Distributionally robust optimization for the berth allocation problem under uncertainty

Agostinho Agra (✉)
aagra@ua.pt
Department of Mathematics and CIDMA, University of Aveiro

Filipe Rodrigues
frodrigues@iseg.ulisboa.pt
ISEG-School of Economics and Management, University of Lisbon
REM - Research in Economics and Mathematics, CEMAPRE

Abstract

Berth allocation problems are amongst the most important problems occurring in port terminals, and they are greatly affected by several unpredictable events. As a result, the study of these problems under uncertainty has been a target of more and more researchers. Following this research line, we consider the berth allocation problem under uncertain handling times. A distributionally robust two-stage model is presented to minimize the worst-case of the expected sum of delays with respect to a set of possible probability distributions of the handling times. The solutions of the proposed model are obtained by an exact decomposition algorithm for which several improvements are discussed. An adaptation of the proposed algorithm for the case where the assumption of relatively complete recourse fails is also presented. Extensive computational tests are reported to evaluate the effectiveness of the proposed approach and to compare the solutions obtained with those resulting from the stochastic and robust approaches.

keywords: Berth allocation, distributionally robust optimization, uncertain handling times, decomposition algorithm, Wasserstein distance.

Published in: Transportation Research Part B: Methodological 164, 1–24, 2022.
https://www.sciencedirect.com/science/article/pii/S0191261522001291

1 Introduction

Port operations play a key role in supply chain efficiency because they are responsible for connecting sea and land transportation. Among the port operations, berth allocation is the first to take place, and it influences all the subsequent operations. The berth allocation problem (BAP) has been studied for decades [21]. Berthing operations are frequently affected by several uncertainty sources such as weather conditions and mechanical failures. These unpredictable events may lead to delays on load/unload operations that can then propagate to other port operations. Thus, it is crucial to take uncertainty into account in the planning phase to prevent major disruptions caused by such events. During the last decade and half, we have witnessed an increase of studies on the BAP under uncertainty. In this paper, we consider the continuous BAP under uncertain handling times. In the continuous BAP, the vessels are allowed to berth in any place over the wharf. We propose a distributionally robust two-stage model, where the berth allocation decisions are the first-stage decisions (also known as here-and-now decisions), and the scheduling decisions - the start servicing times of the vessels and the delays from the requested departure times - are the second-stage decisions, which are adjustable to the observed handling times. The objective is to determine the berth plan that minimizes the expected sum of delays when the worst-case distribution of the handling times occurs.

The main purpose of this paper is to investigate the application of distributionally robust optimization (DRO) to the BAP under uncertainty. In particular, we intend to discuss implementation issues, to provide managerial insights, and to establish comparisons with the related and the most used methodologies: robust optimization (RO) and stochastic programming (SP).

Contrary to stochastic programming, where the uncertain parameters follow a specified probability distribution, in DRO, it is assumed that the uncertain parameters follow a distribution that lies in a set of probability distributions, known as the *ambiguity set*. In this paper, we consider the well-known Kantorovich ambiguity set, defined by all the probability distributions whose distance to a reference distribution is bounded by a model parameter. The 1-Wasserstein metric is used to compute the distance between the distributions, and the reference distribution is defined by a set of scenarios of handling times with equal probability of occurrence. Unlike robust optimization (RO), where the aim is to obtain the optimal solution when the worst-case scenario occurs, in the DRO the objective is to optimize the expected value of the objective function for the worst-case probability distribution within the ambiguity set. Following [5], we propose a decomposition algorithm for the BAP under uncertainty where, in each iteration, a first-stage solution is obtained by solving a master problem, then a subproblem is solved to determine the second-stage solution for each scenario, and, finally, the worst-case distribution is determined by solving a transportation problem. If a probability distribution leading to a worst expected value of the objective function is found, then an optimality cut is added to the master problem and the procedure is repeated.

In addition to the main goal of comparing the solutions resulting from DRO, SP, and RO, this paper pursues a secondary goal which consists on providing a detailed analysis of the solution procedure. This is extremely relevant because DRO is still a recent technique and little is known on the weaknesses, strengths, and impact of the assumptions of the purposed solution procedures. In particular, we propose several improvement strategies to overcome weaknesses of the proposed algorithm and analyse the impact of the relatively recourse property on the solution procedure, which may not hold when hard deadlines for departure times are imposed.

Our contributions are as follows:

i. We provide the first distributionally robust two-stage model for the berth allocation problem under uncertain handling times.

ii. An exact decomposition algorithm is introduced to solve the problem.

iii. The impact of the model assumptions in the solution procedure is discussed and an adaptation of the solution procedure to the case where the problem does not have the relatively complete recourse property is presented.

iv. Strategies to improve the performance of the solution procedure are discussed.

v. Computational tests are reported to analyse the performance of the purposed decomposition algorithm and to compare the solutions obtained with the distributionally robust model against those obtained from the stochastic and robust models.

The paper outline is as follows. In Section 2, we review the most relevant literature related to berth allocation under uncertainty. The deterministic berth allocation model is described in Section 3, while the distributionally robust two-stage model is given in Section 4. In Section 5, we introduce the decomposition algorithm, discuss the adaptation of the algorithm to the case where the relatively complete recourse property fails, and provide several improvement strategies. The computational results are reported in Section 6 and final conclusions and future research directions are discussed in Section 8.

2 Literature review

In this section, we relate our paper with the existing literature. Berth allocation problems have been intensively studied in the past. The deterministic problem was proven to be NP-hard by Lim [21]. BAPs under uncertainty have received significant attention during the last fifteen years since the seminal works of Moorthy and Theo [26] and Zhou et al. [45]. A review on BAPs under uncertainty can be extracted from [29], where a recent survey on berth allocation and quay crane assignment/scheduling problems under uncertainty is provided. Here we follow the terminology proposed in that survey.

Most of the papers on BAP under uncertainty consider the arrival times and/or the handling times as uncertain (see details on [29]). In this paper, we assume the arrival times are certain and the handling times are uncertain. The reasons for this option can be summarized as follows: (i) the certainty assumption of the arrival times does not limit the proposed approach because a delay in an arrival either it has no impact on the planned solution or it can be converted into a similar delay in the handling time; (ii) it makes it possible to ease the notation and presentation; (iii) assuming uncertainty in handling times leads a better and more flexible approximation to more complex problems where berth allocation and quay crane scheduling decisions are integrated. The reason is that possible conflicts occurring in the allocation of cranes to vessels can be easily captured as delays in the handling times. Nevertheless, we note that the approach proposed here can be easily extended to the case of uncertain arrival times.

The approaches for BAPs under uncertainty have been classified into different categories, namely, proactive, reactive, and proactive/reactive. In the proactive approaches, the operations plan is established before the events (e.g. arrival of vessels) take place; however, the uncertainty is taken into account to construct such a plan. An example of a proactive approach is to devise a plan that remains feasible for all possible realizations of the uncertain parameters. Proactive approaches have been widely studied in the past [4, 7, 8, 12, 19, 20, 22, 38, 44]. Reactive approaches are usually employed for rescheduling the operations when an unpredicted event forces the planner to deviate from a given baseline plan. These approaches were proposed in [3, 32, 34]. The proactive/reactive approaches combine the planning of decisions that are fixed or act as a baseline plan during the execution phase with reactive decisions to be taken in the execution phase when uncertainty is revealed. Proactive/reactive approaches can be found in [9, 16, 23, 24, 25, 27, 31, 33, 35, 37, 40, 41, 42]. A trend can be observed over the last 15 years: while the first approaches were mostly proactive, in the recent years, we can observe an increase of the publications proposing proactive/reactive approaches. In this paper, we propose a proactive/reactive approach, where the berth allocation positions are determined before the execution phase (corresponding to static decisions) and the scheduling of operations - recourse decisions - are adjustable to the observed handling times. Note that the berth allocation decisions are proactive because they take into account the uncertainty.

According to [29], four different classes of models have been proposed for solving BAPs under uncertainty: stochastic programming, robust optimization, fuzzy theory, and deterministic models. The DRO model proposed here belongs to a new class of models whose, to the best of our knowledge, have never been used before for solving BAPs under uncertainty. The most related models to the one presented here are stochastic programming and robust optimization models. Stochastic programming models for BAPs under uncertainty can be found in [19, 31, 40, 37], while robust optimization models were proposed in [25, 39, 40, 41]. These two modeling approaches were compared in [25, 40, 44]. Zhen [44] considered the BAP with periodical vessel arrivals under uncertain handling times. A sample average approximation method is used to approximate the expected cost for handling potential conflicts between vessels and derived a robust model based on dualization techniques. Liu et al. [25] and Xiang and Liu [40] presented several scenario-based robust and stochastic models for the BAP under uncertain arrival and handling times. In robust optimization models, uncertainty sets are frequently described by scenarios [25, 39, 40, 41]. For stochastic programming models, the uncertain parameters are usually assumed to follow a normal distribution [19, 25, 31, 40] and/or a uniform distribution [37, 40]. Decomposition approaches have also been proposed to BAPs. In the most related work, Xiang and Liu [40] combined a decomposition algorithm with a rolling horizon heuristic for solving large size instances. In their approach, multiple uncertainty sets are constructed by applying K-means clustering techniques and the problem is solved by an exact column-and-row generation algorithm.

Distributionally robust optimization (DRO) is a recent technique lying in the intersection of RO and SP. Although some authors claim that Scarf [30] was the first to employ this technique, just recently it became popular. DRO assumes that the exact probability distribution of the parameters is not known, but a partial knowledge of that distribution is known (e.g. some information regarding the model parameters). The true distribution is assumed to belong to the ambiguity set. The solutions are evaluated for the worst-case of the expected value considering all probability distributions of the uncertain parameters within the ambiguity set. If the ambiguity set contains only one distribution, then DRO coincides with SP. If the ambiguity set contains all possible probability distributions on a given support, then DRO coincides with RO. Thus, DRO generalizes both SP and RO. This relation is explored in our computational results. For a recent and very complete survey on DRO see [17]. An important issue in the DRO methodology is the choice of the ambiguity set. Here we assume a discrepancy-based ambiguity set, which includes all the probability distributions that are within a predefined maximum distance to a reference distribution. As distance metric, we use the Wasserstein metric of order 1. This ambiguity set, also known as the Kantorovich ambiguity set, has been intensively studied in recent years, both from theoretical and practical points of view. Some works combine both because the presented theoretical contributions are applied to practical problems such as the portfolio optimization [10, 15, 18]. In addition, one can find applications of distributionally robust approaches, using the Kantorovich ambiguity set, to other optimization problems, such as the telecommunication network expansion problem [43] and the strategic energy planning [14]. See [17] for an overview of the ambiguity sets and [11] for a discussion of the advantages of using this metric. From the practical point of view, the use of the Kantorovich ambiguity set has the advantage of requiring the control of a single non-negative parameter that limits the distance of the probability distribution to the reference distribution. Moreover, the stochastic and robust models are obtained for particular values of this parameter which helps to provide intuition on the performance of the solution procedure when tuning the parameter. To solve the DRO model we follow the decomposition approach proposed by [5] where convergence is proved for families of ambiguity sets such as the moment matching set and the Kantorovich set. For the reasons explained above we use the Kantorovich set, but the methodology used in this paper could be easily adapted for the moment matching set.

3 The formulation for the deterministic BAP

In this section, we present a formulation for the deterministic continuous BAP based on the time-space diagram, known as the Relative Position Formulation [13]. Consider a set of N heterogeneous vessels, a time horizon of M time periods, and assume that the wharf is divided into $J+1$ berth sections. Let us define the following sets: $V = \{1, \ldots, N\}$ the set of vessels, $T = \{1, \ldots, M\}$ the set of time periods, and $B = \{0, \ldots, J\}$ the set of berth positions.

The length of each vessel $k \in V$ - measured in number of berth sections - is given by L_k, the estimated handling time is given by H_k, the nominal arrival time is A_k, and the requested departure time is D_k. We assume that the requested departure time may not be satisfied and, in that case, there may occur delays. To prevent undesirable unbalanced cases where some vessels may have quite large delays while other vessels are served on time, we impose a limit C on the maximum allowed delay for each vessel. To ensure safety, it is also imposed a temporal gap of F time periods between the finishing time of a vessel and the start time of the next arriving vessels sharing common berth positions with it. The berthing position, berthing time, and delay of each vessel $k \in V$ are decision variables of the problem, represented by b_k, t_k, c_k, respectively. Additionally, to establish spacial and temporal relations between each pair of vessels, we consider the binary variables $x_{k\ell}$ and $y_{k\ell}$. Variables $x_{k\ell}$ assume value one when ship ℓ berths after ship k had departed, while variables $y_{k\ell}$ assume value one when ship ℓ berths below the berth position of ship k. With these decision variables, the deterministic continuous BAP can then be formulated as follows:

$$
\begin{aligned}
\min \quad & \sum_{k \in V} c_k && && (1)\\
s.t. \quad & x_{\ell k} + x_{k\ell} + y_{\ell k} + y_{k\ell} \geq 1, && k, \ell \in V, k < \ell, && (2)\\
& x_{\ell k} + x_{k\ell} \leq 1, && k, \ell \in V, k < \ell, && (3)\\
& y_{\ell k} + y_{k\ell} \leq 1, && k, \ell \in V, k < \ell, && (4)\\
& b_k \geq b_\ell + L_\ell + J(y_{k\ell} - 1), && k, \ell \in V, k \neq \ell, && (5)\\
& b_k \leq J - L_k, && k \in V, && (6)\\
& b_k \in \mathbb{Z}_0^+, && k \in V, && (7)\\
& x_{k\ell}, y_{k\ell} \in \{0, 1\}, && k, \ell \in V, k \neq \ell && (8)\\
& t_\ell \geq t_k + H_k + F + M(x_{k\ell} - 1), && k, \ell \in V, k \neq \ell, && (9)\\
& c_k \geq t_k + H_k - D_k, && k \in V, && (10)\\
& c_k \leq C, && k \in V, && (11)\\
& t_k \geq A_k, && k \in V, && (12)\\
& t_k, c_k \geq 0, && k \in V. && (13)
\end{aligned}
$$

The objective function (1) minimizes the sum of the delays over all vessels, henceforward referred to as tardiness. This objective was chosen because it is one of the objectives most frequently used in literature [29]; however, we would like to note that the algorithm proposed in this paper can be directly applied to BAPs with different objective functions. Constraints (2)-(8) model the berth allocation, while constraints (9)-(13) model the start time of the operations on the vessels and the delays. Constraints (2)-(4) are the usual constraints ensuring no overlap between each pair of vessels either in time or in space. Constraints (5) relate the berth position variables b_k with variables y_{kl}. Constraints (6) and (7) define the domain of the variables b_k. Constraints (8) define the binary variables. Constraints (9) relate the time variables of two vessels with the variables x_{kl}. Constraints (10) define the delay from the requested departure times, while constraints (11) impose a maximum allowed delay for each vessel. Constraints (12) ensure that each vessel starts to be served after its arrival, and constraints (13) define the domain of the continuous variables c_k and t_k. We note that model (1)-(13) can be strengthened by using the discretized formulation proposed in [2], which makes it possible to avoid the big-M constraints (5) and (9).

4 A distributionally two-stage robust formulation

The deterministic model for the BAP described in the previous section assumes that the arrival and the handling times of all vessels are exactly known in advance. However, this assumption is not true in most practical problems because both times may be uncertain due to several factors such as weather conditions and mechanical failures. In this paper, we assume the arrival times are certain and the handling times are uncertain; however, our approach can be easily extended to the case of uncertain arrival times.

To take the uncertainty in the handling times into account, we propose a two-stage model for the distributionally robust BAP (DRBAP). In the proposed two-stage model, the berth positions of the vessels - defined by variables x, y, and b - are the first-stage decisions, that is, decisions that must be taken before the uncertainty is revealed, while the service start times and delays - defined by variables t and c, respectively - are second-stage decisions taken after the uncertain handling times are revealed. We assume the handling time of each vessel $k \in V$ is a random variable defined by a probability distribution P with finite support Ω. The probability distribution $P = \{\Omega, \pi_P\}$ - where $\pi_P = (\pi_P^1, ..., \pi_P^{|\Omega|})$ and π_P^ω denotes the probability of scenario $\omega \in \Omega$ under probability distribution P - is not known with certainty, but it is assumed to belong to the ambiguity set $\mathbb{P}$. Moreover, H_k^ω represents the handling time of vessel $k \in V$ in scenario $\omega \in \Omega$, and it is assumed that any possible handling time of a given vessel is compatible with its maximum allowed delay and with the time horizon, that is, the following assumption holds:

Assumption 1: We assume $A_k + H_k^\omega \leq \min\{D_k + C, M\}, \quad \forall k \in V, \omega \in \Omega.$

Unlike stochastic programming, where the expected value of a given function is optimized under a specific probability distribution, distributionally robust optimization (DRO) aims to optimize the expected value of a function for the worst probability distribution in an ambiguity set. Hence, the distributionally robust formulation for the BAP under uncertain handling times is as follows:

$$\min \left\{ \max_{P \in \mathbb{P}} \ \mathbb{E}_P[Q(x, y, b)] \ | \ (2) - (8) \right\} \Leftrightarrow \min \left\{ \max_{P \in \mathbb{P}} \ \sum_{\omega \in \Omega} \pi_P^\omega Q_\omega(x, y, b) \ | \ (2) - (8) \right\}. \tag{14}$$

For any scenario $\omega \in \Omega$, the recourse function $Q_\omega(x, y, b)$ gives the minimum delay that can occur when the allocation decisions are fixed and scenario ω is realized. The recourse function is itself an optimization problem and to define it, new second-stage variables t_k^ω and c_k^ω depending on the scenario must be considered. These variables have the same meaning of the corresponding variables in the deterministic model. The recourse function $Q_\omega(x, y, b)$ is then given by:

$$
\begin{aligned}
Q_\omega(x,y,b) = \min \quad & \sum_{k\in V} c_k^\omega && && (15)\\
s.t. \quad & t_\ell^\omega - t_k^\omega \geq H_k^\omega + F + M(x_{k\ell} - 1), && k,\ell \in V, k \neq \ell, && (16)\\
& t_k^\omega \geq A_k, && k \in V, && (17)\\
& c_k^\omega - t_k^\omega \geq H_k^\omega - D_k, && k \in V, && (18)\\
& c_k^\omega \leq C, && k \in V, && (19)\\
& t_k^\omega, c_k^\omega \geq 0, && k \in V. && (20)
\end{aligned}
$$

An important observation on the recourse function is the absence of the relatively complete recourse property due to the constraints (19). This means that the feasibility of the second-stage problem (15)-(20) is not guaranteed for any first-stage solution, which has a great impact in the derivation of the algorithm proposed for solving the DRBAP. We elaborate on this in the next section. For the particular BAP we consider in this paper, only the variables x are required to determine the value of the recourse function. Hence, we could simply write $Q_w(x)$ instead of $Q_w(x, y, b)$.

Problem (14) includes three optimization problems because each recourse function is itself an optimization problem. In what follows, we refer to the problem of determining the value of the recourse function for a specific scenario as a second-stage problem. Note that the second-stage problems are solved assuming the first-stage decision variable are fixed. The inner maximization problem in (14) is called the distribution separation problem, and it consists of determining the probability distribution that maximizes the expected value of function $Q(x, y, b)$ with the first-stage decisions fixed.

For the proposed DRO model, we use the well-known Kantorovich ambiguity set [5]. Let us denote by $P = \{\Omega, \pi_P\}$ an unknown general probability distribution and by $P^* = \{\Omega, \pi_{P^*}\}$ a known reference probability distribution. A natural choice for the reference probability distribution is, for instance, the empirical distribution where all scenarios have the same probability of occurrence, that is, $\pi_{P^*}^\omega = 1/|\Omega|,\ \omega \in \Omega$. For a given parameter $\epsilon > 0$, the Kantorovich set can be written as follows [5]:

$$
\begin{aligned}
\mathbb{K}_\epsilon = \Bigg\{ & \pi_P \in \mathbb{R}^{|\Omega|} : \sum_{\omega\in\Omega}\sum_{\omega'\in\Omega} \| \omega - \omega' \|_1 \, k_{\omega\omega'} \leq \epsilon && && (21)\\
& \sum_{\omega'\in\Omega} k_{\omega\omega'} = \pi_P^\omega, && \omega \in \Omega, && (22)\\
& \sum_{\omega\in\Omega} k_{\omega\omega'} = \pi_{P^*}^{\omega'}, && \omega' \in \Omega, && (23)\\
& \sum_{\omega\in\Omega} \pi_P^\omega = 1, && && (24)\\
& \pi_P^\omega \geq 0, && \omega \in \Omega, && (25)\\
& k_{\omega\omega'} \geq 0, && \omega, \omega' \in \Omega\Big\}. && (26)
\end{aligned}
$$

Through the paper, we assume the support Ω is fixed, thus the set of the unknown probability distributions is given by $\mathbb{P} = \{P = \{\Omega, \pi_P\} : \pi_P \in \mathbb{K}_\epsilon\}$.

The Kantorovich ambiguity set defines the space of the allowed probability distributions in terms of the distance to the reference probability distribution. The dimension of such space is controlled by the parameter ϵ. When $\epsilon = 0$, $k_{\omega\omega'} = 0$, for all $\omega, \omega' \in \Omega$ such that $\omega \neq \omega'$. As a result, $\pi_P^\omega = \pi_{P*}^\omega$ for all $\omega \in \Omega$ and the Kantorovich set reduces to the singleton that includes the reference probability distribution, that is, $\mathbb{K}_0 = \{\pi_{P*}^\omega\}$. Hence, the DRBAP defined over the set $\mathbb{K}_0$ reduces to the stochastic BAP. When $\epsilon = \infty$, constraints (21) become redundant. As a result, the Kantorovich set will be defined by all probability distributions with support Ω, that is, $\mathbb{K}_\infty = \{\pi_P \in \mathbb{R}^{|\Omega|} : \sum_{\omega\in\Omega} \pi_P^\omega = 1, \pi_P^\omega \geq 0, \omega \in \Omega\}$. In particular, the set $\mathbb{K}_\infty$ includes all degenerated probability distributions where the corresponding probability vector π_P is composed by a single value 1 and $|\Omega| - 1$ values zeros. This corresponds to the case where a single scenario is considered. Hence, the DRBAP defined over the set $\mathbb{K}_\infty$ reduces to the robust BAP.

When the Kantorovich set is used, the distribution separation problem

$$\max \left\{ \sum_{\omega\in\Omega} \pi_P^\omega Q_\omega(x, y, b) \mid \pi_P \in \mathbb{K}_\epsilon \right\} \tag{27}$$

which aims to find the probability distribution π_P that maximizes the inner maximization problem in (14), where π_P is sufficiently close to the reference distribution π_{P^*}. The distance between the two distributions can be viewed as transportation problem (whose transportation plan is given by the value of the variables $k_{\omega\omega'}$) for moving the probability mass from π_{P^*} to π_P. The transportation cost is given by $\sum_{\omega\in\Omega} \sum_{\omega'\in\Omega} \parallel \omega - \omega' \parallel_1 k_{\omega\omega'}$ and bounded by ϵ, see [10] for details.

5 Exact decomposition algorithm

In this section, we present an exact decomposition algorithm for solving the DRBAP under uncertain handling times. The algorithm proposed is an iterative procedure inspired in Bansal et al. [5]. In their work, the authors consider three important assumptions: i) the feasible region defined by the first-stage variables is non-empty; ii) the model considered has relatively complete recourse (RCR), that is, for any feasible first-stage solution there is at least one feasible second-stage solution; and iii) the set Ω is finite. Although the first and the third assumptions hold in our case, the second assumption does not hold due to the presence of constraints (19). However, the RCR property can be guaranteed for our problem by defining parameter C as a sufficiently large value such that constraints (19) become redundant. For ease of presentation and understanding, we start by presenting the exact decomposition algorithm for the DRBAP assuming that C is a sufficiently large value and, consequently, the two-stage BAP has relatively complete resource. Next, in Section 5.1, we explain the importance of the RCR assumption and redefine the proposed algorithm for solving the DRBAP for any value of C, that is, for the case where this assumption does not hold.

The DRBAP can be rewritten using the epigraph reformulation as follows:

$$\min \quad \theta \tag{28}$$

$$s.t. \quad (2) - (8) \tag{29}$$

$$\theta \geq \max \left\{ \sum_{\omega \in \Omega} \pi_P^\omega Q_\omega(x, y, b) : P \in \mathbb{P} \right\}. \tag{30}$$

The proposed algorithm follows a Benders decomposition where constraint (30) is replaced by a set of optimality cuts that avoid to use explicitly the subproblems $Q_\omega(x, y, b)$. For each first-stage solution (x, y, b), an optimality cut can be devised using the optimal dual multipliers of the corresponding subproblem $Q_\omega(x, y, b)$, for each scenario ω. This leads to an exponential set of optimality cuts. The decomposition algorithm is an iterative procedure that solves, at each iteration, a relaxation of the model (28)-(30) - called master problem - where only a small subset of optimality cuts is considered. Then, for the (first-stage) solution of the master problem, a separation problem is solved to identify a violated optimality cut or to show that no such a cut exists. If a violated cut is found, it is added to the master problem and the process is repeated. Otherwise, the algorithm terminates.

This algorithm - called the distributionally robust L-shaped algorithm - consists on solving three problems at each iteration. The master problem is the first to be solved to provide a first-stage solution. With that solution, we solve the second-stage problems that consist of determining the recourse function $Q_\omega(x, y, b)$ for each $\omega \in \Omega$. The values of the recourse functions are then used to solve the distribution separation problem, making it possible to identify the worst vector of probabilities in the ambiguity set with respect to the current first-stage solution. Lastly, such a vector of probabilities is used together with the optimal dual solutions of the second-stage problems to generate a new cut for the master problem. The optimal value of the distribution separation problem is an upper bound for the optimal value of the DRBAP, while the master problem provides lower bounds for such a value. Both bounds are successively improved from iteration to iteration until they coincide. When it happens, the process ends, ensuring that the optimal solution was found.

Now, we explain in more detail how the optimality cuts are obtained. Given a first-stage solution (θ, x, y, b) to the master problem and considering a scenario $\omega \in \Omega$, the recourse function $Q_\omega(x, y, b)$ is given by the linear problem (15)-(20), which can be solved in polynomial time. Assigning the dual variables $u_{k\ell}^\omega, k, \ell \in V, k \neq \ell$, $r_k^\omega, k \in V$, $z_k^\omega, k \in V$, and $v_k^\omega, k \in V$ to constraints (16), (17), (18), and (19), respectively, the dual problem associated with scenario $\omega \in \Omega$ is as follows:

$$\max \quad \sum_{k,\ell \in V, k \neq \ell} (H_k^\omega + F + M(x_{k\ell} - 1)) u_{k\ell}^\omega + \sum_{k \in V} A_k r_k^\omega + \sum_{k \in V} (H_k^\omega - D_k) z_k^\omega + \sum_{k \in V} (-C) v_k^\omega \tag{31}$$

$$s.t. \quad z_k^\omega - v_k^\omega \leq 1, \qquad k \in V, \tag{32}$$

$$r_k^\omega - \sum_{\ell \in V, \ell \neq k} u_{k\ell}^\omega + \sum_{\ell \in V, \ell \neq k} u_{\ell k}^\omega - z_k^\omega \leq 0, \quad k \in V, \tag{33}$$

$$r_k^\omega, z_k^\omega, v_k^\omega \geq 0, \qquad k \in V, \tag{34}$$

$$u_{k\ell}^\omega \geq 0, \qquad k, \ell \in V, k \neq \ell. \tag{35}$$

Assume this dual problem has an optimal solution. Denoting by $\overline{u}_{k\ell}^\omega$, $\overline{r}_k^\omega$, $\overline{z}_k^\omega$, and $\overline{v}_k^\omega$ the optimal value of the dual variables, the following equality holds

$$Q_\omega(x,y,b) = \sum_{k,\ell \in V, k\neq \ell} (H_k^\omega + F + M(x_{k\ell}-1))\overline{u}_{k\ell}^\omega + \sum_{k\in V} A_k \overline{r}_k^\omega + \sum_{k\in V}(H_k^\omega - D_k)\overline{z}_k^\omega + \sum_{k\in V}(-C)\overline{v}_k^\omega,$$

and, consequently, the following optimality cut is obtained

$$\theta \geq \sum_{\omega\in\Omega} \pi_P^\omega \left(\sum_{k,\ell \in V, k\neq \ell} (H_k^\omega + F + M(x_{k\ell}-1))\overline{u}_{k\ell}^\omega + \sum_{k\in V} A_k \overline{r}_k^\omega + \sum_{k\in V}(H_k^\omega - D_k)\overline{z}_k^\omega + \sum_{k\in V}(-C)\overline{v}_k^\omega \right) \tag{36}$$

where the probabilities $\pi_P^\omega, \omega \in \Omega$, are obtained by solving the distribution separation problem. If (36) is not satisfied by the current solution of the master problem, then the cut is added to the master problem to obtain a new first-stage solution.

The full description of the decomposition algorithm is given in Algorithm 1. The first master problem to solve - denoted by M_0 - does not include any optimality cut.

Algorithm 1 Distributionally Robust L-shaped method for the BAP.

Initialize $i \leftarrow 0$; $LB \leftarrow -\infty$; $UB \leftarrow +\infty$ and construct the initial master problem M_0
Solve the master problem M_0. Let (x^0, y^0, b^0) be the optimal solution
Define the initial solution as the best solution found: $(x^*, y^*, b^*) = (x^0, y^0, b^0)$
for each scenario $\omega \in \Omega$ **do**
  Solve the second-stage problem $Q_\omega(x^i, y^i, b^i)$
end for
while $UB > LB$ **do**
  Solve the distribution separation problem (transportation problem (27)) considering the first-stage solution (x^i, y^i, b^i), and $Q_\omega(x^i, y^i, b^i)$ for all $\omega \in \Omega$. Let $\pi_P^\omega, \omega \in \Omega$, be the resulting probabilities
  if $UB > \sum_{\omega\in\Omega} \pi_P^\omega Q_\omega(x^i, y^i, b^i)$ **then**
    $UB \leftarrow \sum_{\omega\in\Omega} \pi_P^\omega Q_\omega(x^i, y^i, b^i)$
    Update the best solution found: $(x^*, y^*, b^*) = (x^i, y^i, b^i)$
  end if
  if $UB \leq LB$ **then**
    STOP
  end if
  Set $i \leftarrow i + 1$
  Construct master problem M_i by adding the optimality cut (36) to M_{i-1}
  Solve the master problem M_i
  Set (x^i, y^i, b^i) to the optimal solution of M_i and LB to its value
end while
Return the optimal solution (x^*, y^*, b^*) and its cost LB (=UB)

Although the convergence of the L-shaped algorithm was proven in [5] for the case where both the master problem and the second-stage problems are linear, the proof holds true when only the second-stage problems are linear.

5.1 The case of no relatively complete recourse

As remarked before, the L-shaped algorithm presented is valid for two-stage problems with relatively complete recourse (RCR), which is not the case of our problem for all values of C. In this section, we start by explaining why the RCR property is important. Then, we explain how to adapt the algorithm presented in the absence of the RCR property.

The optimality cuts added to the master problem are based on the dual optimal solutions of the second-stage problems defined for each scenario. If the RCR property does not hold, some second-stage problems may be infeasible, and, consequently, by linear programming theory, the corresponding dual problem may be either infeasible or unbounded. As we will see in the next section, we can show that this dual becomes unbounded. Thus, it is not possible to derive any cut for the master problem. In addition, finding an optimality cut requires to solve the distribution separation problem, which in turn requires the determination of the recourse functions $Q_\omega(x, y, b)$ associated with all second-stage problems. When the RCR property does not hold, it may not be possible to obtain the value of the recourse function for all scenarios because some resulting problems may be infeasible.

The RCR property is important to ensure the validity and convergence of the Algorithm 1; however, we can also ensure such validity and convergence by replacing the RCR property by a weaker property. In fact, it is enough to guaranty that any solution generated by the master problem is feasible for all scenarios considered. This means that we can restrict the solution space of the master problem to the set of robust solutions, that is, to the set of solutions that are feasible for all the scenarios considered. A possible way of doing this is to include the time constraints associated with the second-stage problem of each scenario into the master problem. The drawback of this approach is that it may lead to large size complex models. However, such a robust model can be approached using efficient techniques such as decomposition algorithms that generate only a subset of scenarios, see [28] for such an example. For simplicity, here we follow an intermediate approach that consists of including only the constraints of the second-stage problems associated with the non-dominated scenarios [1]. To the best of our knowledge, this approach is new in the context of decomposition algorithms for distributionally robust optimization. In our case, the non-dominated scenarios are those where a maximum number of delays occurs and the size of those delays is larger. Hence, denoting by $\Xi \subset \Omega$ the set of non-dominated scenarios and by RHS_Cut_j the right-hand side of the optimality cut (36) derived at iteration j, the master problem M_i at each iteration $i > 0$ is as follows:

$$
\begin{aligned}
M_i = \min \quad & \theta && && (37)\\
s.t. \quad & (2) - (8) && && (38)\\
& \theta \geq RHS\_Cut_j, && j = 1, ..., i && (39)\\
& t_\ell^\omega - t_k^\omega \geq H_k^\omega + F + M(x_{k\ell} - 1), && k, \ell \in V, k \neq \ell, \omega \in \Xi && (40)\\
& t_k^\omega \geq A_k, && k \in V, \omega \in \Xi && (41)\\
& c_k^\omega - t_k^\omega \geq H_k^\omega - D_k, && k \in V, \omega \in \Xi && (42)\\
& c_k^\omega \leq C, && k \in V, \omega \in \Xi && (43)\\
& t_k^\omega, c_k^\omega \geq 0, && k \in V, \omega \in \Xi && (44)\\
& \theta \geq 0. && && (45)
\end{aligned}
$$

The first master problem solved, M_0, corresponds to the robust model and it is defined by the objective function (37), by the constraints (38) and (40)-(45), and by the additional constraints $\theta \geq \sum_{k \in V} c_k^\omega,\ \omega \in \Xi$. With this reformulated master problem, Algorithm 1 converges to the optimal solution of the DRBAP for any value of the parameter C. As a consequence, in what follows, we consider Algorithm 1 with the master problem (37)-(45). In section 6, we explain in detail how to obtain the set Ξ for the BAP considered.

5.2 Dual problem seen as a flow problem

In this section, we analyse in detail the dual problem (31)-(35) to derive important results. We start by introducing the following proposition, whose proof is given in the Appendix B.

Proposition 5.1. *In any optimal solution of the dual problem, inequalities* (33) *are satisfied at equality.*

Based on Proposition 5.1, we can assume that inequalities (33) are replaced by the following equations in the dual problem

$$r_k^\omega + \sum_{\ell \in V, \ell \neq k} u_{\ell k}^\omega = \sum_{\ell \in V, \ell \neq k} u_{k\ell}^\omega + z_k^\omega, \qquad k \in V. \tag{46}$$

Replacing constraints (33) by constraints (46), it is possible to observe that the dual problem without constraints (32) has the structure of a flow problem in an auxiliary network where the set of nodes is $V' = V \cup \{O, D\}$. Node O is an artificial origin and node D is an artificial destination. There is an arc between O and each $k \in V$, an arc between k and ℓ whenever $x_{k\ell} = 1$, and an arc between each $k \in V$ and D. The flow associated with each type of arc is represented by variables r_k^ω, $u_{\ell k}^\omega$, and z_k^ω, respectively, and the corresponding objective function coefficients are given by $A_k, H_k^\omega + F + M(x_{k\ell} - 1)$, and $H_k^\omega - D_k$. Equalities (46) can be seen as flow conservation constraints on this auxiliary graph. Therefore, we can use the fact that the coefficients matrix is totally unimodular (TUM) and that the RHS are integral to derive the following proposition used in the following section, whose proof is given in the Appendix C.

Proposition 5.2. *Any basic solution of the dual problem associated with scenario $\omega \in \Omega$ is integral.*

In addition, it is important to note that variables u_{kl}^ω are not bounded and, consequently, they may assume integer values larger than one.

Example 5.3. *In Figure 1, we present an example of a solution of the dual problem when $v_k^\omega = 0, \forall k \in V$, that is, the case where C is large and no hard deadlines are imposed. In this example, we omit the ω from the variables for simplicity.*

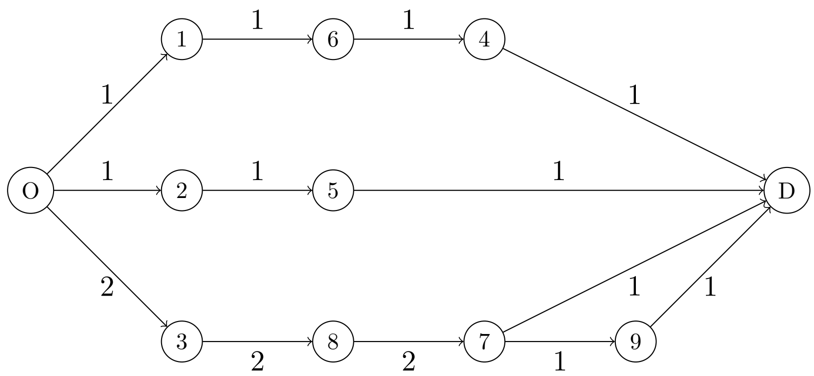

Figure 1: Example of a dual solution on a network for a dual problem, where the value associated with each arc represents the value of the corresponding variable.

There are four paths in Figure 1: i) $P_1 : \; O \to 1 \to 6 \to 4 \to D$*, corresponding to* $r_1 = u_{16} = u_{64} = z_4 = 1$*; ii)* $P_2 : \; O \to 2 \to 5 \to D$*, corresponding to* $r_2 = u_{25} = z_5 = 1$*; iii)* $P_3 : O \to 3 \to 8 \to 7 \to D$ *; and iv)* $P_4 : \; O \to 3 \to 8 \to 7 \to 9 \to D$*. Paths* P_3 *and* P_4 *share some arcs and the dual variables associated with those arcs count the number of paths crossing each arc. Thus,* $r_3 = 2, u_{38} = 2, u_{87} = 2, u_{79} = 1, z_7 = 1, z_9 = 1$.

The objective function value of the dual problem corresponding to each path - assuming that variables $x_{k\ell}$ *associated with variables* $u_{k\ell}$ *are equal to one - is:*
Path $P_1 : \eta_1 = A_1 + (H_1 + F) + (H_6 + F) + (H_4 - D_4)$
Path $P_2 : \eta_2 = A_2 + (H_2 + F) + (H_5 - D_5)$
Path $P_3 : \eta_3 = A_3 + (H_3 + F) + (H_8 + F) + (H_7 - D_7)$
Path $P_4 : \eta_4 = A_3 + (H_3 + F) + (H_8 + F) + (H_7 + F) + (H_9 - D_9)$
Each path has a clear interpretation. For example, path P_1 *is associated with the delay of ship* 4 *(denoted by* η_1*) that occurs when this ship is operated after ship* 6*, which in turn is operated after ship* 1*. With this dual solution, composed of four paths, constraint* (36) *for a single scenario with probability 1 becomes*

$$\theta \geq A_1 + (H_1 + F) + M(x_{16} - 1) + (H_6 + F) + M(1 - x_{64}) + (H_4 - D_4)$$
$$+A_2 + (H_2 + F) + M(1 - x_{25}) + (H_5 - D_5) + A_3 + (H_3 + F)$$
$$+M(1 - x_{38}) + (H_8 + F) + M(1 - x_{87}) + (H_7 - D_7)$$
$$+A_3 + (H_3 + F) + M(1 - x_{38}) + (H_8 + F) + M(1 - x_{87}) + (H_7 + F) + M(1 - x_{79}) + (H_9 - D_9).$$

Setting the first-stage variables $x_{kl} = 1$ *to the corresponding paths, we are imposing the constraint:*

$$\theta \geq \eta_1 + \eta_2 + \eta_3 + \eta_4.$$

Next we discuss the role of variables v_k^ω.

Proposition 5.4. *The dual problem associated with scenario* $\omega \in \Omega$ *is either unbounded or has an optimal solution with* $v_k^\omega = 0, \forall k \in V$.

The proof of proposition 5.4 is given in the Appendix D. The arguments used in the proof also show that in case multiple optimal solutions exist, then the solutions with $v_k^\omega > 0$ cannot be basic.

In the case of relatively complete recourse, the first-stage solution must be feasible for each scenario. This means that the critical path associated with each vessel has a delay lower than or equal to C. Thus, only Case 2 in the proof of Proposition 5.4 can occur. If the first-stage solution is obtained without guaranteeing feasibility for each possible scenario, then a violation can be identified by the unbounded dual subproblem.

5.3 Improvement strategies

Preliminary tests showed that the convergence of Algorithm 1 can be quite slow, providing consecutive lower bounds equal to zero. In order to improve the proposed algorithm, it is important to look deeper into the optimality cuts and understand why they may be inefficient. Consider again the first-stage solution (x, y, b) of the master problem, a scenario $\omega \in \Omega$, the recourse function $Q_w(x, y, b)$ given by the linear problem (15)-(20), and its dual problem given by (31)-(35).

In each iteration of Algorithm 1, an optimality cut (36) with some variables $\overline{u}_{k\ell}^\omega$ taking a positive integer value is added, see Proposition 5.2. As these variables have coefficient $H_k^\omega + F + M(x_{k\ell} - 1)$, it is easy to see that any alternative first-stage solution with $x_{k\ell} = 0$ makes this term to take a large negative value and, consequently, the optimality cut (36) is satisfied with $\theta = 0$. As a result, with the exception of the first master problem M_0, several subsequent master problems may provide lower bounds equal to zero. This indicates that several alternative first-stage feasible solutions are found without repeating at least one of the variables $x_{k\ell}$ set to one in the previous iterations. Given that the master problem has typically many alternative feasible solutions, the number of iterations leading to lower bounds equal to zero may be large, which may result in a slow convergence of the proposed L-shaped algorithm. The previous analysis of the dual problem and the optimality cuts make it possible to understand some convergence issues associated with the proposed L-shaped algorithm. To improve its efficiency, we propose four improvement strategies.

Replace M by a tighter value

The first improvement strategy consists of replacing the M value by a tighter value in constraints (40) and in the optimality cut (36). With this strategy, we aim to increase the coefficient of variable u_{kl}^ω in the optimality cut without losing optimality. This may speed up the convergence of the algorithm by increasing the value of the master problem faster. For the BAP considered, we can take $M_{kl} = \max\{\min\{M, D_k + C\} + F - A_\ell\}, 0\}$, for each $k, l \in V, k \neq l$. To show that the tightened inequality

$$t_\ell^\omega \geq t_k^\omega + H_k^\omega + F + M_{k\ell}(x_{k\ell} - 1),$$

is valid for the set of feasible solutions under scenario $\omega \in \Omega$, we consider the case $x_{k\ell} = 0$ because the case $x_{k\ell} = 1$ is trivial. In addition, the proof is only presented for the case $M_{kl} = D_k + C + F - A_\ell$ because the remaining ones are trivial.

From constraints (42) and (43), it follows that $t_k^\omega + H_k^\omega - D_k - C \leq 0$. Combining this inequality with $t_\ell^\omega \geq A_\ell$, we have $t_\ell^\omega \geq A_\ell + (t_k^\omega + H_k^\omega - D_k - C)$. This implies

$$t_\ell^\omega \geq t_k^\omega + H_k^\omega + F - F + A_\ell - D_k - C \quad \Rightarrow \quad t_\ell^\omega \geq t_k^\omega + H_k^\omega + F - (D_k + C + F - A_\ell)$$

$$\Rightarrow t_\ell^\omega \geq t_k^\omega + H_k^\omega + F - M_{k\ell} \quad \Rightarrow \quad t_\ell^\omega \geq t_k^\omega + H_k^\omega + F + M_{k\ell}(x_{k\ell} - 1).$$

Reduce the number of alternative optimal solutions

The second enhancement consists of reducing the number of alternative solutions of the master problem by using the following set of constraints:

$$x_{\ell k} = 0, \qquad k, l \in V : A_k < A_\ell.$$

These constraints - that we refer to as non-overlapping constraints - prevent situations where the vessel k arrives before vessel ℓ, but the operations on vessel k start after the vessel ℓ has been processed. They do not avoid the case where the start time of vessel k is lower than the finishing time of vessel ℓ; however, it is important to note that these constraints may cut some feasible solutions. In such a case, the proposed algorithm may provide a heuristic solution.

To prove that the final solution obtained with the non-overlapping constraints is optimal or to derive an optimal solution from it, we can apply again Algorithm 1 with some modifications. Such modifications consist of replacing the non-overlapping constraints by the constraint $\sum_{k \in V | A_k < A_\ell} x_{\ell k} \geq 1$ and setting the initial upper bound equal to the value of the final solution obtained. This second execution of the algorithm ensures that either the solution obtained with the non-overlapping constraints or the new solution obtained is optimal. Some preliminary tests not reported here revealed that this two-phase procedure is, in general, not faster than applying Algorithm 1 only once without imposing the non-overlapping constraints. As a result, when this strategy is used in the computational experiments reported in Section 6, the second execution of the algorithm mentioned here is not considered.

Tighten of inequalities (36)

A third improvement strategy is the tighten of inequalities (36). First, observe that we can drop the term in v_k^ω, since we are interested only in the case where the dual problem has a finite optimal solution, see Proposition 5.4. The inequality can be written as follows.

$$\theta \geq \sum_{\omega \in \Omega} \pi_P^\omega \left(S^\omega + \sum_{k,\ell \in V, k \neq \ell} M(x_{k\ell} - 1)\overline{u}_{k\ell}^\omega \right) \tag{47}$$

where

$$S^\omega = \sum_{k,\ell \in V, k \neq \ell} (H_k^\omega + F)\overline{u}_{k\ell}^\omega + \sum_{k \in V} A_k \overline{r}_k^\omega + \sum_{k \in V} (H_k^\omega - D_k)\overline{z}_k^\omega.$$

As explained in Section 5.2, the term inside the parenthesis appearing in the RHS of inequalities (47) is the cost of the aggregation of paths using arcs that correspond to first-stage solutions with $x_{k\ell} = 1$ and $\overline{u}_{k\ell}^\omega > 0$. If $x_{k\ell} = 0$, a value $M\overline{u}_{k\ell}^\omega$ is subtracted to the RHS to ensure that for such scenario ω the corresponding term in not positive, given that $\overline{u}_{k\ell}^\omega$ is an integer positive value. Now, let us define the set of arcs used in such paths with $\overline{u}_{k\ell}^\omega > 0$:

$$U^\omega = \left\{ (k,\ell) \in V \times V : k \neq \ell \wedge \overline{u}_{k\ell}^\omega > 0 \right\},$$

and set

$$X^\omega = \sum_{(k,\ell) \in U^\omega} x_{k\ell}.$$

Hence, inequalities (47) can be tighten as follows:

$$\theta \geq \sum_{\omega \in \Omega} \pi_P^\omega (S^\omega - S^\omega(|U^\omega| - X^\omega)). \tag{48}$$

Formally, we state the validity of the new inequalities in the following proposition, whose proof is presented in the Appendix E.

Proposition 5.5. *Given a first-stage solution* (x, y, b)*, for each* $\omega \in \Omega$ *let* $(\overline{r}_k^\omega, \overline{u}_{k\ell}^\omega, \overline{z}_k^\omega, 0)$ *denote a dual solution to* $Q_\omega(x, y, b)$ *and let* π_P *be a probability distribution belonging to the ambiguity set. Then, inequality* (48) *is valid for the feasible region of the master problem.*

Readjustment of the first-stage solution

By taking the definition of the first-stage variables into account, we can see that if the berth position of vessel ℓ does not intersect the berth position of vessel k, then we can either set $y_{k\ell}$ to 1 (if vessel ℓ berths below vessel k) or $y_{\ell k}$ to 1 (if vessel k berths below vessel ℓ). In this case, the optimal value of the variables $x_{k\ell}$ and $x_{\ell k}$ can be either set to one or zero. The choice is of main importance in this problem for two reasons. The first follows from the observation made above that only the variables x are required to determine the value of the recourse function. Setting an x variable to 1 implies to add a new constraint (40) to the problem $Q_\omega(x, y, b)$. Such a constraint not only complicates the problem $Q_\omega(x, y, b)$, by adding an unnecessary constraint, but it also may affect the second-stage solution because it restricts the berthing time of a ship and, consequently, it may have an impact on the delay of that ship. For this reason, this strategy must be used whenever the decomposition algorithm is applied because otherwise the solution of each second-stage problem may not be optimal (as needed). The second reason follows from another observation, made in the second enhancement strategy, stating that alternative first-stage solutions with different sets of variables $x_{k\ell}$ fixed to one will generate additional iterations of Algorithm 1. Hence, when multiple first-stage solutions exist we should choose a solution with a minimal set of x variables set to one. For these two reasons, whenever a first-stage solution is generated with $x_{k\ell} = 1$, if there is no intersection between the berth positions occupied by vessels k and ℓ, then we flip the value of variable $x_{k\ell}$ to zero and set the corresponding variable y to one.

Additionally, since the readjustment increases the number of x variables equal to zero, we use the following property to fix dual variables $\overline{u}_{k\ell}^\omega$ to zero in the dual problems.

Proposition 5.6. *In any optimal solution of the dual problem,* $\overline{u}_{k\ell}^\omega = 0$ *for all* $k, l \in V, \omega \in \Omega$ *such that* $x_{k\ell} = 0$ *in the current first-stage solution of the master problem.*

The proof of Proposition 5.6 comes directly from the fact that if $x_{k\ell} = 0$, then the coefficient of $\overline{u}_{k\ell}^\omega$ in the objective function (31) (which is $H_k^\omega + F - M$) is a large negative number because M acts as a big-M constant. This means that $\overline{u}_{k\ell}^\omega = 0$ in any optimal dual solution.

6 Computational Results

In this section, we report extensive computational results having three main goals in mind: i) to investigate the performance of the decomposition algorithm and the effect of the improvement strategies discussed in Section 5.3; ii) to explore in detail the structure of the distributionally robust solutions; and iii) to compare the DRO solutions with stochastic and robust solutions.

Our experiments are conducted on the set of 100 instances[1] used in [28]. The instances are labelled as *R_N_i*, where $N \in \{6, 7, ..., 15\}$ is the number of vessels and $i \in \{1, ..., 10\}$ is a number

[1] online available at http://sweet.ua.pt/aagra/

that identifies the instance. The instances were originally designed for the berth allocation and quay crane scheduling problem and consequently, several modifications were applied to make them suitable for the BAP. We kept the original arrival times and length of the vessels. The time horizon considered is one week (128 hours) and each time period represents two hours. Hence, we consider $M = \frac{128}{2} = 84$. We assume that the quay area is divided into 21 berth sections, which makes it possible to operate 3 vessels simultaneously at the port. Lower numbers of berths may lead to infeasible problems because some deadlines may not be respected, while greater values may lead to problems where there are no interferences among vessels. The deterministic handling time of vessel $k \in V$ is defined by $H_k = Q_k/1.5c_1$, where Q_k is the total cargo of vessel k and c_1 is the processing rate of the first crane (both values are the ones reported in the original instances provided in [28]). The coefficient 1.5 is used to compute the handling times because it makes it possible to obtain a range of handling times from 4 hours to 2 days, which reflects real-world situations. The scenarios for the uncertain handling times that form set Ω are generated through the multiple constrained budget (MCB) uncertainty set used in [28] and defined as follows.

$$\Omega = \left\{ H^\omega : H_k^\omega = H_k + [\hat{H}_k \delta_k],\ 0 \leq \delta_k \leq 1,\ k \in N, \sum_{k=1+j[N/G]}^{\min\{(j+1)[N/G],N\}} [\delta_k] \leq \Gamma_j,\ j \in \{0, 1, \ldots, G-1\} \right\}$$

In this uncertainty set, H_k and $\hat{H}_k$ are, respectively, the deterministic handling time and the maximum allowed delay in the handling time of vessel k. The round operator $[\cdot]$ is used to ensure that only integer handling times are obtained. This uncertainty set groups the vessels into G groups according to their arrival times and associates a budget constraint to each group. The vessels in the first group are the ones with the lowest arrival times and the first budget constraint imposes that at most Γ_0 of those vessels can suffer delays on the handling times. In this paper, the parameters in the uncertainty set are the same used in [28]: $\hat{H}_k = 2,\ k \in V$, $G = 3$, and $\Gamma_j = 1, j = 0, 1, 2$. This means that the vessels are grouped into three sets according to their arrival times, and at most one vessel in each group can suffer a delay by either one or two time periods on its handling time. As a result, the number of scenarios considered for each instance only depends on the number of existing vessels, see Table 1.

Table 1: Number of scenarios for each instance in terms of the number of vessels

N	6	7	8	9	10	11	12	13	14	15
$\lvert\Omega\rvert$	125	175	245	343	441	567	729	891	1089	1331

For each vessel $k \in V$, the requested departure time is defined as $D_k = A_k + \max_{\omega\in\Omega}\{H_k^\omega\} + 0.2H_k$, and the maximum allowed delay C is fixed to 10. Over this section, we are going to use the master problem defined by (37)-(45) in the proposed decomposition algorithm (Algorithm 1). This master problem involves the definition of the set Ξ of the non-dominated scenarios. In the BAP, any delay may have a negative impact on the objective function value, and such impact increases with the magnitude of the delay. In the scenarios defined in Ω, any vessel can either suffer no delays or suffer a delay of one or two time periods. As a consequence, the non-dominated scenarios are the ones where the number of vessels delayed is the maximum allowed and all of them have a delay of two time periods.

For the experiments, we use the Xpress Optimizer Version 27.01.02 with the default options on a computer with a CPU Intel(R) Core i7, with 16GB RAM.

6.1 The effect of the improvement strategies

In this section, we test the improvement strategies defined in Section 5.3 on a subset of 10 instances: instances R_j_1, $j \in \{6, 7, ..., 15\}$. For the experiments, we consider $\epsilon = 5$. As mentioned before, the last improvement strategy - which consists of readjusting the first-stage variables x - is a crucial strategy in our procedure because the second-stage problems solved at each iteration only use the value of variables x. If the readjustment of the variables x is not applied, then the solution of each second-stage problem may not be optimal (as required). For this reason, the last improvement strategy is included in all the computational results reported.

Hence, we only test the first three strategies that are summarized as follows.

S_M : Replace the value M by a tighter value;

S_x : Use of the non-overlapping constraints;

S_c : Replace inequalities (36) by the tighten inequalities (48).

The running times, in seconds, resulting from the combination of the different improvement strategies are displayed in Table 2. The first row identifies the strategy or the combination of strategies used. The second column, identified by "$-$", corresponds to the case where none of the improvements strategies is used. The lowest computational times for each instance are marked in bold.

Table 2: Performance of the improvement strategies

Strategies	$-$	S_M	S_c	S_x	$S_M + S_x$	$S_x + S_c$	$S_M + S_c$	$S_M + S_x + S_c$
R_6_1	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
R_7_1	1	**0**	**0**	**0**	**0**	**0**	**0**	**0**
R_8_1	7	**5**	10	7	**5**	10	8	9
R_9_1	32	15	25	21	**14**	19	21	16
R_10_1	655	261	629	287	**135**	278	560	188
R_11_1	105	**34**	78	77	40	49	80	37
R_12_1	9676	5252	5601	5021	**2610**	3776	5595	3361
R_13_1	132	91	122	56	35	**12**	238	**12**
R_14_1	43459	28921	43322	6553	**1799**	3430	32215	3354
R_15_1	25484	8241	11311	5656	**1534**	3431	11101	3388

The results in columns 3-5 show the individual performance of each improvement strategy, and they clearly show that strategies S_M and S_x drastically reduce the total running time. Strategy S_c seems not to be very efficient, specially when combined with strategy S_M. The shortest running times are consistently obtained when strategies S_M and S_x are combined.

It is important to recall that while strategies S_M and S_c keep the optimality of the obtained solutions, strategy S_x may cut some feasible solutions and, consequently, it may eliminate optimal solutions. To understand the impact on the optimal solution caused by the strategy S_x, we display, in Table 3, the optimality gaps associated with the solutions obtained by this strategy, with respect to the optimal solutions for the 10 instances considered.

Table 3: Optimality gaps associated with the solutions obtained by using strategy S_x

Instance	R_6_1	R_7_1	R_8_1	R_9_1	R_10_1	R_11_1	R_12_1	R_13_1	R_14_1	R_15_1
Gap(%)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	31.03	0.00	0.00

The obtained results show that with the strategy S_x it was possible to obtain optimal solutions for 9 of the 10 instances considered. This means that the non-overlapping constraints do not frequently cut optimal solutions. In fact, the additional results reported in the Appendix F show that strategy S_x does not cut optimal solutions in around 85% of the cases.

The preliminary tests reported in this section can be summarized as follows: Both strategies S_M and S_x clearly contribute to reduce the computational time. The strategy S_x may cut optimal solutions, but it seems that this is not the most frequent case. The strategy S_c does not contribute to reduce the computational time in the presence of both strategies S_M and S_x. Hence, in what follows, all the results were obtained by using the combination of the strategies $S_x + S_M$.

6.2 An illustrative example

This section provides a detailed discussion about a specific instance, the instance R_10_1. The parameters of this instance - arrival time, length, deterministic handling time, and requested departure time for each vessel $k \in V$ - are displayed in Table 4.

Table 4: Parameters of instance R_10_1

k	1	2	3	4	5	6	7	8	9	10
A_k	10	13	13	15	19	21	24	30	41	50
L_k	5	7	6	5	5	6	6	6	6	6
H_k	5	9	8	11	5	3	6	3	5	4
D_k	18	26	25	30	27	27	33	36	49	57

The parameter ϵ used in the Kantorovich ambiguity set strongly affects the structure and the cost of the distributionally robust solutions. The cost of the resulting solutions increases as the value of the parameter ϵ increases. The case $\epsilon = 0$ corresponds to the stochastic case, where it is assumed that the distribution of the uncertain handling times is the reference probability distribution considered. In our experiments, the reference probability distribution is the empirical distribution, which is described by a set Ω of scenarios, all having the same probability of occurrence. The case $\epsilon = \infty$ corresponds to the robust case, where only the worst-case scenario is considered. This robust solution can be obtained by using finite values for ϵ, and such values can vary from instance to instance. For the instance R_10_1, the robust solution is obtained for any $\epsilon \geq 5$ as shown in Figure 2.

The behaviour of the cost of the DRO solutions in terms of the parameter ϵ is similar to the behaviour of the parameter Γ used in the budget uncertainty set proposed by Bertsimas and Sim [6], which is known as the *price of robustness*.

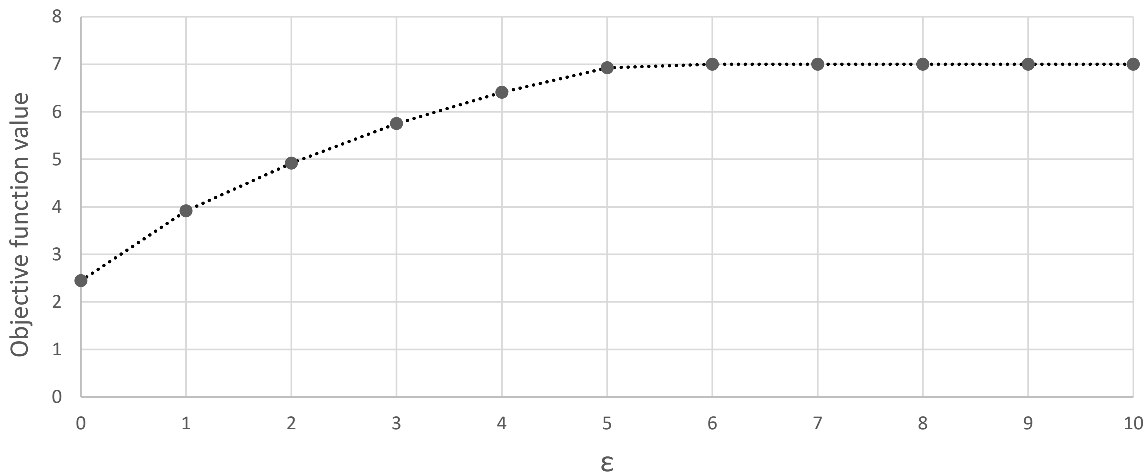

Figure 2: Cost of the distributionally robust solution as a function of the parameter ϵ.

The stochastic solutions, distributionally robust solutions, and robust solutions can be structurally very different, as happens in this particular instance. In Figures 3, 4, and 5, we provide the graphical representation of the stochastic solution, of the distributionally robust solution (with $\epsilon = 1$), and of the robust solution, respectively. Each vessel is represented by a rectangle whose height corresponds to the length of the vessel and the width corresponds to the deterministic handling time. Each 1×1 black rectangle represents the requested departure time of a specific vessel. In the three figures presented, vessels 9 and 10 are omitted to make the representation of the remaining vessels clearer. There is a large temporal slack between the arrivals of vessels 9 and 10 and the departures of the remaining vessels, and vessels 9 and 10 occupy the same space-time position in all the three solutions presented. Remember that, in our case, the function to optimize is the sum of the delays of the vessels with respect to the requested departure times, that is, the tardiness. Additionally, as stated before, the MCB uncertainty set considers $G = 3$ sets of vessels: $G_1 = \{1, 2, 3\}, G_2 = \{4, 5, 6\}$, and $G_3 = \{7, 8, 9, 11\}$. In each one of these groups at most one delay of one or two time periods in a single vessel can occur.

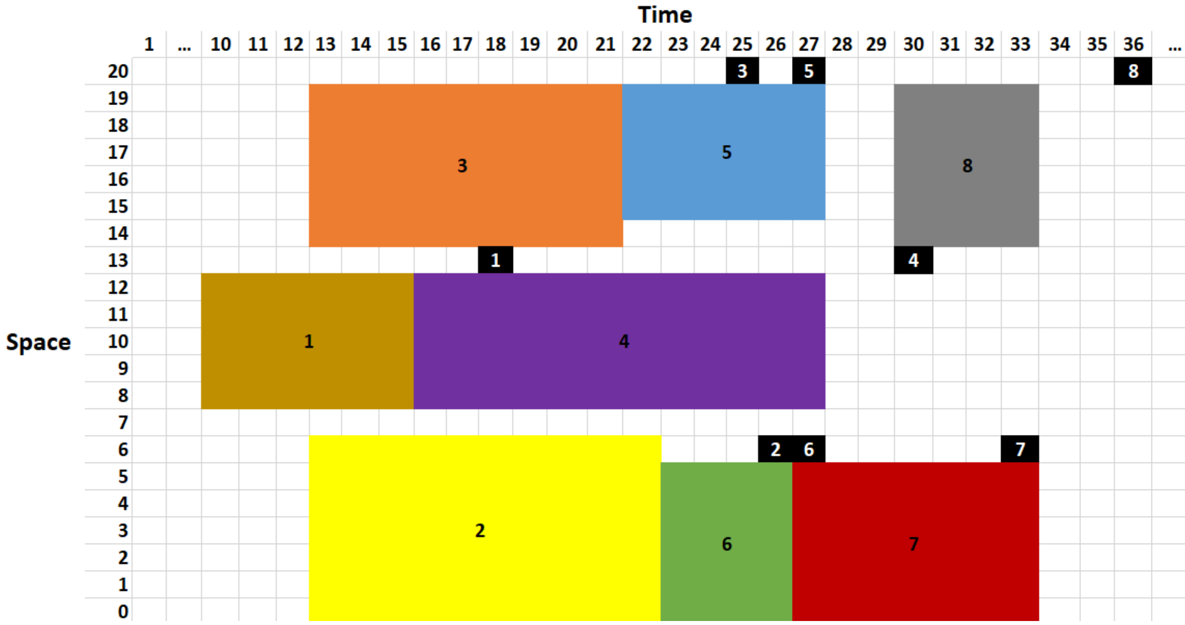

Figure 3: Stochastic solution ($\epsilon = 0$).

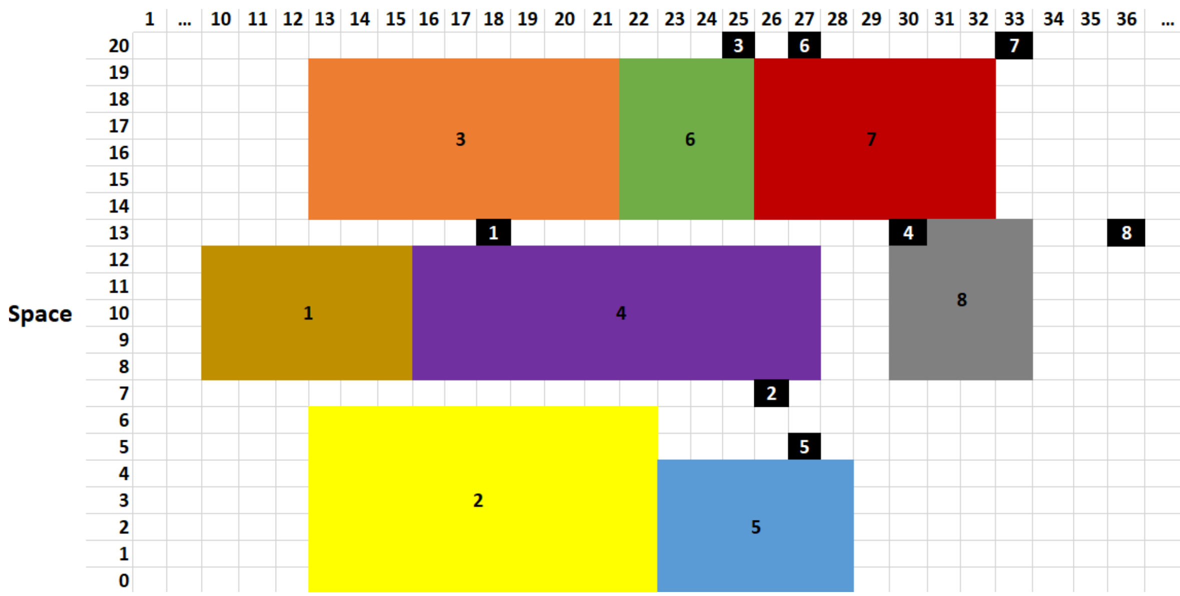

Figure 4: Distributionally robust solution ($\epsilon = 1$).

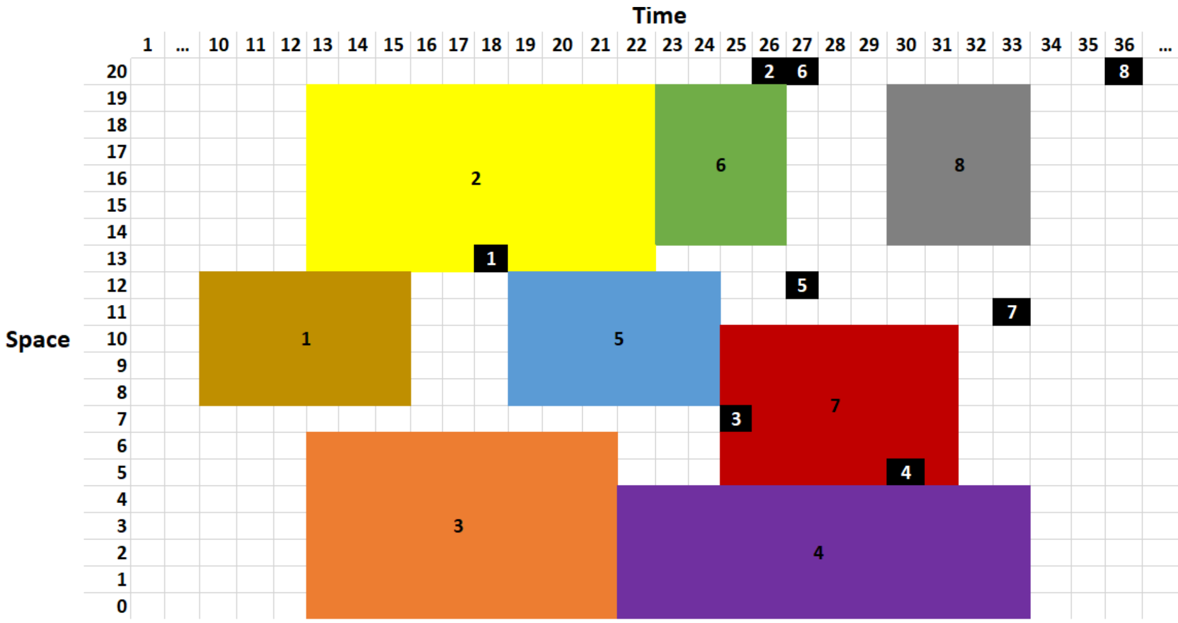

Figure 5: Robust solution ($\epsilon = \infty$).

Figures 3, 4, and 5 clearly show that the solutions obtained by stochastic programming, robust optimization, and DRO can be structurally very different. Such differences occur because these approaches were projected to optimize specific different situations.

Stochastic programming was especially designed to optimize an expected function assuming a specific reference probability distribution, the empirical probability distribution in our case. Figure 3 shows a solution that has no delays in the deterministic case. However, one can observe that for vessels 5 and 7 there are no buffers, that is, the ending time in the deterministic case is equal to the requested departure time. Additionally, for vessel 6 the buffer is just one time period. This means that delays in handling the vessels can easily lead to delays at their departure. In particular, the berth allocation of vessels 2, 5, 7 can easily generate large delays since a delay in handling vessel 2 can propagate to vessels 6 and 7, and a delay in handling vessel 6 can propagate to vessel 7. By setting a delay of 2 periods on the handling time of vessels 2, 5, and 6 we get a tardiness value of 9, which is in fact the worst-case scenario for this solution. Nevertheless, considering all the possible scenarios of delays with equal probability, this is the optimal solution when the goal is to minimize the expected tardiness value.

DRO assumes that the distribution of the uncertain parameters can deviate from the reference probability distribution and, as a result, it optimizes the expectation of a function assuming the worst possible probability distribution defined by the ambiguity set. Figure 4 shows a solution where a delay of one period is assumed for vessel 5. Comparing this solution with the stochastic one we observe that, except for vessel 5, all the remaining vessels have a buffer of at least one period. This minimizes the impact of the possible propagation of delays. The worst-case scenario will occur when a delay of 2 periods occurs in the handling times of vessels 3, 5, and 7, which gives a tardiness of 8.

Robust optimization aims to optimize the worst-case scenario, that is, to optimize a function considering the scenario of uncertainty that leads to the worst value of such a function. Since this procedure disregards the deterministic scenario (because in our case it cannot be the worst-case scenario), we see from Figure 5 that the robust solution assumes a delay of 3 periods even in the case where no handling operations are delayed. However, the solution generates large buffers for the remaining vessels, with the exception of vessel 6 for which the buffer is one, and avoids the situations that occurred in the two previous solutions where three vessels share the same berthing positions and no slack time between the handling operations is considered to avoid the propagation of delays. In this case, a worst-case scenario will occur when the handling of vessels 3 and 4 suffers a delay of 2 periods each, giving a tardiness value of 7.

Next, we summarize some performance measures for the three reported solutions when evaluated under four different situations: i) deterministic scenario - we compute the tardiness assuming that the deterministic handling times occur; ii) worst-case scenario - we compute the tardiness on the scenario leading to the highest tardiness; iii) empirical distribution - we compute the expected tardiness assuming the empirical distribution occurs; and iv) worst-case distribution - we compute the expected tardiness assuming the worst distribution in the Kantorovich ambiguity set $\mathbb{K}_1$ occurs. The obtained results are reported in Table 5.

Table 5: Evaluation of the obtained solutions under different situations

Situation	Stochastic	DRO	Robust
Deterministic scenario	0.00	1.00	3.00
Worst-case scenario	9.00	8.00	7.00
Empirical distribution	2.43	2.54	4.28
Worst-case distribution	4.16	3.91	5.29

Table 5 clearly show that each approach leads to the best solution regarding the situation for which it was designed, that is: i) stochastic programming provides solutions with the lowest expected tardiness in the empirical distribution; ii) DRO generates solutions with the lowest expected tardiness for the worst probability distribution in the ambiguity set; and iii) robust optimization derives solutions with the lowest tardiness when the worst-case scenario occurs. Another interesting observation is that DRO provided a solution that - in terms of quality - is in the middle of the solutions obtained by stochastic programming and robust optimization, when evaluated in terms of the worst-case scenario, the deterministic scenario, and the empirical distribution. Hence, the solution obtained by DRO is, simultaneously, less conservative than the robust solution in terms of the expected value, and more protected against worst-case scenarios and worst-case distributions than the stochastic solution.

6.3 Main results

In this section, we present results for all the instances considered, except for the instances R_14_3 and R_14_7 because they are infeasible for the parameters considered. The results are presented in an aggregated way such that all instances having the same number of vessels are grouped. For each $N \in \{6, ..., 15\}$, we denote by R_N_* the set of instances R_N_i, $i \in \{1, ..., 10\}$. This section is divided into two parts. In the first part, we analyse the impact of parameter ϵ in the obtained solutions. In the second part, we present a comparative study between distributionally robust optimization, stochastic programming, and robust optimization.

6.3.1 The impact of parameter ϵ

Parameter ϵ is used in the Kantorovich ambiguity set to control the range of probability distributions allowed for the uncertain parameters, and it has a great impact on the obtained solutions. We start by analysing the impact of the parameter ϵ in the average computational times associated with all sets of instances considered. The obtained results are displayed in Table 6 for three distinct values of ϵ: 1, 5, and 10. The first column identifies the set of instances. Columns M, SS, and DS report the average total computation time spent on solving the master problems, the second-stage problems, and the distribution separation problems, respectively. The columns *Total* report the total computational time.

Table 6: Average computational times (in seconds)

	$\epsilon = 1$				$\epsilon = 5$				$\epsilon = 10$			
	M	SS	DS	*Total*	M	SS	DS	*Total*	M	SS	DS	*Total*
R_6_*	0	0	0	0	0	0	0	0	0	0	0	0
R_7_*	0	0	1	1	0	0	0	1	0	0	0	0
R_8_*	0	0	1	1	0	0	1	1	0	1	1	2
R_9_*	1	1	3	5	1	1	3	6	1	1	3	5
R_10_*	16	6	15	38	18	6	13	37	12	5	12	29
R_11_*	97	22	63	182	16	9	24	50	12	8	21	42
R_12_*	285	46	124	455	286	43	112	441	199	32	94	325
R_13_*	1015	95	287	1397	621	88	242	951	519	72	211	802
R_14_*	5367	249	732	6347	2552	198	572	3322	1902	163	472	2537
R_15_*	10845	632	1729	13207	10925	645	1528	13099	9284	572	1388	11244

The obtained results indicate that the computational time required to solve the instances increases as the number of vessels increases, and it decreases as the value of ϵ increases. For the larger instances, the time spent on solving the second-stage problems is lower than the time required to solve the distribution separation problem, which in turn is much lower than the time required for solving the master problems. In fact, the percentage of the total time required for solving all second-stage problems, distribution separation problems, and master problems is, respectively, 5%, 14%, and 81%.

Table 7 reports average objective function values for all the sets of instances in terms of the parameter ϵ. In particular, we consider the cases $\epsilon = 0$ (stochastic case), $\epsilon = 1$, $\epsilon = 5$, $\epsilon = 10$, and $\epsilon = \infty$ (robust case). Also in this table, we present the average number of iterations required to obtain the final solutions for the cases $\epsilon = 1, 5, 10$. The solutions of stochastic and robust approaches can be obtained directly by using Algorithm 1 with ϵ equal to zero and with ϵ equal to a sufficiently large value, respectively; however, for the particular case of the BAP, this is not the most efficient way of obtaining such solutions. A more efficient way consists of solving a single model that includes all the scenarios considered. The model used for the robust case is exactly the first master problem solved, that is, M_0. The model used for the stochastic case is given in the Appendix A. Considering these single models, it was possible to obtain all stochastic and robust solutions with an average computational time for each set of instances R_k_*, with $k = 6, \ldots, 15$, lower than 250 seconds.

Table 7: Average objective function value and average number of iterations for the instances considered.

	Objective function value					Iterations		
	$\epsilon = 0$	$\epsilon = 1$	$\epsilon = 5$	$\epsilon = 10$	$\epsilon = \infty$	$\epsilon = 1$	$\epsilon = 5$	$\epsilon = 10$
R_6_*	0.9	1.1	1.2	1.2	1.2	2	2	2
R_7_*	0.9	1.3	2.1	2.1	2.1	5	3	3
R_8_*	1.2	1.6	2.4	2.4	2.4	5	5	6
R_9_*	2.0	2.9	4.3	4.5	4.5	9	10	10
R_10_*	1.5	2.1	3.5	3.5	3.5	26	26	24
R_11_*	5.1	6.3	8.2	8.4	8.4	41	27	24
R_12_*	4.7	5.7	7.7	7.9	7.9	73	70	61
R_13_*	3.3	4.2	5.7	6.0	6.0	114	98	85
R_14_*	5.4	8.3	10.8	11.3	11.3	192	149	124
R_15_*	6.5	8.1	10.8	11.2	11.2	297	282	239

The obtained results clearly show that the average objective function value increases as the value of ϵ increases. For most of the instances, the case $\epsilon = 10$ is equivalent to the case $\epsilon = \infty$, and consequently, the costs in both cases are the same. For this reason, the case $\epsilon = 10$ will be omitted in the next section. We can also see that the average number of iterations tends to decrease as the value of ϵ increases, which may justify why the average total computational time tends to decrease as the value of ϵ increases.

The results in Table 7 are used to understand the impact of the parameter ϵ in the value of the obtained solutions. As mentioned before, the results were all obtained by using strategy S_x that, although speed up the algorithm a lot, may cut optimal solutions. For this reason, the optimality gaps of the obtained solutions and the percentage of instances for which the optimal solution was found are reported in the Appendix F.

6.3.2 A comparative study

In this section, we compare the distributionally robust solutions obtained with $\epsilon = 1$ and $\epsilon = 5$, denoted by DRO_1 and DRO_5, respectively, with the stochastic and robust solutions ($SP = DRO_0$ and $RO = DRO_\infty$, respectively). The solutions for all instances and approaches are initially obtained in a first phase by using the proposed L-shaped algorithm. Such solutions are then evaluated in particular probability distributions in a second phase and their corresponding value - the expected tardiness - is computed. As in the previous section, the instances are grouped by the number of vessels. The probability distributions used for evaluating the solutions in the second phase correspond to the worst probability distributions belonging to the following Kantorovich ambiguity sets:

i) Kantorovich set $\mathbb{K}_0$. This is the case where a single probability - the empirical probability - is considered;

ii) Kantorovich set $\mathbb{K}_1$;

iii) Kantorovich set $\mathbb{K}_5$;

iv) Kantorovich set $\mathbb{K}_\infty$. This is the case where a single scenario - the worst-case scenario - is considered.

Our experiments - reported in Figure 6 - aim to investigate the impact of obtaining distributionally robust solutions with ambiguity sets different from the ones in which the solutions are evaluated. It is important to note that the average value of the solutions is the best (the lowest), when the same value of ϵ - and consequently the same Kantorovich ambiguity set - is used in both phases. Hence, to better visualize the results, we compute the average gap of the obtained solutions with respect to the solutions of the best approach. For example, when the solutions are evaluated in the ambiguity set $\mathbb{K}_1$, we compute the gaps with respect to the solutions obtained with $\epsilon = 1$. To do so, we use the formula $Gap = \frac{DRO_j - DRO_1}{DRO_1} \times 100$, where DRO_j is the average cost of the solutions obtained with $\epsilon = j$, for $j = 0, 1, 5, \infty$. These gaps are zero when $j = 1$ and consequently, the line associated with the approach DRO_1 does not appear in the graphic associated with $\mathbb{K}_1$ because it coincides with the horizontal axis.

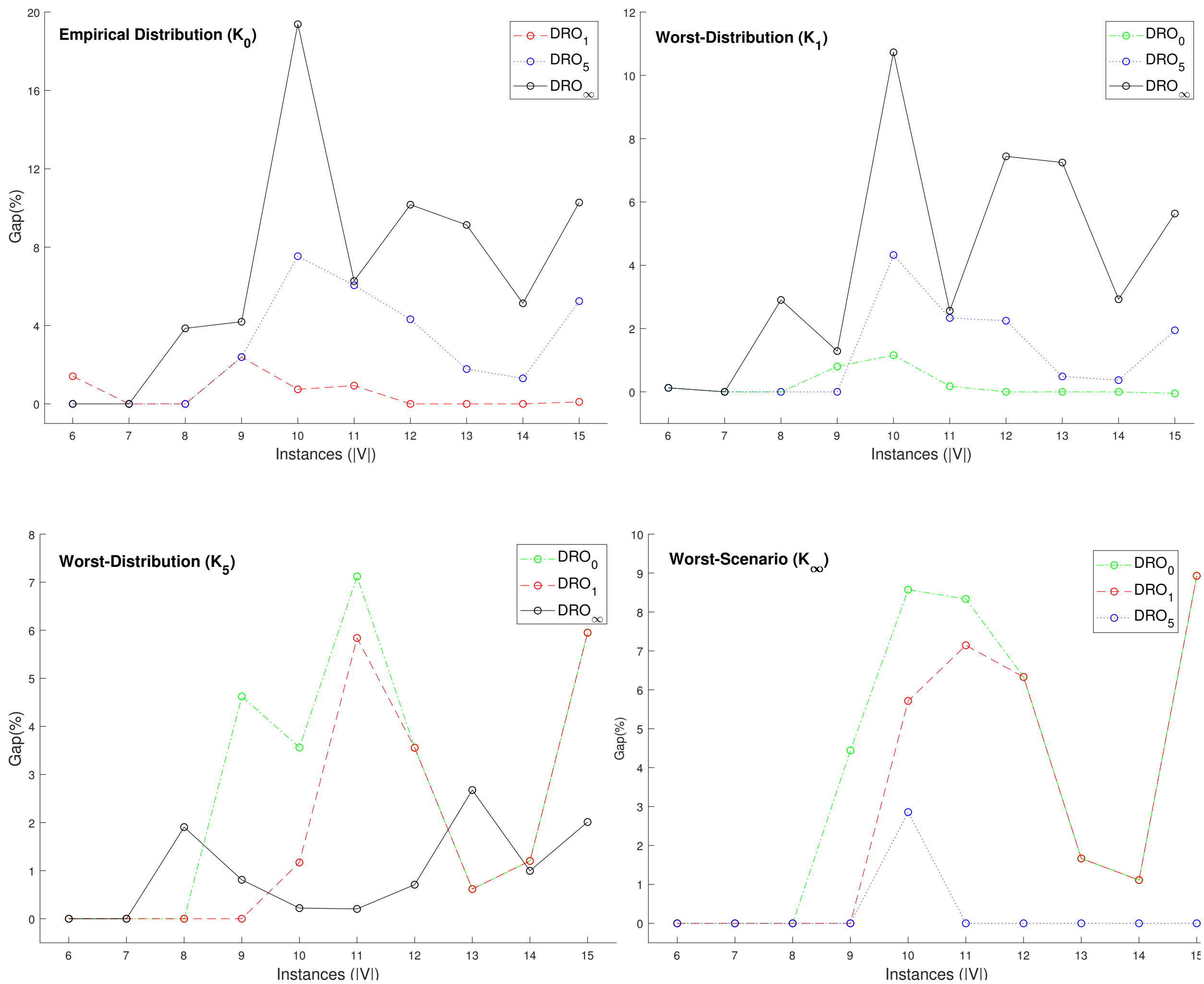

Figure 6: Evaluation of the distributionally robust solutions in different ambiguity sets.

The obtained results show that DRO_1 outperforms $SP = DRO_0$ when the uncertainty is high, that is, when the Kantorovich sets $\mathbb{K}_5$ and $\mathbb{K}_\infty$ are considered. The DRO_1 cannot be better than SP when evaluated in the empirical distribution; however, our experiments indicate that both approaches lead to solutions with the same average objective function value for the largest instances, the ones with $N = 12, 13, 14, 15$. In addition, we can conclude that DRO_1 clearly outperforms $RO = DRO_\infty$ when the uncertainty is low, that is, when the ambiguity set used to evaluate the solutions is $\mathbb{K}_0$ and $\mathbb{K}_1$. The approach DRO_5 outperforms RO when the uncertainty is low. When the worst-case scenario is considered, the average cost of both approaches is the same for all sets of instances, except for the instances with 10 vessels. DRO_5 is clearly outperformed by SP when the uncertainty is low; however, this behaviour is inverted when the uncertainty is high.

A general conclusion that can be drawn from the reported results is that the distributionally robust solutions are never worse than the stochastic and robust solutions simultaneously. Saying different, the value of the distributionally robust solutions is better than or equal to the value of the stochastic solution or to the value of the robust solution.

Now, we present the minimum and maximum differences between the value of the distributionally robust solutions and the value of the stochastic and robust solutions when evaluated on the different Kantorovich ambiguity sets considered. The minimum and maximum differences were computed considering the full set of 98 instances. The obtained results are displayed in Table 8. The first column identifies the ambiguity set used to evaluate the solutions. The second and third columns report the minimum and maximum difference between the value of the stochastic solutions and the value of the distributionally robust solutions obtained with $\epsilon = 1$. The fourth and fifth columns report the minimum and maximum differences between the value of the robust solutions and the value of the distributionally robust solutions with $\epsilon = 1$. The last four columns report equivalent results for the distributionally robust solutions obtained with $\epsilon = 5$. Positive values indicate a better performance of the distributionally robust solutions.

Table 8: Minimum and maximum differences between the distributionally robust solutions and the stochastic and robust solutions

	$SP - DRO_1$		$RO - DRO_1$		$SP - DRO_5$		$RO - DRO_5$	
	Min	Max	Min	Max	Min	Max	Min	Max
$\mathbb{K}_0$	-0.5	0.0	-0.1	2.6	-1.7	0.0	-0.3	2.6
$\mathbb{K}_1$	0.0	0.2	0.0	2.1	-1.0	0.2	-0.2	2.1
$\mathbb{K}_5$	0.0	2.0	-2.5	1.5	0.0	2.8	0.0	1.5
$\mathbb{K}_\infty$	0.0	2.0	-4.0	0.0	0.0	4.0	-1.0	0.0

The obtained results indicate that DRO_1 is always better than SP when the last three ambiguity sets are considered and it can reduce the expected tardiness in at most 2 time periods (4 hours). For the first ambiguity set, where SP is the best approach, the gains obtained by SP with respect to DRO_1 are at most 0.5 time periods (1 hour). Hence, we can conclude that DRO_1 usually performs better than SP. DRO_1 also performs better than RO in the first two ambiguity sets and the total reduction in the expected tardiness is at most 2.6; however, when the last two sets are considered, RO can reduce the expected tardiness in 4 time periods (8 hours) in relation to DRO_1.

DRO_5 performs better than SP when the last two ambiguity sets are considered and it can reduce the expected tardiness in at most 4 time periods. When the uncertainty is low, SP can decrease the expected tardiness in at most 1.7 time periods. Taking the four ambiguity sets considered into account, we can see the following: when DRO_5 performs better than RO, the reduction in the expected tardiness is at most 2.6 time periods, but when DRO_5 performs worse than RO, the increase in the expected tardiness is no more than 1 time period. This means that DRO_5 is a nice alternative approach to RO.

In the experiments reported before, we used Kantorovich ambiguity sets to evaluate the obtained solutions, all having the same set of scenarios as basis. With the next experiments, we aim to investigate how the SP, DRO_1, DRO_5, and RO solutions react when different scenarios are considered. To do so, we randomly generate a sample of 1000 scenarios of uncertainty, where the handling time of each vessel can vary between its deterministic handling time and its deterministic handling time plus τ time periods. This means that the increase on the handling time of each vessel is given by a Uniform distribution in $[0, \tau]$. We tested the values $\tau = 2$ and $\tau = 3$. It is important to note that in these experiments infeasible scenarios may be generated because our BAP has not relatively complete recourse. Hence, after evaluating the obtained solutions in the new generated scenarios, we computed the following performance measures: the average tardiness in all the feasible scenarios (*Average*), the tardiness in the worst feasible scenario (*Worst-case*), and the rate of infeasible scenarios (*%Infeasible*). The reported results are displayed in Tables 9 and 10.

Table 9: Evaluation of the obtained solution with scenarios from the distribution U(0,2)

	Average				*Worst-Case*				*%Infeasible*			
	SP	DRO_1	DRO_5	RO	SP	DRO_1	DRO_5	RO	SP	DRO_1	DRO_5	RO
R_6_*	1.0	1.0	1.0	1.0	9.0	9.0	9.0	9.0	2.2	2.2	2.2	2.2
R_7_*	1.5	1.5	1.5	1.5	9.0	9.0	9.0	9.0	0.0	0.0	0.0	0.0
R_8_*	1.8	1.7	1.7	1.9	10.0	10.0	10.0	12.0	2.5	2.5	2.5	2.5
R_9_*	3.8	3.8	3.8	3.8	11.0	11.0	11.0	13.0	0.0	0.0	0.0	0.0
R_10_*	3.1	3.0	3.1	3.3	22.0	22.0	22.0	22.0	0.0	0.0	0.0	0.0
R_11_*	8.7	8.5	8.7	8.9	38.0	38.0	38.0	38.0	4.1	4.1	4.1	4.1
R_12_*	8.2	8.2	8.4	8.8	46.0	46.0	46.0	48.0	2.0	2.0	2.0	2.0
R_13_*	7.3	7.0	7.1	7.4	36.0	35.0	36.0	37.0	0.6	0.0	0.6	1.1
R_14_*	14.6	14.6	14.5	14.9	50.0	50.0	50.0	50.0	5.0	5.2	7.8	5.8
R_15_*	15.4	15.4	15.5	16.0	50.0	51.0	50.0	55.0	9.2	9.2	8.9	10.3

Table 10: Evaluation of the obtained solution with scenarios from the distribution U(0,3)

	Average				*Worst-Case*				*%Infeasible*			
	SP	DRO_1	DRO_5	RO	SP	DRO_1	DRO_5	RO	SP	DRO_1	DRO_5	RO
R_6_*	1.4	1.4	1.4	1.4	10.0	10.0	10.0	10.0	2.2	2.2	2.2	2.2
R_7_*	2.6	2.6	2.6	2.5	15.0	15.0	15.0	15.0	0.0	0.0	0.0	0.0
R_8_*	2.9	2.7	2.7	3.1	15.0	15.0	15.0	22.0	5.0	5.0	5.0	5.0
R_9_*	6.8	6.6	6.6	6.5	23.0	23.0	23.0	25.0	0.0	0.0	0.0	0.0
R_10_*	5.5	5.5	5.6	5.6	32.0	32.0	32.0	32.0	0.0	0.0	0.0	0.0
R_11_*	12.0	11.3	11.5	11.7	47.0	47.0	45.0	45.0	14.0	16.6	16.7	16.1
R_12_*	12.4	12.4	12.3	13.0	49.0	49.0	49.0	54.0	12.8	12.8	12.7	11.5
R_13_*	13.4	12.6	12.7	12.7	51.0	51.0	50.0	52.0	8.2	7.2	8.7	10.5
R_14_*	22.1	21.6	21.6	22.2	61.0	61.0	61.0	61.0	27.4	27.9	34.6	31.7
R_15_*	23.9	23.7	23.5	24.2	67.0	77.0	64.0	74.0	26.3	27.0	28.2	30.1

Tables 9 and 10 indicate that the relative differences between the average costs of the solutions obtained by the four approaches compared are not large. In fact, the average relative difference between the cost of the solutions obtained by DRO_1, DRO_5, and RO with respect to the cost of the solutions obtained by SP is, respectively, -1.3%, -0.0%, and +3.2% for Table 9 and -2.5%, -2.3%, and +0.1% for Table 10. These small average relative differences can be explained by the following fact. Each row of Tables 9 and 10 corresponds to a set of 10 instances (except row R_14_*) and, in some of those instances, the congestion at the port is low because the vessel arrival times to the port are spread over time. This results in slack times between vessels sufficiently large to mitigate the impact of the uncertainty, and therefore, no significant differences between the approaches used to deal with the uncertainty are observed. The existence of this kind of instances clearly explains the small average relative differences between the average cost of the solutions obtained by the different approaches. Note that there are several instances where the relative difference between some approaches is larger than 100%.

The reported results also indicate that robust optimization frequently leads to overconservative solutions because in both tables RO has the worst values for the quality parameters considered. DRO_1 is the approach that more consistently performs better for all the quality parameters considered when the distribution U(0,2) is used. In particular, this is the approach that, in general, leads to the best values for each group of instances in terms of the quality parameter $Average$. However, it is important to note that DRO_1 does not outperforms SO for all the instances considered. SP outperforms DRO_1 in 20% of the instances, while DRO_1 outperforms SO in 26% of the instances. The better performance of DRO_1 comparing to SO can easily be explained by the fact that the solutions of the DRO_1 were obtained taking into account an ambiguity set where several probability distributions were considered and not only the reference probability distribution as in SO. Hence, the solutions of DRO_1 are expected to be more *robust* than the solutions of SO when evaluated under different probability distributions.

We can also see that the best results in terms of the highest sum of delays for the 10 groups of instances considered were always obtained by DRO_5 when the distribution U(0,3) is considered. A general conclusion that can be drawn from the reported results is that, in terms of the quality parameters considered, there are very few groups of instances where the solutions obtained by DRO_1 and DRO_5 are worst than the stochastic and robust solutions simultaneously.

7 Scalability analysis

In this section, we analyse the behaviour of the proposed decomposition algorithm in terms of the size of the instances and the value of some parameters. The proposed decomposition algorithm involves three optimization problems: the master problem, the second-stage problems, and the distributionally separation problem. The hardness of the master problem is mainly affected by i) the size of the instances considered (number of vessels and number of time periods), ii) the number of non-dominated scenarios, and iii) the structure of the instances itself (in particular, instances with a small number of berth positions where the arrivals of the vessels to the port are closer are more difficult to solve). This last factor clearly explains why the master problems associated with some instances with 12 vessels are harder to solve than the ones associated with some instances with 15 vessels. Therefore, the hardness of the master problems does not necessarily increases as the size of the instances increases. After solving the master problem, the second step of the DA is to solve the second-stage problems, that is, to solve a linear programming problem for each scenario considered. Although also affected by the size of the instances, the speed of this step mainly depends on the number of second-stage problems to solve, that is, the number of scenarios considered. The distribution separation problem does not depend on the size of the instances, only the number of scenarios is relevant.

This behaviour analysis indicates that increasing the size of the instances - in particular the number of vessels - has a greater impact on the master problem than in both second-stage problems and distribution separation problem. Hence, the main concern when solving large size instances is defining strategies to reduce the time of the master problems. One of such strategies is to impose a time limit when solving each master problem. As mentioned in Section 5.3, the convergence of the decomposition algorithm may be slow because several optimality cuts added to the master problem are satisfied with $\theta = 0$. Hence, the increase of the lower bound is, in general, slow, which makes the proof of the optimality slow as well (see Appendix F). In practice, we verified that, in general, the optimal solution is found in the earlier iterations and most of the time of the DA is used to prove the optimality of such a solution. To illustrate this, we ran the DA for the large instances with 14 and 15 vessels for 120 seconds and computed the average gaps between the obtained solution and the optimal solution. The obtained results are reported in Table 11.

Table 11: Results of the decomposition algorithm with a time limit of 120 seconds for instances with 14 and 15 vessels.

	$\epsilon = 1$	$\epsilon = 5$	$\epsilon = 10$	*Average*
Gap(%)	5.1	1.6	0.0	2.2

The results reported in Table 11 clearly show that small computational times (and consequently few iterations) are usually needed to obtain optimal or near optimal solutions. Running the DA just by 120 seconds made it possible to obtain heuristic solutions with average optimality gaps of around 2.2%. Hence, imposing a global time limit on the DA for solving large size instances is a reliable heuristic application of the proposed DA.

Table 12 reports results for large size instances with a number of vessels ranging from 20 to 60, considering a global time limit of 3600 seconds and a time limit for each master problem of 100 seconds. Each row reports average results for a set of five randomly generated instances. Column J indicates the number of berths defined for each set of instances. This number is kept equal to 21 for the instances with 20 and 30 vessels; however, this value increases to 41 for the instances with more than 30 vessels because otherwise most of the instances are infeasible due to the fact that it is not possible to meet all the hard deadlines of the vessels. Columns *Cost* report the average cost of the obtained solutions, while columns TM and $\#it$ report, respectively, the average total time spent on solving the master problems and the average number of global iterations performed. The results were obtained by using a set of 50 scenarios randomly selected from the ones generated by the MBC uncertainty set. Note that the total number of scenarios defined by the MBC uncertainty set is greater than 3×10^{10} for instances with more than 30 vessels.

Table 12: Results for large size instances.

		$\epsilon = 1$			$\epsilon = 5$			$\epsilon = 10$		
N	J	$Cost$	TM	$\#it$	$Cost$	TM	$\#it$	$Cost$	TM	$\#it$
20	21	7.5	2466	638	8.9	2603	597	9.0	2367	546
30	21	29.7	3469	385	33.2	3515	369	34.0	3276	377
40	41	3.9	3384	471	4.7	3386	478	5.6	3385	472
50	41	5.7	2106	156	6.7	2112	153	8.6	3171	145
60	41	36.3	3578	128	34.4	3464	139	43.2	3533	124

The results in Table 12 clearly show that the proposed DA can be successfully adapted for solving large size instances by imposing a time limit on each master problem and a global time limit. We can also see that several master problems - specially the ones associated with instances with 20, 30, and 40 vessels - are solved to optimality before the imposed time limit because the total number of iterations multiplied by the time limit is greater than the total time spent on solving the master problems. Additionally, there are also two instances with 20 vessels and two instances with 50 vessels that were solved to optimality.

8 Conclusions and future research

The BAP under uncertainty has been extensively studied in the literature under different approaches: stochastic programming, robust optimization, fuzzy theory, and deterministic approaches. However, to the best of our knowledge, there are no studies on BAPs using distributionally robust optimization. In this paper, we proposed a distributionally robust optimization model and an exact decomposition algorithm for solving the BAP under uncertain handling times. A deep analysis of the subproblems and assumptions associated with the proposed algorithm is carried out and several improvement strategies to make the algorithm more efficient are discussed. Our experiments clearly show the positive effect of using some of those improving strategies because they lead to a huge reduction of the required computational time.

We consider the well-known Kantorovich ambiguity set where a parameter ϵ is used to control the space of the allowed probability distributions in terms of the distance to a reference probability distribution. We conducted extensive computational experiments on instances adapted from the literature to understand the impact of the parameter ϵ in terms of the required computational time, number of iterations, and cost of the obtained solutions. The results obtained show that the number of iterations performed by the proposed algorithm decreases as the value of ϵ increases, which may explain why the total computational time also decreases. As expected, the cost of the obtained solutions has the opposite behaviour because it increases as the value of ϵ increases.

The distributionally robust optimization algorithm proposed makes it possible to establish direct relations with stochastic programming, by setting $\epsilon = 0$, and with robust optimization, by setting $\epsilon = \infty$. Hence, in this paper, we also compare distributionally robust optimization with stochastic programming and robust optimization under different situations. We concluded that these approaches can lead to structurally different solutions and, consequently, they can achieve different performances in different situations. In general, we concluded that the distributionally robust optimization makes it possible to obtain solutions that are simultaneously more protected against uncertainty than the ones obtained by stochastic programming and that are less conservative than the ones obtained by robust optimization. In particular, the distributionally robust approach with $\epsilon = 1$ performs, in general, better than the stochastic approach, while the distributionally robust approach with $\epsilon = 5$ is very competitive when compared with the robust approach.

As future work, we aim to extend the proposed algorithm and approach to more complex problems, such as the berth allocation and quay crane assignment/scheduling problem (BACASP); however, such extension is not trivial. First, the proposed algorithm is only valid for problems where the second-stage decisions are continuous variables, which is not the case of the BACASP. Second, the proposed algorithm requires to compute at each iteration a second-stage solution for each scenario of uncertainty, which is very time consuming, as reported in [28]. Lastly, each master problem in the BACASP is much harder to solve than in the BAP. The hardness of the resulting master problem increases as the number of iterations increases, which may be an obstacle to the application of the proposed distributionally robust optimization algorithm to the BACASP.

To study strategies to reduce the number of global iterations performed by the proposed algorithm and to study strategies to eliminate unnecessary probability distributions from the ambiguity set would also be an interesting research direction. The development of methods, perhaps based on machine learning techniques, to choose the most suitable parameter ϵ in the DRO model in terms of the scenarios of uncertainty considered is also a research line that is worth to be explored.

Acknowledgements

The research of the first author was partially supported by the Center for Research and Development in Mathematics and Applications (CIDMA) through the Portuguese Foundation for Science and Technology (FCT), references UIDB/04106/2020 and UIDP/04106/2020. The research of the second author was also partially supported by the Project CEMAPRE/REM - UIDB/05069/2020 - financed by FCT/MCTES through national funds.

References

[1] A. Agra, M. Christiansen, L. M. Hvattum, and F. Rodrigues. Robust optimization for a maritime inventory routing problem. *Transportation Science*, 52(3):509–525, 2018.

[2] A. Agra and M. Oliveira. Mip approaches for the integrated berth allocation and quay crane assignment and scheduling problem. *European Journal of Operational Research*, 264(1):138–148, 2018.

[3] A. Al-Refaie and H. Abedalqader. Optimal berth scheduling and sequencing under unexpected events. *Journal of the Operational Research Society*, pages 1–15, 2020.

[4] G. Alsoufi, X. Yang, and A. Salhi. Robust berth allocation using a hybrid approach combining branch-and-cut and the genetic algorithm. In Blesa M. et al., editor, *Hybrid Metaheuristics. Lecture Notes in Computer Science*, volume 9668, (Springer International Publishing), pages 187–201, 2016.

[5] M. Bansal, K.-L. Huang, and S. Mehrotra. Decomposition algorithms for two-stage distributionally robust mixed binary programs. *SIAM Journal on Optimization*, 28(3):2360–2383, 2018.

[6] D. Bertsimas and M. Sim. The price of robustness. *Operations Research*, 52:35–53, 2004.

[7] A. Budipriyanto, B. Wirjodirdjo, N. Pujawan, and S. Gurning. Berth allocation problem under uncertainty: A conceptual model using collaborative approach. *Procedia Manufacturing*, 4:429 – 437, 2015. Industrial Engineering and Service Science 2015, IESS 2015.

[8] A. Budipriyanto, B. Wirjodirdjo, N. Pujawan, and S. Gurning. A simulation study of collaborative approach to berth allocation problem under uncertainty. *The Asian Journal of Shipping and Logistics*, 33(3):127 – 139, 2017.

[9] Y. Du, Y. Xu, and Q. Chen. A feedback procedure for robust berth allocation with stochastic vessel delays. In *2010 8th World Congress on Intelligent Control and Automation*, pages 2210–2215, 2010.

[10] P. M. Esfahani and D. Kuhn. Data-driven distributionally robust optimization using the wasserstein metric: performance guarantees and tractable reformulations. *Mathematical Programming*, 171:115–166, 2028.

[11] R. Gao and A. J. Kleywegt. Distributionally robust stochastic optimization with wasserstein distance. Technical report, arXiv preprint arXiv:1604.02199v2, 2016.

[12] M. Golias, I. Portal, D. Konur, E. Kaisar, and G. Kolomvos. Robust berth scheduling at marine container terminals via hierarchical optimization. *Computers & Operations Research*, 41:412–422, 2014.

[13] Y. Guan and R. Cheung. The berth allocation problem: models and solution methods. *OR Spectrum*, 26:75–92, 2004.

[14] E. Guevara, F. Babonneau, T. H. de Mello, and S. Moret. A machine learning and distributionally robust optimization framework for strategic energy planning under uncertainty. *Applied Energy*, 271:115005, 2020.

[15] S. Guo and H. Xu. Distributionally robust shortfall risk optimization model and its approximation. *Math. Program.*, 174(1-2):473–498, 2019.

[16] F. Gutiérrez, E. Luján, R. Asmat, and E. Vergara. Fuzziness in the berth allocation problem. In S. Fidanova, editor, *Recent Advances in Computational Optimization. Studies in Computational Intelligence*, volume 795, pages 149–174, 2019.

[17] S. M. H. Rahimian. Distributionally robust optimization: a review. Technical report, arXiv preprint arXiv:1908.05659, 2019.

[18] R. Ji, M. A. Lejeune, and Z. Fan. Distributionally robust portfolio optimization with linearized starr performance measure. *Quantitative Finance*, 22(1):113–127, 2022.

[19] S. Jia, C.-L. Li, and Z. Xu. A simulation optimization method for deep-sea vessel berth planning and feeder arrival scheduling at a container port. *Transportation Research Part B: Methodological*, 142:174 – 196, 2020.

[20] J. Karafa, M. M. Golias, S. Ivey, G. K. D. Saharidis, and N. Leonardos. The berth allocation problem with stochastic vessel handling times. *The International Journal of Advanced Manufacturing Technology*, 65:473–484, 2013.

[21] A. Lim. The berth planning problem. *Operations Research Letters*, 22(2):105–110, 1998.

[22] C. Liu. A note on tactical berth allocation under uncertainty. *European Journal of Operational Research*, 278(1):363 – 364, 2019.

[23] C. Liu, X. Xiang, C. Zhang, and L. Zheng. A decision model for berth allocation under uncertainty considering service level using an adaptive differential evolution algorithm. *Asia-Pacific Journal of Operational Research*, 33(6):615–627, 2016.

[24] C. Liu, X. Xiang, and L. Zheng. Two decision models for berth allocation problem under uncertainty considering service level. *Flexible Services and Manufacturing Journal*, 29:312–344, 2017.

[25] C. Liu, X. Xiang, and L. Zheng. A two-stage robust optimization approach for the berth allocation problem under uncertainty. *Flexible Services and Manufacturing Journal*, 32(2):425–452, 2020.

[26] R. Moorthy and C. Teo. Berth management in container terminal: the template design problem. *OR Spectrum*, 28:495–518, 2006.

[27] B. Pérez-Cañedo, J. L. Verdegay, A. Rosete, and E. R. Concepción-Morales. Fully fuzzy multi-objective berth allocation problem. In E. A. de la Cal, J. R. V. Flecha, H. Quintián, and E. Corchado, editors, *Hybrid Artificial Intelligent Systems*, pages 261–272, Cham, 2020. Springer International Publishing.

[28] F. Rodrigues and A. Agra. An exact robust approach for the integrated berth allocation and quay crane assignment problem under uncertain arrival times. *European Journal of Operational Research*, 295(2):499–516, 2021.

[29] F. Rodrigues and A. Agra. Berth allocation and quay crane assignment/scheduling problem under uncertainty: a survey. *European Journal of Operational Research*, 2022. To appear.

[30] H. Scarf. A min-max solution of an inventory problem. In H. Scarf, K. Arrow, and S. Karlin, editors, *Studies in the mathematical theory of inventory and production*, volume 10, pages 201–209. Stanford University Press, Stanford, CA, 1958.

[31] X. Schepler, N. Absi, D. Feillet, and E. Sanlaville. The stochastic discrete berth allocation problem. *EURO Journal on Transportation and Logistics*, 8:363–396, 2019.

[32] F. G. Segura, E. L. Segura, E. V. Moreno, and R. A. Uceda. A fully fuzzy linear programming model to the berth allocation problem. In *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pages 453–458, 2017.

[33] A. Sheikholeslami and R. Ilati. A sample average approximation approach to the berth allocation problem with uncertain tides. *Engineering Optimization*, 50(10):1772–1788, 2018.

[34] N. Umang, M. Bierlaire, and A. L. Erera. Real-time management of berth allocation with stochastic arrival and handling times. *Journal of Scheduling*, 20:67–83, 2017.

[35] E. Ursavas and S. X. Zhu. Optimal policies for the berth allocation problem under stochastic nature. *European Journal of Operational Research*, 255(2):380 – 387, 2016.

[36] A. Wolsey. *Integer Programming.* Wiley, second edition, 2020.

[37] Y. Wu and L. Miao. A robust scheduling model for continuous berth allocation problem under uncertainty. In *2020 5th International Conference on Electromechanical Control Technology and Transportation (ICECTT)*, pages 43–49, 2020.

[38] Y. Wu and L. Miao. An efficient procedure for inserting buffers to generate robust berth plans in container terminals. *Discrete Dynamics in Nature and Society*, 2021:9 pages, 2021.

[39] X. Xiang and C. Liu. An almost robust optimization model for integrated berth allocation and quay crane assignment problem. *Omega*, 104, 2021.

[40] X. Xiang and C. Liu. An expanded robust optimisation approach for the berth allocation problem considering uncertain operation time. *Omega*, 2021. To Appear.

[41] X. Xiang, C. Liu, and L. Miao. A bi-objective robust model for berth allocation scheduling under uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, 106:294 – 319, 2017.

[42] X. Xiang, C. Liu, and L. Miao. Reactive strategy for discrete berth allocation and quay crane assignment problems under uncertainty. *Computers & Industrial Engineering*, 126:196–216, 2018.

[43] Z. Zhang, S. Ahmed, and G. Lan. Efficient algorithms for distributionally robust stochastic optimization with discrete scenario support. *SIAM Journal on Optimization*, 31(3):1690–1721, 2021.

[44] L. Zhen. Tactical berth allocation under uncertainty. *European Journal of Operational Research*, 247(3):928–944, 2015.

[45] P. Zhou, H. Kang, and L. Lin. A dynamic berth allocation model based on stochastic consideration. In *2006 6th World Congress on Intelligent Control and Automation*, volume 2, pages 7297–7301, 2006.

Appendix

A - Stochastic model

The stochastic model is defined as follows:

$$
\begin{aligned}
\min \quad & \frac{1}{|\Omega|}\sum_{\omega\in\Omega}\sum_{k\in V} c_k^\omega && && (49)\\
s.t. \quad & x_{\ell k} + x_{k\ell} + y_{\ell k} + y_{k\ell} \geq 1, && k,\ell \in V, k<\ell, && (50)\\
& x_{\ell k} + x_{k\ell} \leq 1, && k,\ell \in V, k<\ell, && (51)\\
& y_{\ell k} + y_{k\ell} \leq 1, && k,\ell \in V, k<\ell, && (52)\\
& b_k \geq b_\ell + L_\ell + J(y_{k\ell}-1), && k,\ell \in V, k\neq\ell, && (53)\\
& b_k \leq J - L_k, && k\in V, && (54)\\
& b_k \in \mathbb{Z}_0^+, && k\in V, && (55)\\
& x_{k\ell}, y_{k\ell} \in \{0,1\}, && k,\ell\in V, k\neq\ell && (56)\\
& t_\ell^\omega \geq t_k^\omega + H_k + F + M(x_{k\ell}-1), && k,\ell\in V, k\neq\ell, \omega\in\Omega, && (57)\\
& c_k^\omega \geq t_k^\omega + H_k - D_k, && k\in V, \omega\in\Omega, && (58)\\
& c_k^\omega \leq C, && k\in V, \omega\in\Omega, && (59)\\
& t_k^\omega \geq A_k, && k\in V, \omega\in\Omega, && (60)\\
& t_k^\omega, c_k^\omega \in \mathbb{R}_0^+, && k\in V, \omega\in\Omega. && (61)
\end{aligned}
$$

B - Proof of Proposition 5.1

Proof. Let $(\overline{u}, \overline{r}, \overline{z}, \overline{v})$ denote the optimal solution of the dual problem. Assume that, for a $k \in V$, inequality (33) is not satisfied in the equality, that is,

$$\overline{r}_k^\omega + \sum_{\ell\in V,\ell\neq k} \overline{u}_{\ell k}^\omega < \overline{z}_k^\omega + \sum_{\ell\in V,\ell\neq k} \overline{u}_{k\ell}^\omega.$$

The objective function coefficient of r_k^ω, that is, A_k, is positive. Hence, because r_k^ω has no upper bound, we can increase its value by $\overline{z}_k^\omega + \sum_{\ell\in V,\ell\neq k} \overline{u}_{k\ell}^\omega - \overline{r}_k^\omega - \sum_{\ell\in V,\ell\neq k} \overline{u}_{\ell k}^\omega$ and keep the remaining variables unchanged. This makes it possible to obtain a better solution, contradicting the assumption that $(\overline{u}, \overline{r}, \overline{z}, \overline{v})$ is optimal. □

C - Proof of Proposition 5.2

Proof. The dual problem has the following matricial structure:

$$\begin{matrix} r & u & z & v \end{matrix}$$
$$\begin{pmatrix} 0 & 0 & I & -I \\ I & B & -I & 0 \end{pmatrix}$$

Since equations (46) are flow conservation constraints, the submatrix $\begin{pmatrix} I & B & -I \end{pmatrix}$ is TUM. Hence the sub matrix $\begin{pmatrix} 0 & 0 & I \\ I & B & -I \end{pmatrix}$ is also TUM because we are adding an identity matrix to a TUM matrix. For similar arguments the complete matrix is TUM, see [36] for details. □

D - Proof of Proposition 5.4

Proof. First, we assume that in any dual solution, for each $k \in V$ with z_k^ω positive, the corresponding constraint (32) is satisfied as equation because the objective function coefficient of variable v_k^ω is negative. Consider such a solution with value $\overline{Z}$. Assume that for a given vessel k we have $v_k^\omega = \kappa > 0$. Thus, from constraints (32), it follows that $z_k^\omega = 1 + \kappa$. Assume, from Proposition 5.2, that κ is integer. Using the flow constraints (46) we can identify paths starting on O and ending in D using the arc associated with z_k^ω (that is, we can identify a set of dual variables with positive integer value corresponding to those paths). Choose the path, that is, a flow of one unit, from node O to node D, with highest sum of coefficients in the objective function. Let C^* denote that sum. Recalling that C is the coefficient of variables v_k^ω, two cases can occur:

Case 1. $C^* > C$. In this case we can increase the value of v_k^ω and the value of the variables associated with the path in one unit. The new solution has an objective function value $Z' = \overline{Z} + (C^* - C) > \overline{Z}$. The increase can be repeated without any bound. This means that a ray has been identified and the solution is improved along this ray. Thus, the dual problem is unbounded.

Case 2. $C^* \leq C$. In this case, by decreasing the value of v_k^ω and the value of the variables associated with the path in one unit, we obtain a new solution to the dual problem with objective function value $Z' = \overline{Z} - (C^* - C) \geq \overline{Z}$. Thus, this solution is at least as good as the original one. The process can be repeated until a solution with $v_k^\omega = 0$ is obtained (observe that $z_k^\omega = 1 + v_k^\omega$). Note that different paths can be used in each iteration but, by assumption, the sum of the coefficients in the objective function of each one of these paths is at most C^*. Thus, in each iteration, the value of the objective function does not decrease. Hence, either the final solution is improved or an alternative optimal solution is obtained with $v_k^\omega = 0$. □

E - Proof of Proposition 5.5

Proof. By construction, for each scenario ω, $X^\omega \leq |U^\omega|$. Two cases can occur. Either $X^\omega = |U^\omega|$, or $X^\omega \leq |U^\omega| - 1$ because X^ω is integer. In the first case, the term S^ω is added to the RHS for scenario ω, which is exactly the value obtained using the dual solution $(\overline{r}_k^\omega, \overline{u}_{k\ell}^\omega, \overline{z}_k^\omega, 0)$. This follows directly from the definition of S^ω. In the second case, $S^\omega - S^\omega(|U^\omega| - X^\omega) \leq 0$. Thus, the term associated with scenario ω is non positive, which is implied by the dual solution consisting of the null vector (which leads to a term of zero associated to the scenario ω). □

F - Results obtained without strategy S_x

Table 13 reports the results obtained for all instances with a number of vessels ranging from 6 to 13, and they show the impact of using strategy S_x. Columns *Gap* report the average optimality gap, in percentage, between the solutions obtained without strategy S_x (optimal solutions) and the solutions obtained with strategy S_x. Columns *Opt* report the percentage of instances for which the use of strategy S_x made it possible to obtain the optimal solution. Columns S_x and $\overline{S_x}$ report the average computation time, in seconds, required to obtain the final solution when the strategy S_x is used and when it is not used, respectively.

Table 13: The impact of using strategy S_x on instances with a small number of vessels.

	$\epsilon = 1$				$\epsilon = 5$				$\epsilon = 10$			
N	*Gap*	*Opt*	S_x	$\overline{S_x}$	*Gap*	*Opt*	S_x	$\overline{S_x}$	*Gap*	*Opt*	S_x	$\overline{S_x}$
6	2.2	90	0	0	2.2	90	0	0	2.2	90	0	0
7	0.0	100	0	1	0.0	100	1	1	0.0	100	0	0
8	0.0	100	1	2	0.0	100	1	2	0.0	100	2	2
9	0.7	90	6	11	2.9	80	6	9	2.9	80	5	8
10	0.5	90	33	110	5.8	80	37	103	5.8	80	29	90
11	6.1	80	44	137	5.0	80	50	150	5.3	80	42	148
12	5.6	70	434	1135	4.2	70	441	1231	3.1	70	325	822
13	6.0	70	1397	5160	6.4	80	951	4000	6.5	80	802	2605

The obtained results show that the percentage of instances for which the use of strategy S_x does not cut optimal solutions is around 85%. In particular, the solutions obtained with and without strategy S_x are the same for the 10 instances with 7 and 8 vessels. The average gaps between the solutions obtained with strategy S_x and the solutions obtained without strategy S_x are around 3.3%. However, it is important to note that the total computational time is drastically reduced when strategy S_x is used.

As mentioned before, the use of strategy S_x in the proposed DA makes the DA a heuristic algorithm. A different strategy that can be used for solving hard instances with the proposed DA consists of imposing a global time limit on the DA instead of using strategy S_x. This time limit based heuristic (TL heuristic) makes it possible to determine not only an upper bound for the optimal solution but also a lower bound. Table 14 reports results for all instances with 14 and 15 vessels for the three different values of ϵ tested. Columns *LB* and *UB* report the lower bounds and the upper bounds obtained by using the TL heuristic, respectively. The time limit imposed for the TL heuristic varies from instance to instance because, for fairness, it was defined as the same time required for running the DA with strategy S_x until the end. Columns S_x report the value of the solutions determined by the DA combined with strategy S_x. The best upper bounds for each instance are marked in bold.

Table 14: Comparison between the results obtained with strategy S_x and with the TL heuristic for instances with 14 and 15 vessels.

	ϵ=1			ϵ=5			ϵ=10		
Inst.	*LB*	*UB*	S_x	*LB*	*UB*	S_x	*LB*	*UB*	S_x
R_14_1	0.2	**16.2**	16.4	2.0	**19.4**	**19.4**	0.0	**20.0**	**20.0**
R_15_1	0.0	10.4	**8.3**	0.0	11.5	**11.3**	0.0	**13.0**	**13.0**
R_14_2	0.0	**17.3**	**17.3**	0.0	21.6	**21.5**	0.0	**22.0**	**22.0**
R_15_2	0.0	3.3	**3.2**	0.0	6.6	**6.3**	0.0	**7.0**	**7.0**
R_15_3	5.3	**6.1**	6.2	6.7	**8.8**	**8.8**	4.0	**9.0**	**9.0**
R_14_4	0.0	**0.9**	**0.9**	0.0	**2.0**	**2.0**	0.0	**2.0**	**2.0**
R_15_4	0.0	8.2	**8.0**	0.0	11.0	**10.7**	0.0	**11.0**	**11.0**
R_14_5	0.0	4.2	**3.5**	0.0	**5.9**	7.0	0.0	**6.0**	8.0
R_15_5	0.2	**13.9**	15.6	1.3	**15.0**	18.5	1.0	**16.0**	19.0
R_14_6	0.0	1.4	**1.3**	0.0	**2.0**	**2.0**	0.0	**2.0**	**2.0**
R_15_6	0.0	**7.9**	**7.9**	0.0	**9.0**	**9.0**	0.0	**9.0**	**9.0**
R_15_7	0.0	**19.3**	**19.3**	0.0	**23.5**	**23.5**	0.0	**24.0**	**24.0**
R_14_8	0.0	**0.7**	**0.7**	0.0	**1.0**	**1.0**	0.0	**1.0**	**1.0**
R_15_8	0.0	**10.0**	11.0	0.0	**13.0**	14.6	0.0	**13.0**	15.0
R_14_9	0.0	15.1	**14.4**	4.8	**18.9**	19.7	0.0	**20.0**	21.0
R_15_9	0.0	**1.0**	**1.0**	0.0	**2.8**	**2.8**	0.0	**3.0**	**3.0**
R_14_10	6.5	**11.7**	**11.7**	9.3	**13.9**	**13.9**	3.0	**14.0**	**14.0**
R_15_10	0.0	1.7	**1.2**	0.0	**2.0**	**2.0**	0.0	**2.0**	**2.0**

The obtained results do not clearly show which of the strategies is the best in terms of the obtained upper bounds because each heuristic is outperformed by the other in some instances and it outperforms the other in other instances. However, a clear conclusion that can be drawn from the obtained results is that there are very few instances for which the lower bound obtained by the TL heuristic is greater than zero. This means that the lower bounds generated by the TL heuristic are not suitable values for determining the optimality gap of the obtained solutions.