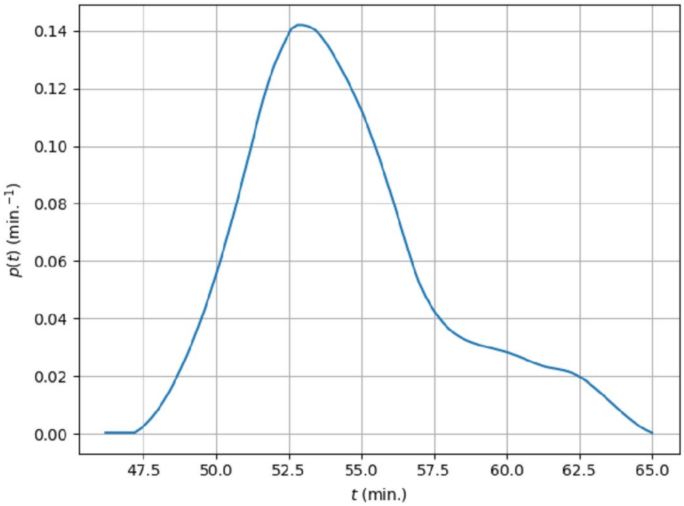

Fig. 8 Distribution of the truck turnaround time at a container terminal

4.2.3 Handling time at customer

The (un)loading time at the customer vertices is assumed to be distributed according to a log-normal distribution, since no quantitative data is available:

$$\begin{aligned} p_h(t_h) &= \frac{1}{t_h\sigma\sqrt{2\pi}} e^{-\frac{(\ln t_h-\mu)^2}{2\sigma^2}} \\ \mu &= \ln\left(\frac{\mu_h^2}{\sqrt{\mu_h^2+\sigma_h^2}}\right), \quad \sigma = \ln\left(1+\frac{\sigma_h^2}{\mu_h^2}\right) \end{aligned} \tag{16}$$

Both the mean and standard deviation are assumed to depend on the attributes ϕ^Q and ϕ^L:

$$(\mu_h, \sigma_h) = \begin{cases} (30\text{ min}, 10\text{ min}) & \phi^{Q/L} = 0 \\ (60\text{ min}, 15\text{ min}) & \phi^{Q/L} = 1 \end{cases} \tag{17}$$

5 Results

The linearized optimization model is solved using the commercial solver Gurobi Optimization (2022). All experiments were performed on a computer with an Intel Core i7-8650U CPU @ 1.90GHz×8 processor and 16 GB of RAM, under Ubuntu 18.04 x64.

Table 4 Solutions for different problem sizes, where $\delta = 10$ min, $\alpha = \beta = 0.1$ and $\epsilon = 0.2$

# Orders	# Trucks	Comp. time (s)	Objective (min.)	Minutes per order
10	2	0.009	–	–
10	3	0.06	1699	169.9
10	4	0.06	1699	169.9
20	4	0.26	–	–
20	6	0.26	3046	152.3
20	8	0.26	3046	152.3
50	10	6.2	7989	159.8
50	15	2.4	7720	154.4
50	20	3.6	7642	152.8
100	20	95	15,334	153.3
100	30	23	14,958	149.6
100	40	16	14,762	147.6
150	30	256	22,046	147.0
150	45	146	21,718	144.8
150	60	84	21,494	143.3
200	40	574	29,288	146.4
200	60	329	28,871	144.4
200	80	452	28,646	143.2
250	50	1396	36,411	145.6
250	75	474	35,865	143.5
250	100	416	35,578	142.3
300	60	2213	43,383	144.6
300	90	1299	42,871	142.9
300	120	775	42,612	142.0

When no feasible solution exists, this is denoted by "–"

5.1 Varying instance size

Let us first consider problem instances of varying numbers of orders and trucks, with a fixed planning without dynamic orders. Setting $\delta = 10$ min, $\alpha = \beta = 0.1$ and $\epsilon = 0.2$, the results are summarized in Table 4, where the computing time, the objective value, and the operating time (in minutes) per order is given for the optimal solutions. We were able to solve problems of substantial size in less than an hour, which is sufficient for a daily static planning. Moreover, for average-sized problems of 150 orders, a solution can be found in under 5 min. As a general trend, the computing time decreases with an increase in the number of available trucks, for a given number of orders. The reason for this is that for a smaller number of trucks, the constraints are tighter and more orders have to be combined in larger tours. It is also clear that the operating time per order decreases for an increasing number

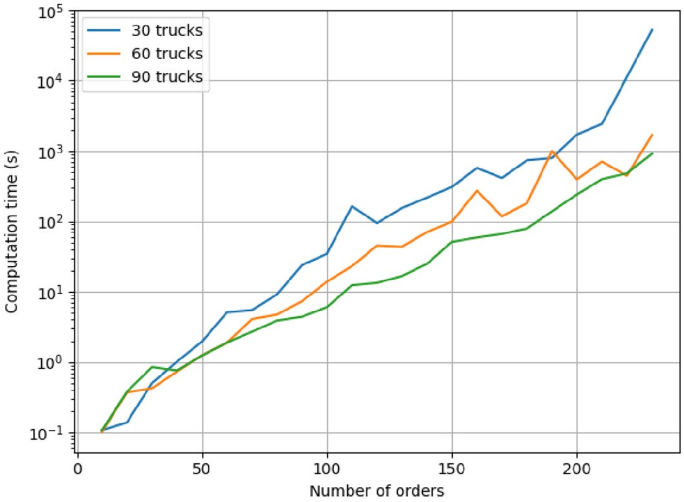

Fig. 9 Computation time for different amounts of orders

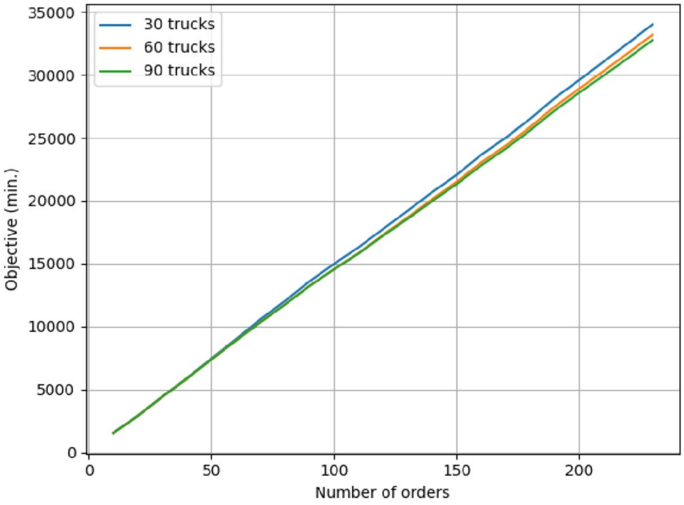

Fig. 10 Objective value of the optimal solution for different amounts of orders

of orders and trucks, which can be expected since this results in a greater solution space meaning further optimization is possible.

Figures 9, 10 and 11 depict the computation time, objective value and operating time per order, respectively, for a varying number of orders, further illustrating these trends. In Fig. 9, the computation time is on average a factor 2–3 higher for 30 trucks compared to 60 trucks, and a factor of 1.5–2 for 60 trucks compared to 90. The operating time per order in Fig. 11 is equal for small orders set sizes, since 30 trucks is excessive

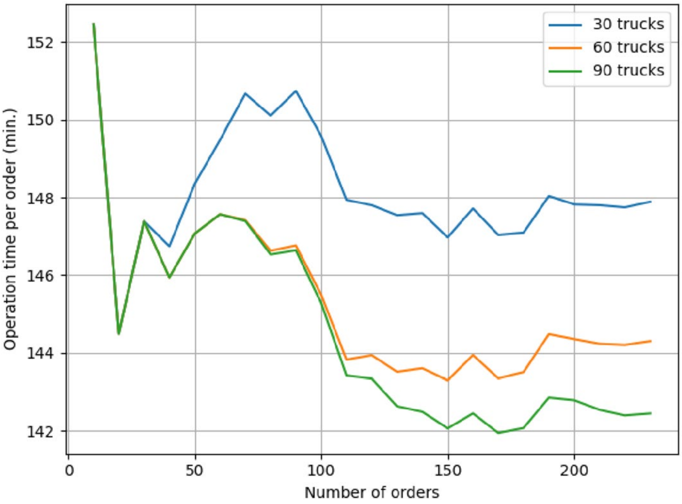

Fig. 11 Operation time per order in the optimal solution for different amounts of orders

and not all trucks are used. On the other hand, for a greater number of orders, the operating time per order is about 1.9% higher in the case of 60 trucks compared to the case of 90 trucks. The effect of increasing the number of trucks from 60 to 90 is smaller, since the number of tours that is impacted or can be split up is smaller (all trucks were used in all three cases for order set sizes equal to or greater than 160).

5.2 Stochastic versus deterministic

In order to validate the stochastic model presented in this work, let us compare it to a deterministic model. The deterministic model is similar to model (13a)–(13r), but the chance constraints (13f), (13g), (13h), and (13k) are changed to their deterministic counterparts:

$$(b_i + \mathbb{E}[t_i] + \mathbb{E}[\tau_{ij}])x_{ij} \leq b_j, \quad \forall i,j \in \omega \tag{18a}$$

$$(b_i + \mathbb{E}[t_i] + \mathbb{E}[\tau_{ij}])x_{ij} \leq b_j, \quad \forall i \in \omega_T, \quad \forall j \in \omega \tag{18b}$$

$$(s + \mathbb{E}[\tau_{ij}])x_{ij} \leq b_j, \quad \forall i \in V_D, \quad \forall j \in \omega \tag{18c}$$

$$L_l^{m(i)} r_{il} \leq b_i + \mathbb{E}[t_i^O] + \mathbb{E}[\tau_i], \quad \forall i \in \omega_E, \quad \forall l \in T_{m(i)} \tag{18d}$$

Table 5 Percentage of feasible instances in Monte Carlo simulations for the deterministic and the stochastic models

# Orders	# Trucks	Deterministic	Stochastic
10	2	–	–
10	3	99.3%	100.0%
10	4	99.6%	100.0%
20	4	65.1%	–
20	6	90.7%	99.0%
20	8	92.2%	99.5%
50	10	35.5%	94.3%
50	15	72.5%	95.6%
50	20	58.8%	96.6%
100	20	8.4%	82.1%
100	30	11.0%	92.4%
100	40	19.0%	95.9%
150	30	19.1%	95.4%
150	45	6.3%	95.0%
150	60	10.8%	99.3%
200	40	11.4%	89.7%
200	60	13.2%	92.7%
200	80	19.1%	91.1%
250	50	2.5%	86.9%
250	75	4.7%	90.7%
250	100	14.4%	92.5%
300	60	1.3%	82.4%
300	90	1.1%	82.6%
300	120	1.4%	90.9%

When no feasible solution exists, this is denoted by "–"

$$(b_i + \mathbb{E}\,[t_i^O] + \mathbb{E}\,[\tau_i])r_{il} \leq U_l^{m(i)}, \quad \forall i \in \omega_E, \quad \forall l \in T_{m(i)} \tag{18e}$$

As a means of testing the feasibility and robustness of a solution under uncertain process times, a number of Monte Carlo simulations will be performed, where each random variable is drawn from its respective probability distribution. In a solution, each tour is simulated by generating random variables for the different steps in the tour, i.e. traveling from a depot to the first customer, loading the container, traveling to the destination, etc., advancing the time. For each time window, it is checked whether the current time falls in this window; if it is too early, the time is set to the lower bound of the time window; when it is too late, the solution is marked as failed or infeasible for that particular generated instance.

In Table 5 a comparison is made between the deterministic model and the stochastic model ($\delta = 10$ min, $\alpha = \beta = 0.1$ and $\epsilon = 0.2$), in terms of the number of feasible instances for the optimal solution found with the respective model. The instances are generated by Monte Carlo simulations, each solution was tested 1000 times. First of all, the impact of the stochastic model is very clear. The probability of the solutions

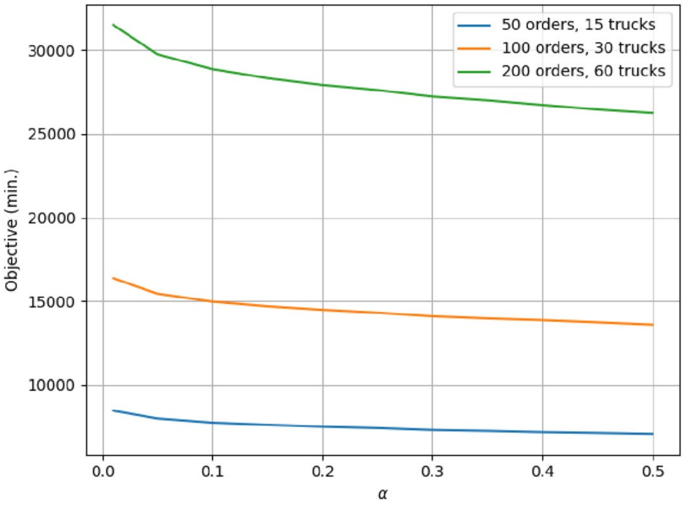

Fig. 12 Objective value of the optimal solution for values of α (= β) ranging from 0.01 to 0.5

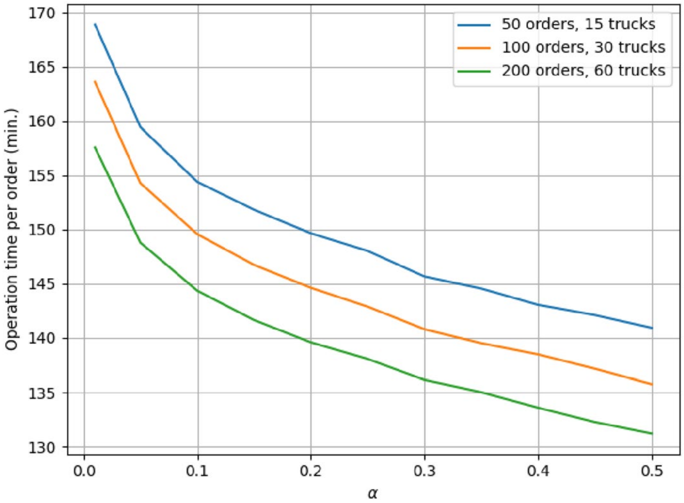

Fig. 13 Operation time per order in the optimal solution for values of α (= β) ranging from 0.01 to 0.5

found with the deterministic model failing due to varying processing times is significantly higher than for the solutions found with the stochastic model, especially for larger instances. On the other hand, the probability of the solutions found with the stochastic model being feasible under randomly sampled instances is very high, decreasing slightly with increasing instance sizes, as can be expected since the probability of something going wrong somewhere in the planning is higher.

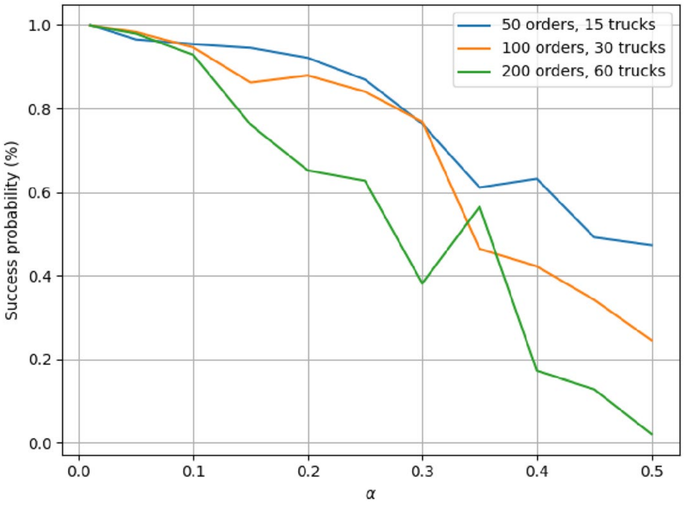

Fig. 14 Probability of the optimal solution being feasible for values of α (= β) ranging from 0.01 to 0.5

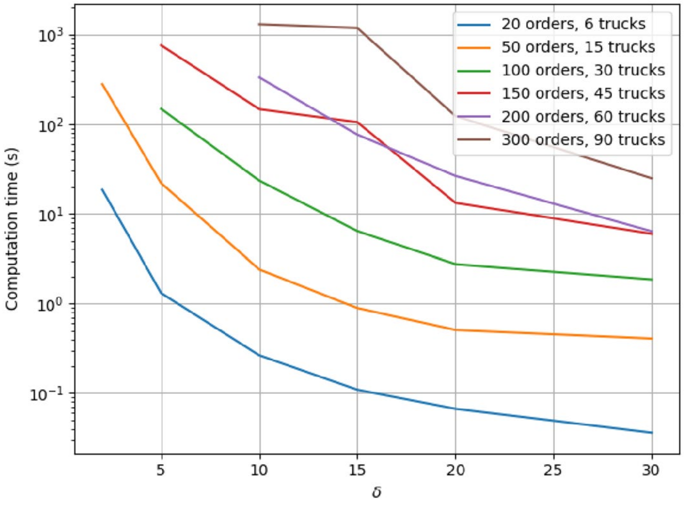

Fig. 15 Computation time for varying values of δ

5.3 Influence of α, β and δ

Let us now consider the influence of the confidence-parameters α and β in the chance constraints on the obtained solutions. Let us set $\beta = \alpha$ in the remainder of this subsection. Figure 12 depicts the objective value for the optimal solutions found for different values of α ranging from 0.01 to 0.5 (setting α to even lower values is futile, since one needs very good knowledge of the tails of the distributions in this case). The other parameters are set to $\delta = 10$ min and $\epsilon = 0.2$. When making the

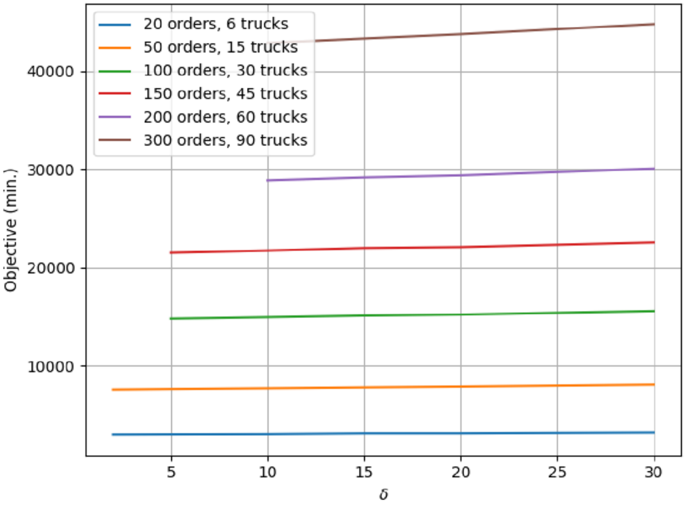

Fig. 16 Objective value of the optimal solution for different values of δ

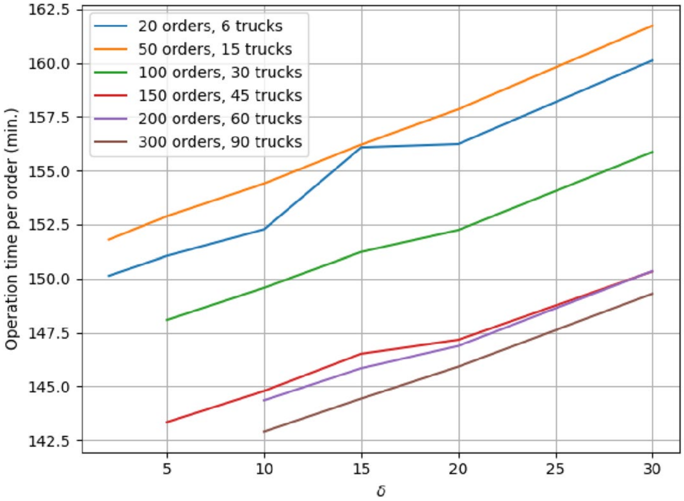

Fig. 17 Operation time per order in the optimal solution for different values of δ

chance constraints less strict, i.e. increasing α, the obtained objective (total operating time) is lower, as is to be expected, the relative change in the objective value is however limited. In Fig. 13 the operating time per order is given for different α -values, clearly illustrating the influence of α. Comparing the two extreme values of $\alpha = 0.01$ and $\alpha = 0.5$, the relative difference in the operating time is approximately 16%. In Fig. 14 the probability of the optimal solution being feasible under random realizations of the model, for varying values of α, is shown. It is clear that the

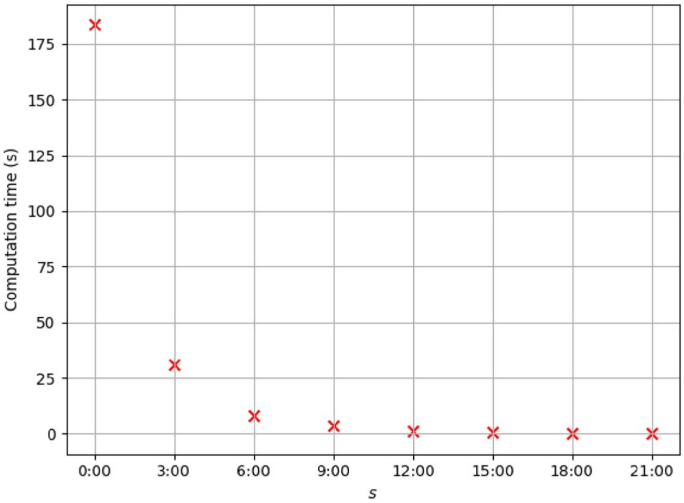

Fig. 18 Computation time and re-computation time for the dynamic instance at each epoch

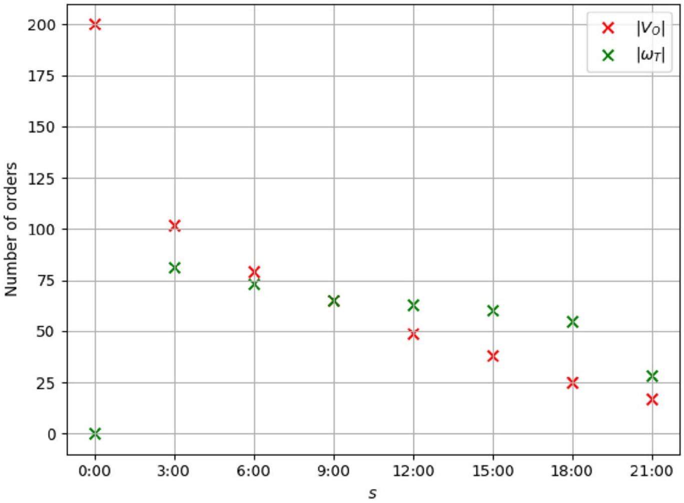

Fig. 19 Number of orders and temporal orders for the dynamic instance at each epoch

probability of a solution proving to be infeasible greatly increases when increasing α. We can conclude that a value of $\alpha = 0.1$ results in a good balance between the slight increase in operating times, and the success probability of the planning.

Another important parameter of the model is the discretization width δ. Figure 15 shows the computation time for different values of δ, where it is apparent that there is a strong increase in computation time for smaller δ-values. This is to be expected, as the number of variables in the ILP formulation scales as $\sim 1/\delta^2$. In Fig. 16 the optimal objective value is given for varying δ. Since the approximate model based on time window partitioning provides an upper bound to the exact model, the optimal objective value decreases with decreasing δ, however, the influence is limited. Figure 17 contains the operating time per order for varying δ, clearly illustrating the δ-dependence. Over the considered range of δ, the operating time varies only by approximately 5%. For the value of $\delta = 10$ min. used in the experiments before, the difference is only about 1.5%. From this one could conclude that by increasing δ, problems of far greater size can be solved efficiently, in exchange for only a small increase in the objective.

5.4 Dynamic orders

Finally, let us consider the dynamic aspect of the model. To this end, a given set of 200 orders is considered at the start of the day, for which an initial planning is computed. Next, a total of 7 decision epochs is considered, with times $s \in$ [3:00, 6:00, 9:00, 12:00, 15:00, 18:00, 21:00], where during each epoch, 12 new orders (with the lower bound of their time window, A_i, between s and 23:59) are added to the pool orders still to be initiated at that time, and the model is re-optimized. The number of trucks is set equal to 90, $\delta = 10$ min, $\alpha = \beta = 0.1$ and $\epsilon = 0.2$.

Figure 18 depicts the computation time needed to optimize and re-optimize the model at each decision epoch. The epoch at time 0:00 represents the initial planning for the initial 200 known orders. This initial planning takes longer to compute as the set of given orders is large. Each consecutive re-optimization however takes very little time, making it feasible to do in real-time. In Fig. 19 the number of orders in the order set V_O as well as the number of temporal orders (or suborders, since only one temporal suborder exists for each temporal order) is given for each epoch. The number of orders is steadily decreasing as more orders are executed as time passes. The number of temporal orders remains relatively constant during operations, since at a given time, the number of orders being processed at that time remains mostly the same, only to drop off near the end of the day.

6 Conclusion

In this work, a model for the dynamic stochastic container drayage problem with a truck appointment system operating at the different terminals is presented. Stochastic truck turnaround times at the terminals, loading and unloading times at the customers, and travel times conditioned upon the departure time, are incorporated in the form of conditional chance constraints. The general formulation results in a mixed integer nonlinear program, which we linearized by partitioning time windows and discretizing the time variable. The model is tested on instances based on a real-world case based in the port of Antwerp.

The experiments showed that the model is efficiently solvable, even for large instances of up to 300 orders. It was also illustrated that the obtained solutions are robust with respect to stochastic operating times. Based on Monte Carlo simulations, the probability of a solution or planning not failing was computed and was shown to remain high, even for large instances, while the solutions obtained with a deterministic model had a low probability of succeeding overall. By varying the confidence parameters in the chance constraints a trade-off can be made between robustness (i.e. probability of a planning succeeding) and minimizing the objective. We demonstrated that by lowering these parameters, a great increase in success probability can be obtained in exchange for only a limited increase in total operating times. We also showed that varying the time discretization width δ only had a minor impact on the resulting objective, but can greatly decrease computation times, which might be very useful when considering very large problem instances. Finally, it was demonstrated that in the case of flexible orders, the model can be re-optimized efficiently.

Future research might encompass extending the framework to include a live data stream of e.g. traffic information or delays at different terminals, updating the probability distributions accordingly. Another possible extension might be to make the discretization width variable instead of fixed, based on e.g. the time-sensitivity of certain distributions at certain time intervals, increasing efficiency.

Author contributions K. Stoop worked on the mathematical model, methodology, experiments, and the manuscript. M. Pickavet, D. Colle and P. Audenaert gave feedback on the used methods and formulations, and reviewed the manuscript.

Funding This research is made possible by the UGent grant BOF/STA/202009/039, Mobilidata, Imec COOCK, Imec Icon AI4FL and Imec Icon OptiRoutS.

Data availibility Part of the dataset generated during the current study is not publicly available due to confidential company data. Other data and information on how to obtain it and reproduce the analysis is available from the corresponding author upon reasonable request.

Declarations

Conflict of interest The authors have no conflict of interest to declare that are relevant to the content of this article.

References

Appelgren LH (1971) Integer programming methods for a vessel scheduling problem. Transp Sci 5(1):64–78

Birge JR, Louveaux F (2011) Introduction to stochastic programming. Springer, New York. https://doi.org/10.1007/978-1-4614-0237-4

Bjelić N, Vidović M, Popović D, Ratković B (2022) Rolling-horizon approach in solving dynamic multi-size multi-trailer container drayage problem. Expert Syst Appl 201:117170

Braekers K, Janssens GK, Caris A (2011) Challenges in managing empty container movements at multiple planning levels. Transp Rev 31(6):681–708

Chang H, Jula H, Chassiakos A, Ioannou P (2008) A heuristic solution for the empty container substitution problem. Transp Res Part E Logist Transp Rev 44(2):203–216

Cheikhrouhou O, Khoufi I (2021) A comprehensive survey on the multiple traveling salesman problem: applications, approaches and taxonomy. Comput Sci Rev 40:100369

Chen R, Chen S, Cui H, Meng Q (2021) The container drayage problem for heterogeneous trucks with multiple loads: a revisit. Transp Res Part E Logist Transp Rev 147:102241

Chen R, Jia S, Meng Q (2023) Dynamic container drayage booking and routing decision support approach for e-commerce platforms. Transp Res Part E Logist Transp Rev 177:103220

Escudero A, Muñuzuri J, Arango C, Onieva L (2011) A satellite navigation system to improve the management of intermodal drayage. Adv Eng Inform 25(3):427–434

Escudero A, Muñuzuri J, Guadix J, Arango C (2013) Dynamic approach to solve the daily drayage problem with transit time uncertainty. Comput Ind 64(2):165–175

Escudero-Santana A, Muñuzuri J, Cortés P, Onieva L (2021) The one container drayage problem with soft time windows. Res Transp Econ 90:100884

Gurobi Optimization, LLC: Gurobi Optimizer Reference Manual (2022). https://www.gurobi.com

Huynh N, Smith D, Harder F (2016) Truck appointment systems: where we are and where to go from here. Transp Res Rec 2548(1):1–9

Imai A, Nishimura E, Current J (2007) A Lagrangian relaxation-based heuristic for the vehicle routing with full container load. Eur J Oper Res 176(1):87–105

Jia S, Cui H, Chen R, Meng Q (2022) Dynamic container drayage with uncertain request arrival times and service time windows. Transp Res Part B Methodol 166:237–258

Jula H, Dessouky M, Ioannou P, Chassiakos A (2005) Container movement by trucks in metropolitan networks: modeling and optimization. Transp Res Part E Logist Transp Rev 41(3):235–259

Kuzmicz KA, Pesch E (2019) Approaches to empty container repositioning problems in the context of Eurasian intermodal transportation. Omega 85:194–213

Levin A (1971) Scheduling and fleet routing models for transportation systems. Transp Sci 5(3):232–255

Li P, Arellano-Garcia H, Wozny G (2008) Chance constrained programming approach to process optimization under uncertainty. Comput Chem Eng 32(1–2):25–45

Marković N, Drobnjak Ž, Schonfeld P (2014) Dispatching trucks for drayage operations. Transp Res Part E Logist Transp Rev 70:99–111

Messaoud E (2023) A chance constrained programming model and an improved large neighborhood search algorithm for the electric vehicle routing problem with stochastic travel times. Evol Intel 16(1):153–168

Namboothiri R, Erera AL (2008) Planning local container drayage operations given a port access appointment system. Transp Res Part E Logist Transp Rev 44(2):185–202

Nossack J, Pesch E (2013) A truck scheduling problem arising in intermodal container transportation. Eur J Oper Res 230(3):666–680

OpenStreetMap Contributors: Planet dump retrieved from https://planet.osm.org. https://www.openstreetmap.org (2020)

Oyola J, Arntzen H, Woodruff DL (2018) The stochastic vehicle routing problem, a literature review, part I: models. EURO J Transp Logist 7(3):193–221

Parragh SN, Doerner KF, Hartl RF (2008a) A survey on pickup and delivery problems: Part I: transportation between customers and depot. J Betriebswirtschaft 58:21–51

Parragh SN, Doerner KF, Hartl RF (2008b) A survey on pickup and delivery problems: Part II: transportation between pickup and delivery locations. J Betriebswirtschaft 58:81–117

Rubbrecht I (2022) Economic importance of the Belgian maritime and inland ports—report 2020. Technical report, NBB Working Paper

Shiri S, Huynh N (2016) Optimization of drayage operations with time-window constraints. Int J Prod Econ 176:7–20

Shiri S, Ng M, Huynh N (2019) Integrated drayage scheduling problem with stochastic container packing and unpacking times. J Oper Res Soc 70(5):793–806

Song Y, Zhang Y, Wang W, Xue M (2023) A branch and price algorithm for the drop-and-pickup container drayage problem with empty container constraints. Sustainability 15(7):5638

Spasovic LN, Morlok EK (1993) Using marginal costs to evaluate drayage rates in rail-truck intermodal service. Transp Res Rec 1383:8–16

Sterzik S, Kopfer H (2013) A Tabu search heuristic for the inland container transportation problem. Comput Oper Res 40(4):953–962

Torkjazi M, Huynh N, Shiri S (2018) Truck appointment systems considering impact to drayage truck tours. Transp Res Part E Logist Transp Rev 116:208–228

Toth P, Vigo D (2002) The vehicle routing problem. Society for Industrial and Applied Mathematics, Philadelphia. https://doi.org/10.1137/1.9780898718515

Toth P, Vigo D (2014) Vehicle routing. Society for Industrial and Applied Mathematics, Philadelphia. https://doi.org/10.1137/1.9781611973594

Wang X, Regan AC (2002) Local truckload pickup and delivery with hard time window constraints. Transp Res Part B Methodol 36(2):97–112

Yan X, Xu M, Xie C (2023) Local container drayage problem with improved truck platooning operations. Transp Res Part E Logist Transp Rev 169:102992

You J, Zhang C, Xue Z, Miao L, Ye B (2021) A bi-objective model for robust local container drayage problem. RAIRO Oper Res 55:625–646

You J, Wang Y, Xue Z (2023) An exact algorithm for the multi-trip container drayage problem with truck platooning. Transp Res Part E Logist Transp Rev 175:103138

Yu C-S, Li H-L (2000) A robust optimization model for stochastic logistic problems. Int J Prod Econ 64(1–3):385–397

Zhang R, Yun WY, Moon I (2009) A reactive Tabu search algorithm for the multi-depot container truck transportation problem. Transp Res Part E Logist Transp Rev 45(6):904–914

Zhang R, Yun WY, Kopfer H (2010) Heuristic-based truck scheduling for inland container transportation. OR Spectr. 32:787–808

Zhang R, Yun WY, Moon IK (2011) Modeling and optimization of a container drayage problem with resource constraints. Int J Prod Econ 133(1):351–359

Zhang R, Lu J-C, Wang D (2014) Container drayage problem with flexible orders and its near real-time solution strategies. Transp Res Part E Logist Transp Rev 61:235–251

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.