FEDERAL UNIVERSITY OF RIO GRANDE DO SUL - UFRGS
MANAGEMENT SCHOOL
POSTGRADUATE PROGRAM IN MANAGEMENT

PIETRO TIARAJU GIAVARINA DOS SANTOS

ESSAYS ON MARITIME CARGO ROUTING AND SCHEDULING PROBLEM

PORTO ALEGRE
2024

Pietro Tiaraju Giavarina dos Santos

ESSAYS ON MARITIME CARGO ROUTING AND SCHEDULING PROBLEM

This thesis was developed as a partial requirement to obtain the PhD degree in Managament of Federal University of Rio Grande do Sul.

Supervisor: Prof. Dr. Denis Borenstein

Porto Alegre
2024

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

INDEX

1 INTRODUCTION

This work is a three-article compilation about the Maritime Cargo Routing and Scheduling Problem and its variants. These articles are listed as follows:

- Cargo routing and scheduling problem in deep-sea transportation: Case study from a fertilizer company. Published in Computers & Operations Research (2020);
- Multi-objective optimization of the maritime cargo routing and scheduling problem. Published in International Transactions on Operations Research (2022);
- The segregated storage multi-ship routing and scheduling problem. In review by Information Systems and Operational Research (2023).

The *"Cargo routing and scheduling problem in deep-sea transportation: Case study from a fertilizer company"* was initially motivated by a real-life problem faced by the Brazilian branch of a Norwegian chemical company. The problem consists in generating an operational ship routing and scheduling planning of multi-bulk fertilizers from European ports to meet an estimated demand of fertilizer mixer units in Brazil. The main objective proposed in I is to generate a minimum cost plan, subject to a set of constraints and operational requisites. This plan is used by the company to define the amounts to buy of each fertilizer, and their respective collection ports in Europe, as well as to charter maritime transportation from shipping agencies/companies. The multiple product bulk cargo is transported by a fleet of tramp bulk vessels, where each ship is able to carry different raw materials at the same time, using one compartment for each product. First, the products are collected in different ports in Europe. The cargo is then discharged in one or more destination Brazilian ports. No transshipment among origin and destination ports is allowed. The planning specifies each vessel routing and scheduling, and the amount of each product loaded/unloaded at each visited port, called as route. However, the planning is not responsible for product assignment to compartments, since the vessel's master is responsible for this task, taking into account issues such as vessel stability and structural strength.

Deep sea bulk cargo pickup and delivery is a complex operation due to several reasons. The high costs involved (such as vessel hiring and cargo handling in ports) impose the selection of a plan able to deliver the fertilizers with the shortest cost and time. Further, the problem is subject to several constraints and requirements, especially when multiple products, ports, and ships are involved (ARNESEN et al., 2017). Ports and ships have strict capacities and draft limits that should be respected. Products cannot be mixed in the ship compartments. The demand of destination ports should also be met and preferentially within a time window (TW) specified by each production unit, imposing time limits in the long-time consuming loading/unloading operations in ports. The real-problem investigated in this work presents some characteristics of the static pickup and delivery problem with TWs (PDPTW) as described by Savelsbergh e Sol (1995), a well-known extension of the classic Vehicle Routing Problem (VRP).

The PDPTW and its variants, such as multiple type vehicles (m-PDPTW), have attracted the attention of several researchers in the last decades. Parragh et al. (2008) present a comprehensive survey of the problem and its variants. However, the majority of the research on PDPTW is directed to road-based applications, with different attributes of the maritime problem, in terms of the number of origins and destinations to visit, the time horizon planning, the small quantity of items to be delivered, in comparison with the vehicle capacity (PAPAGEORGIOU et al., 2014). Further, on one hand, a vessel route can have a flexible structure P-P-D-P-P-D-D in the PDPTW, where P denotes a pickup location, and D a delivery location. This is not the case of the deep-sea cargo routing and scheduling problem, where a vessel route has a more rigid structure P-P-P-P-D-D-D, making our problem simpler to solve in comparison with the PDPTW. On the other hand, our problem presents flexible cargo sizes and split loads as characterized in Fagerholt e Ronen (2013), making the problem more complicated to solve in comparison with the PDPTW. As a consequence, the deep-sea cargo routing and scheduling problem requires specific formulations and solution methods, capable of incorporating the peculiar aspects involved in each application. Article I considers a real-life short-term ship routing and scheduling problem of a fertilizer company. The problem was formulated as a mixed integer linear programming (MILP) model with the objective of minimizing costs, based on the m-PDPTW, incorporating several additional constraints, representing the particular requisites of the problem. The resulting formulation can be solved for small-medium size instances using a powerful MILP solver.

Given the complexity of the problem, a matheuristic method was developed for solving large instances. The matheuristic comprises of three steps, namely a relaxation algorithm phase, a modified relax-and-fix (RaF) algorithm, and a post-optimization process. The optimization approach was evaluated using real cases provided by the company, not only offering significantly better solutions than than human schedulers, but also reducing the solution process time from months/days to hours/minutes.

In *“Multi-objective optimization of the maritime cargo routing and scheduling problem”* is introduced a new problem class in the context of maritime transportation, the multi-objective, multi-commodity heterogeneous fleet cargo routing and scheduling problem with time windows, draft limits, and split loads (MO-m-CRSPTW-DL-SL) which generalizes the CRSP characterized by Christiansen et al. (2013). In the MO-m-CRSTW-DL-SL setting, a heterogeneous fleet of vessels operates highly constrained routes to load multiple bulk products from a set of pickup ports and unload them in a set of delivery ports, fulfilling a deterministic demand from a set of customers. Early or late arrivals/departures of a vessel in a port are very costly, and therefore the synchronization of vehicle scheduling and port operating time windows is an important characteristic of the problem. Also, delays in deliveries may cause disruptions in the companies' manufacturing process, jeopardizing the productivity of customers. The nature of the products prevents them from being mixed, they must be transported in dedicated compartments of the vessels. Further, the routes are constrained by several operational requirements such as draft limits and berth utilization of the ports. The solution of the problem specifies (i) the sequence of ports to be visited by each vessel; (ii) the amount of each product to be loaded/unloaded in each port by a vessel; and (iii) the arrivals and departures schedule of the vessels in/from ports. However, as the vessel master is solely responsible for allocating the products in the compartments (CHRISTIANSEN et al., 2011; STANZANI et al., 2018), due to the stability and structural strength of each vessel, and sea conditions on the route, this task is partially addressed in the route planning, only guaranteeing that the transported diversity and amounts of products respect the number and nominal capacity of the ships' compartments, respectively. Furthermore, cargo allocation is a very difficult sub-problem to solve in the context of the CRSP (HVATTUM et al., 2009). The overall plan should simultaneously minimize total transportation costs, scheduling makespan, and delays in some deliveries.

The MO-m-CRSPTW-DL-SL is a maritime variant of the multi-objective pickup and delivery problem with time windows (MO-PDPTW) (DUMAS et al., 1991), a vehicle routing problem (VRP). Although the single objective of minimizing the cost is still dominant in the VRP literature (BRAEKERS et al., 2016), the problem is multi-objective in nature (JOZEFOWIEZ et al., 2008). In real life, decision-makers (DMs) consider additional objectives beyond costs, such as the optimization of the number of customer visits, the minimization of total lengths, and optimization of the makespan. There is crescent literature in the MO-VRP, and also in the MO-m-PDPTW. In a succinct analysis, the multi-objective approach is formulated by introducing extensions or adaptations to the single-objective modeling. The MO-VRP has been solved using (JOZEFOWIEZ et al., 2008): (i) scalar methods, mainly weighted aggregation using local search algorithms (PAQUETE; STÜTZLE, 2003), specific heuristics (ZOGRAFOS; ANDROUTSO-

POULOS, 2004), and genetic algorithms (OMBUKI et al., 2006); (ii) Pareto dominance methods, mainly using multi-objective evolutionary (WANG et al., 2016; BRAVO et al., 2019), and hybrid algorithms (ZHANG et al., 2020); and (iii) non-scalar and non-Pareto algorithms, which includes ant colony systems (ZHANG et al., 2019), and particle swarm (ZOU et al., 2013) optimization.

Studies in using multi-objective in the maritime routing and scheduling problem are still scarce. In a review by Mansouri et al. (2015) about the consideration of multi-objective decisions in sustainable maritime shipping, no explicit multi-objective optimization (MOO) approach was cited. Multiple objectives were mainly considered as constraints. Chan et al. (2014) developed a dynamic scheduling of oil tankers with the splitting of cargo at pickup and delivery ports, using a multi-objective ant colony-based approach. The developed algorithm proved very efficient in comparison with a non-dominated sorting genetic algorithm II (NSGA II) towards finding good solutions for instances with dozens of pickup and delivery ports with a heterogeneous fleet of oil tankers. Although containing interesting ideas, the problem considered by Chan et al. (2014) is much less restrained than the problem tackled in this paper, neglecting time windows, port drafts, and dedicated compartments. The problem only considers a single product. Recently, MOO approach for planning liner shipping service considering uncertain port times were introduced by Song et al. (2015). The problem was formulated as a stochastic nonlinear programming model, considering three objectives, as follows: (i) annual total vessel operating costs; (ii) average schedule unreliability; and (iii) CO_2 emissions. The model was solved using an NSGA II algorithm and applied to a container shipping service route. De et al. (2017) developed a bi-objective model addressing the sustainable ship routing and scheduling with time windows and draft restrictions, maximizing the overall profit incurred of providing shipping operations within a planning horizon, and minimizing the total carbon emission incurred by the ship fleet. The model was solved by combining NSGA-II and multi-objective particle swarm optimization (MOPSO). However, both problems are directed to container ships, which are large ocean vessels that operate, in general, as line service, transporting goods using regular transit routes on fixed schedules. Further, cargoes are aggregated in containers, without considering the products individually. To the best of our knowledge, no previous research work uses MOO for handling the maritime CRSP.

"The segregated storage multi-ship routing and scheduling problem" presents a Lagrangian relaxation (LR)-based solution method for the m-CRSP-TW-SS-SL. LR is widely used in solving hard integer programming problems (FISHER, 1981). The central concept is to decompose the problem into two different types of constraints: "hard" and "soft". The hard

constraints are incorporated into the objective function so that they are penalized by the corresponding Lagrangian multipliers. The resulting relaxed problem should be easier to solve, offering reasonable bounds for the original problem. The LR-based method is developed using an expanded formulation of the MILP presented in Santos e Borenstein (2022), incorporating segregated storage and ship stability constraints. Experiments were conducted in real-world instances, and in a case study in one of the largest fertilizer companies in Brazil. The results show that our Lagrangian approach effectively and efficiently solves the m-CRSP-TW-SS-SL.

2 CARGO ROUTING AND SCHEDULING PROBLEM IN DEEP-SEA TRANSPORTATION: CASE STUDY FROM A FERTILIZER COMPANY

ABSTRACT

This study presents a mixed-integer linear programming (MILP) model and a solution method for a maritime cargo routing and shipping problem faced by a chemical company in Brazil. This problem is associated with the operational planning of multiple raw materials, collected from European ports and delivered to Brazilian ones to supply mixing production plants. First, the problem is modeled as a pickup and delivery problem with time windows, incorporating several constraints and operational requisites of the specific problem. In order to solve large real-world instances, a matheuristic was developed, employing a modified relax-and-fix strategy, a relaxation procedure, and repair and polishing routines for MILP solutions. The matheuristic was evaluated using real-life instances provided by the company, demonstrating the efficiency and efficacy of the developed solution method.

2.1 INTRODUCTION

This study was motivated by a real-life problem faced by the Brazilian branch of a Norwegian chemical company. The problem consists in generating an operational ship routing and scheduling planning of multi bulk fertilizers from European ports to meet an estimated demand of fertilizer mixer units in Brazil. The main objective is to generate a minimum cost plan, subject to a set of constraints and operational requisites. This plan is used by the company to define the amounts to buy of each fertilizer, and their respective collection ports in Europe, as well as to charter maritime transportation from shipping agencies/companies. The multiple product bulk cargo is transported by a fleet of tramp bulk vessels, where each ship is able to carry different raw materials at the same time, using one compartment for each product. First, the products are collected in different ports in Europe. The cargo is then discharged in one or more destination Brazilian ports. No transshipment among origin and destination ports is allowed. The planning specifies each vessel routing and scheduling, and the amount of each product loaded/unloaded at each visited port, called as route. However, the planning is not responsible for product assignment to compartments, since the vessel's master is responsible for this task, taking into account issues such as vessel stability and structural strength.

Deep sea bulk cargo pickup and delivery is a complex operation due to several reasons. The high costs involved (such as vessel hiring and cargo handling in ports) impose the selection of a plan able to deliver the fertilizers with the shortest cost and time. Further, the problem is subject to several constraints and requirements, especially when multiple products, ports, and ships are involved Arnesen et al. (2017). Ports and ships have strict capacities and draft limits that should be respected. Products cannot be mixed in the ship compartments. The demand of destination ports should also be met and preferentially within a time window (TW) specified by each production unit, imposing time limits in the long-time consuming loading/unloading operations in ports.

The real-problem investigated in this work presents some characteristics of the static pickup and delivery problem with TWs (PDPTW) as described by (SAVELSBERGH; SOL, 1995), a well-known extension of the classic Vehicle Routing Problem (VRP). The PDPTW and its variants, such as multiple type vehicles (m-PDPTW), have attracted the attention of several researchers in the last decades. (PARRAGH et al., 2008) present a comprehensive survey of the problem and its variants. However, the majority of the research on PDPTW is directed to road-based applications, with different attributes of the maritime problem, in terms of the number of origins and destinations to visit, the time horizon planning, the small quantity of items to be delivered, in comparison with the vehicle capacity Papageorgiou et al. (2014). Further, on one hand, a vessel route can have a flexible structure P-P-D-P-P-D-D in the PDPTW, where P denotes a pickup location, and D a delivery location. This is not the case of the deep-sea cargo routing and scheduling problem, where a vessel route has a more rigid structure P-P-P-P-D-D-D, making our problem simpler to solve in comparison with the PDPTW. On the other hand, our problem presents flexible cargo sizes and split loads as characterized in (FAGERHOLT; RONEN, 2013), making the problem more complicated to solve in comparison with the PDPTW. As a consequence, the deep-sea cargo routing and scheduling problem requires specific formulations and solution methods, capable of incorporating the peculiar aspects involved in each application.

This paper considers a real-life short-term ship routing and scheduling problem of a fertilizer company. The problem was formulated as a mixed integer linear programming (MILP) model with the objective of minimizing costs, based on the m-PDPTW, incorporating several additional constraints, representing the particular requisites of the problem. The resulting formulation can be solved for small-medium size instances using a powerful MILP solver. Given the complexity of the problem, a matheuristic method was developed for solving large instances. The matheuristic comprises of three steps, namely a relaxation algorithm phase, a modified relax-and-fix (RaF) algorithm, and a post-optimization process. The optimization approach was evaluated using real cases provided by the company, not only offering significantly better solutions than than human schedulers, but also reducing the solution process time from months/days to hours/minutes.

The contributions of this paper are as follows: (i) to introduce the multi-objective deep-sea maritime cargo routing and scheduling problem, simultaneously considering multi-product, heterogeneous fleet with dedicated compartment, draft limits, flexible cargo sizes, split loads, and time windows; (ii) to present a MILP formulation for the problem; (iii) to develop a matheuristic framework capable of offering good solutions in for the size range of evaluated real life instances.

The paper is organized as follows. Section 2.2 reviews the related literature on ship routing and scheduling problem. Section 4.3 describes the problem, explicitly defining its particular constraints and requisites. The developed mathematical formulation of the problem is presented in Section 2.4. The solution method is described in detail in Section 2.5. Section 2.6 describes the evaluation of the developed solution method using real-life instances. Finally, a summary of the results and areas of future research are provided in Section 2.7.

2.2 LITERATURE REVIEW

The literature on ship routing and scheduling has considerably increased in the last two decades Lin e Tsai (2014). After a surprisingly slow start, considering the relevance of maritime transportation in global trade, some of the ideas developed for ground transportation have been successfully applied to sea transportation. As this problem has several different applications, depending on the type of ship (liner, tramp or industry), goods transported, and port operations, the literature presents several different categorizations (CHRISTIANSEN et al., 2013). For convenience, we assume the classification provided by Christiansen et al. (2013) that separates the maritime routing and scheduling problem into *cargo routing and scheduling* and *inventory routing* problems. The former focuses on the cargo to be transported, specified by the demand and supply of the involved ports and the wished time windows (TWs) for loading and unloading the products; while the latter incorporates inventory management constraints to the routing problem. Below, we first succinctly relevant maritime inventory routing problem (MIRP) related to our case. Next, we discuss the cargo routing and scheduling models and solution methods, highlighting our contributions to this most closely related to the problem dealt in this study.

The MIRP is currently a very studied problem, as demonstrated by the surveys from Andersson et al. (2010) and Papageorgiou et al. (2014). As pointed out by Christiansen et al. (2013), the majority of research in MIR was initially directed to the transportation of a single product, in general oil or liquified natural gas (LNG) Christiansen et al. (2004). Several heuristic, metaheuristics, and optimization methods were developed to solve this problem. The first applied methods were developed using arc-flow and path-flow integer linear or nonlinear programming models, presenting some analogy with the road based formulations for the vehicle routing and scheduling problem. The former formulation generates feasible schedules, while the latter uses a set partitioning problem to define the final schedule. Column generation was initially the most widely technique to solve large problems Jetlund e Karimi (2004), Kobayashi e Kubo (2010). Al-Khayyal e Hwang (2007) extended the MIRP by studying a multi-product problem. The major contribution of this work is the reformulation of a non-linear integer programming formulation into an equivalent MILP by linearizing several constraints, process used by several subsequent studies. The introduction of multi-products significantly increased the complexity of the problem, introducing new solution methods, such as branch-and-cut and large neighborhood search (SONG; FURMAN, 2013), hybrid heuristics Agra et al. (2014), genetic algorithms Christiansen et al. (2011), relax-and-fix (UGGEN et al., 2013), and matheuristics Stanzani et al. (2018). Although the literature on MIRP has interesting and useful ideas towards solving the ship routing and scheduling problem, the developed models have focused on the inventories located in the visiting ports in extended horizon planning. However, in the case study here, the company is not worried about inventory levels outside the short-term planning horizon. The emphasis is on the delivery of demanded amounts of products within a time windows, minimizing the total costs involved. Therefore, our problem can be characterized as a classical maritime cargo routing and scheduling problem.

The cargo routing and scheduling problem is much less studied than the MIRP. Fagerholt (2001) developed one of the first studies in this problem, using a similar formulation approach mixing arc and path flow to study ship scheduling with soft TWs. The same described approach was used by some researchers to study the tramp ship routing and scheduling problem and its variant. In general, the problem is formulated as an arc-flow formulation based on the m-PDPTW, and solved using the path-flow model. Brønmo et al. (2007a) and Brønmo et al. (2010) solved the multi-ship pickup and delivery problem with time windows and flexible cargo sizes using a multi-start local search heuristic and a column generation approach, respectively. (BRØNMO et al., 2007b) solved a similar problem, using an enumeration process to generate *a priori* all schedules for each ship. Andersson et al. (2011) developed a similar decomposition and a priori generation of columns to solve the maritime Pickup and Delivery Problem with Time Windows and Split Loads (PDPTWSL), a problem introduced by Korsvik et al. (2011). Subsequently, Stålhane et al. (2012) developed a branch-and-cut method to solve the PDPTWSL. However, the complexity of generating all schedules *a priori* or the well-known convergence and stabilization problems with column generation Lübbecke e Desrosiers (2005) motivated researchers to solve the problem using heuristics, such as tabu search Korsvik et al. (2010), genetic algorithm Lin e Tsai (2014), and neighborhood search approach Korsvik et al. (2011), Malliappi et al. (2011).

Nevertheless, the above cited studies still neglect several aspects presented in real life problems, such as draft limits, multi-compartment cargo transportation, multi-products, and heterogeneous fleet. Despite introducing considerable complexity to the problem, the consideration of real features is fundamental towards solving practical problems Fagerholt e Ronen (2013). (FAGERHOLT; CHRISTIANSEN, 2000a) combined ship scheduling and the allocation problem (SSAP), solving them in an iterative way. The first problem is modeled as a Traveling Salesman Problem with Allocation, TW and Precedence Constraints (TSP-ATWPC), and solved by a dynamic programming algorithm for each vessel in the fleet. After a set of candidate schedules is generated for each ship, the SSAP is solved using a set partitioning problem. However, no draft limits were considered. Rakke et al. (2012) and Battarra et al. (2014) incorporated draft issues in the TSP, using branch and cut algorithms to solve the problem. (MALAGUTI et al., 2018) solved the TSP with pickup and delivery (TSPPD), integrating heuristic procedures and a branch-and-cut exact algorithm. However, the problem was modeled without incorporating TWs. Arnesen et al. (2017) expanded the TSPPD, considering TWs and draft limits (TSPPD-TWDL), using a solution approach based on the dynamic programming developed in Fagerholt e Christiansen (2000b). Rodrigues et al. (2016) developed an arc flow MILP to represent a single-product, heterogeneous fleet PDPTW with draft limit for maritime oil transportation. Given the difficulty of CPLEX to solve some instances, a relax-and-fix and a time decomposition based heuristics were developed. As expected, the heuristics were much more efficient, but with lower quality solutions, than CPLEX. The problem was simplified by considering that split loads are not allowed. As a result, the amounts of products to be picked up or delivered at each visiting site are modeled as constants, rather than decision variables. More recently, Trottier e Cordeau (2019) solved a short-sea vessel routing and scheduling problem for a fleet of dry bulk vessels in tramping operations, considering TWs, heterogeneous fleet, split cargoes, and draft limits. The problem was solved using tabu search, incorporating a constraint relaxation mechanism that avoids the need of a feasible initial solution. The method was successfully applied in a Canadian maritime company. As the modeling approach was developed to be used by a maritime company, the transported cargo was aggregated, without considering the products individually.

In this paper we expand the cargo routing and scheduling problems described in Arnesen et al. (2017) and Rodrigues et al. (2016) by considering a heterogeneous fleet with dedicated compartment, and multiple products in different ports, respectively. To the best of our knowledge, this is the first study on the multi-product, split-load, flexible size cargo, time-windows, and heterogeneous fleet with dedicated compartment on the deep-sea cargo routing and scheduling problem.

2.3 PROBLEM BACKGROUND AND DESCRIPTION

We consider the short-term planning of the cargo routing and scheduling from pickup ports in Europe to delivery ones in Brazil faced by a chemical company. This plan is carried out by the logistic department of the Brazilian branch to define the fertilizer supply of around twenty-four mixing units located in Brazil. Currently, the company uses EXCEL to define the plan, based on data and information from the its enterprise resource planning. Due to the complexity of the plan, the process starts 90 days before its actual implementation. The units are responsible to estimate the monthly consumption of each raw material. Based on this information and the experience of the logistics department, the plan is elaborated and made available in the company's enterprise resource planning system for the raw material buying process and vessel chartering by the Norwegian central administration. It is important to notice that this is a very strategic process, since the company has been increasing the participation of premium products in the Brazilian market. The main objective of the company is to generate a plan that minimizes the logistics costs, determining the number of vessels to be used, their routing and scheduling of collections and deliveries, in addition to the quantities of products which each vessel will be load/unload in each port of its route, taking into account the deadlines and demand of each raw material.

The planning horizon involves the deep-sea transportation of up to 12 different fertilizers stored in warehouses in ports near to the production plants, located in North Africa and Europe. Due to coordination issues, up to six origin ports can be visited by each planning horizon. The mixing units in Brazil can use up to 13 delivery ports by each planning horizon to receive their demanded raw material. Each pickup port can offer one or more products, while one or more delivery ports may have the need for several raw materials, with different quantities. Although some destinations ports may have several berths, only one berth can be used by the chartered vessels at any time, avoiding competition in the use of resources. Further, almost all origin and destination ports have severe draft limitations, imposing restrictions in its use given the load condition of a vessel. We assume that all terminals in a port have their respective draft limits defined by its Port Authority.

The vessels responsible for the deep-sea transportation are voyage chartered. In general, handymax or handysize bulk carriers are used, with capacity of around 40,000 DWT. Each vessel has several compartments that can carry different products. Since the demands of some mixing units are lower than the vessel capacity, a route may comprise the pickup and delivery of several products. To avoid low usage of the vessels, there are restrictions in the minimum load of 20 kt per vessel, and the value of four kt per product transported by a vessel. We assume that all vessels, independently of the size, navigate using an average cruise speed of fifty knots. As a consequence, the sailing times between ports are only dependent on the distances between them. The speed in each port is also constant and follows the Port Authority. Unfortunately, the vessel draft is quite complex to estimate without detailed real time information. Towards simplification, we assume that the vessel draft is fundamentally dependent on its current cargo, following (ARNESEN et al., 2017).

Feasible routes should comply with several requisites and constraints. Any vessel route should be based on a simple rule, all cargoes are picked up in European and African ports, and then delivered in their destination ports. Transshipment is neither allowed among origin ports, nor among destination ports, however partial loading/unloading of a vessel compartment by the same product is allowed. Another important requisite in a route is the delivery of fertilizers as close as possible to the date demanded for each mixing plant to fulfill its production plan. As a consequence, TWs are created for each delivery port. Further times should be considered when designing a route, as follows: loading/unloading time for every cargo, compartment washing time after delivering a cargo, waiting time for release of the product for transport on land, in accordance with customs legislation, and the waiting time for a vessel to dock at a terminal. Therefore, to respect the delivery TWs, it is necessary to establish a TW in the collection of products in each origin port of a route. Normally, a delivery TW has a total duration of ten to sixteen days. Although it is extremely important for the company to respect the specified TWs, sometimes it is difficult to find a viable plan capable of simultaneously respecting all values, giving the large number of operational constraints. As a result, TWs can be seen as soft constraints. The initial time of each route also defines the contractual laycan (abbreviation for laydays and canceling), a clause defining the TW in which the charterers are obliged to accept the vessel in the first loading port. If the vessel arrives before the first date agreed, the vessel probably has to wait. If the vessel arrives too late, the charterers are entitled to not accept the vessel. The laycan is around five days. Figure 1 illustrates possible vessel routes.

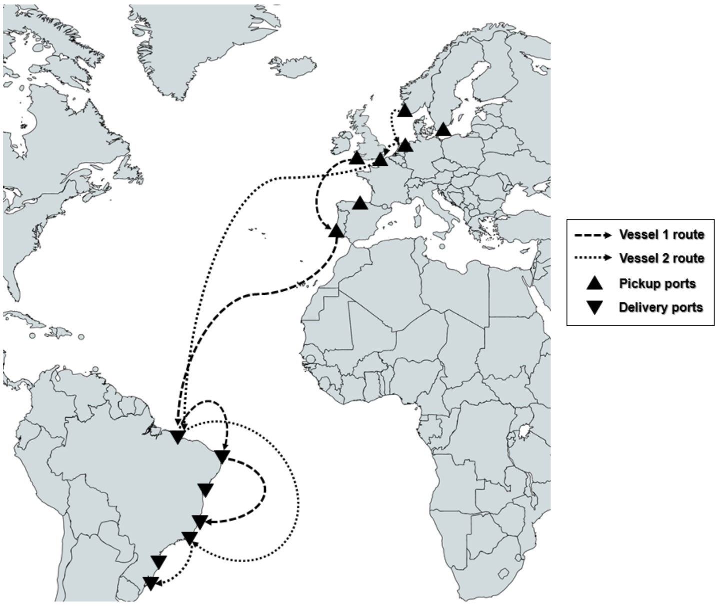


Figura 1 – Examples of vessel routes

As the company does not know the position of the vessels to be chartered, we assume that vessels start their routes from a pool that allows them to respect the laycan time. Another peculiarity of our problem is the adoption of a geographical orientation for the route. By adopting an orientation (for example North/South), the vessel may not change it until the end of its route, subject to a severe penalty in freight costs. In a route, vessels cannot return to an already visited port. Further, there is a limit for the number of collection ports in a route. This limit is associated with reducing the risk of delivery delays when the vessel starts its journey.

The following costs are taking into account in the minimization of total transportation costs of the fertilizers: (i) Chartering of the vessel, which varies according to the time used for each vessel; (ii) Demurrage costs, which are charges levied by the shipping charter to the company in cases where they have not taken delivery of the cargo and move it out of the port area within the allowed free days; (iii) Usage of the ports, which varies with the number of times each port is used by different vessels; (iv) Inversion of the geographical orientation of a vessel route; and (v) Fee for early or late arrival of a vessel in a port, penalizing the service at each port outside the given time window of each port. Note that storing costs in the ports are not explicitly considered in our modeling, since they are implicitly contemplated by item (v) above.

2.4 PROBLEM MODELING

The problem described below presents characteristics of an m-PDPTW. We used the formulations introduced by Al-Khayyal e Hwang (2007) and Stanzani et al. (2018) as a basis for our MILP formulation, specially the ingenious linearization processes describe in both studies. The peculiarities of our specific problem were introduced as constraints. As demands are considered fixed during the planning horizon, we used a continuous-time formulation as recommended by Agra et al. (2013) with the notation presented below. Sets and parameters are expressed as upper case letters, while variables use lower case letters.

Indices and Sets	
$h, i, j, k \in N^V$	set of pickup and delivery ports
$h, i, j, k \in N^P$	set of pickup ports
$h, i, j, k \in N^D$	set of delivery ports
$h, i, j, k \in N$	set of all ports, including the start (s) and finish (f) ports
$p \in P$	set of products
$v \in V$	set of vessels
Parameters	
Q_{ip}	stock level of product p in port i (ton) in the beginning of the planning horizon
TD_{jp}	estimated date of arrival of product p in port j (days)
TDA_j	earliest starting time for the delivery of products in port j (days)
TDL_j	superior starting time for the delivery of products in port j (days)
TO_j	loading/unloading time in port j (days)
TC_j	custom clearance time of products in port j (days)
TQ_j	estimated queuing time in port j (days)
TPS_j	earliest starting time for the pickup of a cargo in port j (days)
TPF_j	superior starting time for the pickup of a cargo in port j (days)
TS_{ijv}	sailing time of vessel v between ports i and j (days)
KP_j	draft limit in port j (ton)
MAX_v	capacity of vessel v (ton)
MIN_v	minimum amount of cargo for vessel v (ton)
$PMIN_{pv}$	minimum amount of product p to be loaded in vessel v (ton)
G_v	number of compartments of vessel v
CD_v	demurrage cost of vessel v (\$)
CP_i	fixed utilization cost of using port i (\$)
CV_v	freight price of vessel v (\$)
CUD	daily cost for disrespecting the TWs of a pickup/delivery (\$)
CSN_v	change fee of geographical orientation of loaded vessel v (\$)
VPL_v	maximum number of pickup visits of vessel v
VDL_v	maximum number of delivery visits of vessel v
PVL_j	maximum number of berths in port $j \in N^D$
SN_{ij}	binary matrix that indicates a geographical orientation between ports $i, j \in N^D$
Decision variables	
l_{ijpv}	amount of product p transported using arc (i, j) by vessel v (ton)
eta_{ijv}	estimated arrival time of vessel v at port j from port i (days)
etd_{ijv}	estimated departure time of vessel v from port j for etd_{ijv} (days)
$etud_{ijpv}$	early arrival or delay of vessel v carrying product p using arc (i, j) (days)
x_{ijv}	binary variable that indicates if arc (i, j) is used by vessel v
y_{ijpv}	binary variable that indicates if arc (i, j) is used by vessel v to transport product p
yd_{ijpv}	binary variable that indicates if product p is unloaded of vessel v in port $j \in N^D$, using arc (i, j)
g_{pv}	binary variable that indicates if product p is transported by vessel v
aux_{vw}	binary variable that indicates if vessels v and w are simultaneously using the same port

We model the problem as a multi-commodity arc-flow formulation defined on a directed graph $G = (N, A)$, where set $N = N^V \cup \{s\} \cup \{f\}$ is the set of nodes and $A = \{(i, j) | (i \in N^P, j \in N^V) \wedge (i \in N^D, j \in N^D) \wedge (i \in \{s\}, j \in N^P) \wedge (i \in N^D, j \in \{f\})\}$ is the set of arcs. Dummy nodes s and t represent artificial arrival and departure nodes, respectively. The set of arcs is defined towards avoiding that a delivery port is connected to a pickup port. Each path in the graph from s to f, that respects all TWs, draft, and capacity constraints, corresponds to a feasible vessel route. The objective is to find the minimum cost feasible routes. Based on the above described network and notation, the problem can be formulated as a MILP, as follows:

$$\begin{aligned}\min \sum_{v\in V} CV_v \left(\sum_{i\in N^D} eta_{ifv} + \sum_{j\in N^P} eta_{sjv} \right) + \sum_{i\in N} \sum_{j\in N^V} \sum_{v\in V} CP_j \, x_{ijv} \\ + \sum_{i\in N} \sum_{j\in N^V} \sum_{v\in V} CD_v \, TQ_j \, x_{ijv} + \sum_{i\in N^V} \sum_{j\in N^D} \sum_{p\in P} \sum_{v\in V} CUD \, etud_{ijpv} \\ + \sum_{i\in N^V} \sum_{j\in N^V} \sum_{v\in V} SN_{ij} \, CSN_v \, MAX_v \, x_{ijv}\end{aligned} \tag{1}$$

Objective function (1) minimizes the freight costs, the utilization of ports costs, the demurrage costs, the penalty for disrespecting the TW of a product delivery, and the penalty for changing the sailing geographical orientation of vessels. These costs are computed in the same way planners do in the company.

subject to:

Supply-demand constraints

$$Q_{ip} - \sum_{j\in N^V} \sum_{v\in V} l_{ijpv} + \sum_{h\in N^V} \sum_{v\in V} l_{hipv} \geq 0 \qquad \forall i \in N^V,\, \forall p \in P \tag{2}$$

Constraints (115) ensure that the supply and demand of each product are respected in the pickup and delivery ports, respectively.

Draft constraints

$$\sum_{j\in N} \sum_{p\in P} l_{ijpv} \leq KP_i \qquad \forall i \in N^V,\, \forall v \in V \tag{3}$$

$$\sum_{j\in N} \sum_{p\in P} l_{ijpv} \leq KP_j \qquad \forall j \in N^V,\, \forall v \in V \tag{4}$$

Constraints (127) and (128) assure that a vessel is entering or leaving a port, respecting the port maximum cargo draft.

Ship load constraints

$$Q_{ip} - \sum_{j \in N^V} l_{ijpv} + \sum_{h \in N^P} l_{hipv} \geq 0 \qquad \forall i \in N^P,\ \forall p \in P,\ \forall v \in V \tag{5}$$

$$\sum_{h \in N^V} l_{hipv} - \sum_{j \in N^D} l_{ijpv} \geq 0 \qquad \forall i \in N^D,\ \forall p \in P,\ \forall v \in V \tag{6}$$

$$\sum_{i \in N^P} \sum_{j \in N^D} \sum_{p \in P} l_{ijpv} \leq MAX_v \qquad \forall v \in V \tag{7}$$

$$\sum_{i \in N^P} \sum_{j \in N^D} \sum_{p \in P} l_{ijpv} \geq MIN_v \sum_{j \in N^P} x_{sjv} \qquad \forall v \in V \tag{8}$$

$$l_{ijpv} \geq PMIN_{pv} y_{ijpv} \qquad \forall i \in N^P,\ \forall j \in N^V,\ \forall p \in P,\ \forall v \in V \tag{9}$$

$$\sum_{p \in P} g_{pv} \leq G_v \qquad \forall v \in V \tag{10}$$

Constraints (116) guarantee that a vessel cannot leave a pickup port with a product that is not stocked in its warehouses or to unload any cargo in a pickup port. Likewise, constraints (117) do not allow a vessel to discharge a larger amount of a product in a delivery port than its current loading. Both restrictions avoid cargo transshipment in ports. Constraints (119) ensure that the maximum capacity of each vessel is respected. Constraints (59) assure that vessels are not under utilized. Constraints (60) define the minimum amount of each product to be transported by a vessel. Constraint (121) limit the number of different products to be transported by a vessel to the number of its compartments.

Flow constraints

$$\sum_{j \in N} x_{ijv} \leq 1 \qquad \forall i \in N,\ \forall v \in V \tag{11}$$

$$\sum_{i \in N} x_{ijv} \leq 1 \qquad \forall j \in N,\ \forall v \in V \tag{12}$$

$$\sum_{j \in N} x_{ijv} - \sum_{h \in N} x_{hiv} = 0 \qquad \forall i \in N,\ \forall v \in V \tag{13}$$

$$\sum_{i \in N^P} \sum_{j \in N^D} x_{ijv} \leq 1 \qquad \forall v \in V \tag{14}$$

$$x_{ijv} = 0 \qquad \forall i \in N^D,\ \forall j \in N^P,\ \forall v \in V \tag{15}$$

$$x_{fsv} = 0 \qquad \forall v \in V \tag{16}$$

$$x_{fjv} = 0 \qquad \forall j \in N^V,\ \forall v \in V \tag{17}$$

$$x_{isv} = 0 \qquad \forall j \in N^V,\ \forall v \in V \tag{18}$$

$$\sum_{p \in P} l_{ijpv} - M x_{ijv} \leq 0 \qquad \forall i, j \in N,\ \forall v \in V \tag{19}$$

$$l_{ijpv} - M g_{pv} \leq 0 \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{20}$$

$$l_{ijpv} - M y_{ijpv} \leq 0 \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{21}$$

where M represents a very large number.

Constraints (133) and (134) limit that any arc (i, j) in the network is transversed more than once by the same vessel. Constraints (135) are flow conservation constraints. Constraints (136) establish a connection between pickup and delivery ports, while constraints (137) prevent the existence of arcs connecting delivery to collection ports. Constraints (138), (139) e (140) delimit the possibilities of arcs with the dummy ports s and t. Constraints (141) and (20) define that the flow of an amount of product p in arc (i, j), being transported by vessel v, only occurs if it transverses arc (i, j), carrying product p in one of its compartment. Constraints (21) link the continuous variable l_{ijpv} to the its binary counterpart, y_{ijpv}.

Time Constraints

$$eta_{ijv} \geq TPS_j x_{ijv} \qquad \forall i \in N,\ \forall j \in N^P,\ \forall v \in V \tag{22}$$

$$etd_{ijv} \leq TPF_j x_{ijv} \qquad \forall i \in N,\ \forall j \in N^P,\ \forall v \in V \tag{23}$$

$$etd_{ijv} \geq eta_{ijv} + TQ_j + TO_j + M(x_{ijv} - 1) \qquad \forall i \in N,\ \forall j \in N^V,\ \forall v \in V \tag{24}$$

$$eta_{ijv} \geq etd_{hiv} + TS_{ijv} + M(x_{ijv} - 1) \qquad \forall h \in N,\ \forall i \in N^V,\ \forall j \in N^V,\ \forall v \in V \tag{25}$$

$$etud_{ijpv} \geq \mid TD_{jp} - TC_j - etd_{ijv} \mid + M(yd_{ijpv} - 1) \qquad \forall i \in N^V,\ \forall j \in N^D,\ \forall p \in P,\ \forall v \in V \tag{26}$$

$$eta_{ifv} \geq etd_{hiv} + M(x_{ifv} - 1) \qquad \forall i, h \in N^V,\ \forall p \in P,\ \forall v \in V \tag{27}$$

$$eta_{siv} \leq eta_{ijv} - TS_{ij} - TO_i - TQ_i - M(x_{ijv} - 1) \qquad \forall i, j \in N^V,\ \forall p \in P,\ \forall v \in V \tag{28}$$

$$eta_{ijv} + TQ_j - M(x_{ijv} - 1) + Maux_{vw} \geq etd_{kjw} + M(x_{kjw} - 1) \qquad \forall i, k \in N,\ \forall j \in N^V,\ \forall v, w \in V, v \neq w \tag{29}$$

$$eta_{kjw} + TQ_j - M(x_{kjw} - 1) + M(1 - aux_{vw}) \geq etd_{ijv} + M(x_{ijv} - 1) \qquad \forall i, k \in N,\ \forall j \in N^V,\ \forall v, w \in V, v \neq w \tag{30}$$

$$etd_{ijv} \leq TD_{jp} + TDL_j - TC_j + M(-yd_{ijpv} + 1) \qquad \forall i \in N^V,\ \forall j \in N^D,\ \forall p \in P,\ \forall v \in V \tag{31}$$

$$etd_{ijv} \geq TD_{jp} - TDA_j - TC_j - M(yd_{ijpv} - 1) \qquad \forall i \in N^V,\ \forall j \in N^D,\ \forall p \in P,\ \forall v \in V \tag{32}$$

Constraints (144) and (145) are related to the time windows in the collection of products, which affect the arrival and departure times, respectively, of each vessel in/from each port.

Constraints (145)– (153) follow Miller-Tucker-Zemlin inequalities for sub-tour elimination. Constraints (146) define that the departure time of each vessel from a port depends on its arrival time, estimated queuing time, and load/unloading time in the port. Constraints (147) determine the arrival time in port j of each vessel as related to its departure time from port i plus the sailing time TS_{ijv} by vessel v. Soft constraints (148) account for the specified TWs of the delivery of a product in a port. They are only considered if an early arrival or a delayed departure of a vessel in a port for each product being transported happens. Constraints (149) and (150) define the initial and ending route times of each vessel. Constraints (151) and (30) ensure that two vessels can not simultaneously use the same port. Constraints (152) and (153), similarly to (144) and (145), establish the departure TW of each vessel from a port, considering the estimated arrival TD_{jp} of product p in port j.

Routing Constraints

$$\sum_{i \in N^P} \sum_{j \in N^P} x_{ijv} \leq VPL_v - 1 \qquad \forall v \in V \tag{33}$$

$$\sum_{i \in N^D} \sum_{j \in N^D} x_{ijv} \leq VDL_v - 1 \qquad \forall v \in V \tag{34}$$

$$\sum_{i \in N} \sum_{v \in V} x_{ijv} \leq PVL_j \qquad \forall j \in N^V \tag{35}$$

$$y_{ijpv} - x_{ijv} \leq 0 \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{36}$$

$$y_{ijpv} - g_{pv} \leq 0 \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{37}$$

$$etd_{ijv} - Mx_{ijv} \leq 0 \qquad \forall i, j \in N,\ \forall v \in V \tag{38}$$

$$eta_{ijv} - Mx_{ijv} \leq 0 \qquad \forall i, j \in N,\ \forall v \in V \tag{39}$$

$$y_{ijpv} - TD_{jp} yd_{ijpv} \leq 0 \qquad \forall i \in N^V,\ \forall j \in N^D,\ \forall p \in P,\ \forall v \in V \tag{40}$$

Constraints (33) and (34) limit the number of pickup and delivery ports, respectively, that each vessel can dock in its route. Constraints (35) limit the number of vessels that can dock at each port during the planning horizon. Constraints (129) and (124) stipulate that each vessel can only transport a product using arc (i, j) if: (i) the vessel uses arc (i, j) in its route; and (ii) the vessel is carrying the product in one of its compartments. Constraints (130) and (131) stipulate that each vessel has an arrival time and a departure time, respectively, in/from a port j coming from port i, if it uses arc (i, j) in its route. Finally, constraints (132) guarantee that a product is only unload from a vessel in a port if: (i) there is a demand for the product in this port; and (ii) the vessel is carrying this product in its route.

Domain of the variables

$$l_{ijpv} \geq 0 \qquad \forall i, j \in N,\ \forall p \in P,\ \forall g \in V \tag{41}$$

$$etd_{ijv} \geq 0 \qquad \forall i, j \in N,\ \forall v \in V \tag{42}$$

$$eta_{ijv} \geq 0 \qquad \forall i, j \in N,\ \forall v \in V \tag{43}$$

$$etud_{ijpv} \geq 0 \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{44}$$

$$x_{ijv} \in \{0, 1\} \qquad \forall i, j \in N,\ \forall v \in V \tag{45}$$

$$y_{ijpv} \in \{0, 1\} \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{46}$$

$$yd_{ijpv} \in \{0, 1\} \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{47}$$

$$g_{pv} \in \{0, 1\} \qquad \forall p \in P,\ \forall v \in V \tag{48}$$

$$aux_{vw} \in \{0, 1\} \qquad \forall v, w_{v \neq w} \in V \tag{49}$$

It should be noticed that variables aux_{vw} and constraints (29), (30), (33), (34), and (35) are very specific to our problem. If we disregard them, the problem becomes a general multi-product, split load, heterogeneous fleet with dedicated compartment cargo routing and scheduling problem with time windows and draft limits, that can be applied to different real-world problems that involve the deep-sea transportation of bulk products. Further, the above mentioned variables and constraints add reasonable complexity towards obtaining a good quality solution with efficiency in comparison with the more generic problem. Given the extensive number of constraints and binary variables in model (115) – (49), commercial MILP packages may have difficulties of solving large instances of the problem. Since our objective is to solve real-world instances, involving large number of products and ports, we develop a heuristic approach fully described in the next section.

2.5 SOLUTION METHOD

To overcome the computational difficulties associated with producing a good solution for very large instances, we develop a modified a RaF based decomposition matheuristic to solve the problem. The solution approach can be characterized as a matheuristic, since it integrates heuristics and MIP strategies and software. RaF is a solution strategy employed in many production planning problems, such as the lot-sizing problem (MOHAMMADI et al., 2010) and scheduling (KELLY; MANN, 2004). More recently, RaF has been successfully applied to solve MIRPs Uggen et al. (2013), Friske e Buriol (2018) and a maritime cargo routing and scheduling problem Rodrigues et al. (2016). RaF is based on an intuitive idea that a planning problem can be decomposed into n time intervals or sub-problems, that are solved sequentially. In the first iteration, the sub-problem is solved with the integer variables corresponding to the first interval, while the remaining variables remain continuous. In subsequent iterations, the previously integer variables are fixed with the values of a solution obtained in the previous step. A new interval receives the integrality condition and the remaining variables remain continuous and unfixed at each new iteration. This process is repeated until n iterations are executed when a complete solution to the original problem is found. (POCHET; WOLSEY, 2006) present a comprehensive description of this solution method.

The developed matheuristic is basically a three-step approach. In the first step, a simple and fast relaxation algorithm is applied to find partial feasible solutions, since some constraints are relaxed on purpose. Next, a modified RaF algorithm is employed towards finding good feasible solutions to the problem. Finally, an improvement phase, based on powerful MILP solver developed routines, is executed in attempt to decrease the final gap of the solution provided by the modified RaF algorithm. Algorithm 8 outlines the matheuristic framework.

Algorithm 1: Matheuristic approach

1. Apply the relaxation heuristic described in Algorithm 2.

2. Apply Algorithm 3.

3. Apply the improvement procedure described Algorithm 4.

2.5.1 Relaxation algorithm

The main objective of the relaxation phase is to find a solution that may be infeasible for the complete model, but can accelerate the process of finding a good solution in the subsequent step. Due to the complexity of the model, it is pretty hard to find a feasible solution to the problem within a reasonable time. Further, during experimentation, we note that the CPU time used in finding a feasible initial solution was not being compensated either in quality or efficiency in the further step, considering the whole solution process.

Based on these observations, we decided to apply a simple and fast relaxation heuristic able to find very quickly a relaxed initial solution to improve the convergence process of the next step of Algorithm 8. The following requisites were either neglected or relaxed: (i) The maximum number of ports visited by a vessel in the delivery was not taken into consideration; (ii) TWs were implicitly considered, but not enforced. Priority was given to pickup ports in which the average TWs is smaller; (iii) Vessels can leave pickup ports without considering the draft limit; (iv) The number of simultaneous vessels in a port is not considered; and (v) The geographical orientation of vessels is neglected.

Before describing the algorithm, it is necessary to introduce some additional notation. Let O be the set of vessels in non-descending order of their freight costs, cl_v be the current load of vessel v, np_v be the number of visited pickup ports by vessel v, vis_{iv} be a binary variable that define if port i was visited by vessel v, and u_v be the last port visited by vessel v. Algorithm 2 presents the pseudocode of the relaxation heuristic.

The algorithm is a process approach routine that follows vessels through two interconnected time-ordered sequences, cargo pickup and delivery. First, the algorithm handles the load of all demanded products in the origin ports, using the required number of vessels to respect the

Algorithm 2: Relaxation heuristic

1. Pickup initialization. Compute $\alpha_k = \frac{TPS_k + TPF_k}{2}, k \in N^P$. Set $cl_v, u_v \leftarrow 0, \forall v \in V$, $vis_{ip} \leftarrow 0, \forall i \in N^P, \forall v \in V \leftarrow 0, g_{pv} \leftarrow 0, \forall p \in P, \forall v \in V, Q^{'}_{ip} = Q_{ip}, l_{ijpv} \leftarrow 0, \forall i, j \in N, \forall p \in P, \forall v \in V$.

2. Pickup. For all $v \in O$ do:

2.1. If $((cl_v < Max_v) \wedge (\sum_p g_{vp} < G_v) \wedge (np_v < VPL_v))$ then select port $k^* \leftarrow k \in N^P \alpha_k | (\sum_p Q^{'}_{kp} > 0) \wedge (cl_v < KP_k) \wedge (vis_{kv} = 0))$. Otherwise, go to Step 2.

2.2. While $((cl_v < Max_v) \wedge (\sum_p g_{vp} < G_v)$ do

2.2.1. Select product $p^* \leftarrow pQ^{'}_{k^*p}$. Load vessel v with amount $load_{p^*} \leftarrow min[Q^{'}_{k^*p^*}, cl_v]$.

2.2.2. Update $cl_v \leftarrow cl_v + load_{p^*}$, $Q^{'}_{k^*p^*} \leftarrow Q^{'}_{k^*p^*} - load_{p^*}$, $g_{p^*v} \leftarrow 1$, and $l_{u_v k^* p^* v} \leftarrow l_{u_v k^* p^* v} + load_{p^*}$.

2.2.3. Set $x_{u_v k^* v} \leftarrow 1$, and $y_{u_v k^* p^* v} \leftarrow 1$.

2.3. Set $eta_{u_v k^* v} \leftarrow etd_{u_v k^* v} + TS_{u_v k^* v}$, $etd_{u_v k^* v} \leftarrow eta_{u_v k^* v} + TO_{k^*} + TC_{k^*}$, $vis_{k^* v} \leftarrow 1$, $np_v \leftarrow np_v + 1$, and $u_v \leftarrow k^*$. Go to Step 2.1.

3. Delivery. Set $np_v \leftarrow 0$. For all $v \in O$ do:

3.1. If $((cl_v > 0) \wedge (np_v < VDL_v))$ then select port $k^* \leftarrow k \in N^D[\sum_{j \in N^D} \sum_{p \in P} l_{u_v jpv}, -\sum_p Q^{'}_{kp}] \mid ((cl_v < KP_k) \wedge (cl_v < KP_k) \wedge (vis_{kv} = 0))$. Otherwise, go to Step 3.

3.2. For all $p \in P | ((Q^{'}_{k^*p} > 0) \wedge (g_{pv} = 1))$ do

3.2.1. Unload amount $uload_p \leftarrow \min[Q_{k^*p}, l_{u_v k^* pv}]$ from vessel v. Update $Q_{k^*p'}$, $l_{u_v k^* pv}$, and cl_v with amount $uload_p$, accordingly. Set $x_{u_v k^* v} \leftarrow 1$, $y_{u_v k^* p^* v} \leftarrow 1$, and $yd_{u_v k^* p^* v} \leftarrow 1$.

3.2.2. If $(l_{u_v k^* pv} = 0)$ then $g_{pv} \leftarrow 0$.

3.3. Set $eta_{u_v k^* v} \leftarrow etd_{u_v k^* v} + TS_{u_v k^* v}$, $etd_{u_v k^* v} \leftarrow eta_{u_v k^* v} + TO_{k^*} + TC_{k^*}$, $u_v \leftarrow k^*$, $np_v \leftarrow np_v + 1$, and $vis_{k^* v} \leftarrow 1$. Go to Step 3.1.

4. Output. Return obtained solution S.

considered constraints (Step 2). Each vessel visits a set of pickup ports with available products and enough draft, giving preference to the ones with more strict TWs. The algorithm attempts to load the maximum amount of each stored product in each port, respecting the capacity and the number of compartments in the vessel. Products are prioritized based on the current amount stocked in the port. All variables of the problem, with the exception of $etud_{ijpv}$, and aux_{vw} are updated accordingly with the cargo load in the vessel. This process is repeated until the vessel is incapable of loading any extra cargo or all products are collected. The whole process is then repeated with a new vessel, until all products are collected. After all vessels are loaded, the algorithm starts the delivery process by each vessel (Step 3), following the same order of the pickup stage. The delivery process is simpler. Each vessel unload the maximum amount transported of all products in each port, respecting the demand for each product. The ports to be visited are ordered by their maximum demand of all products. The process of delivery finishes when all vessels are empty. The same variables are updated accordingly with the cargo unload. A vessel only visits a port, during both the pickup and delivery processes, if it was not previously visited in its route.

2.5.2 Modified relax-and-fix algorithm

The developed algorithm is a modified RaF strategy. A key aspect in designing a RaF algorithm is to select a suitable strategy to partition the integer variables set into sub-sets. In a previous applications on maritime PDPTW problems, the variables were grouped by forward time-based division of the problem (RODRIGUES et al., 2016). Unfortunately, some experimentation has demonstrated that this traditional strategy was not appropriate for our formulation. In opposite to the problem solved by Rodrigues et al. (2016), we consider multiple products and split load/unload in visiting ports, considerably increasing the number of continuous and binary variables. Our formulation also involves a much larger number of time variables related with vessel routes in comparison with Rodrigues et al. (2016), making it difficult to find suitable time intervals employing the usual RaF strategy. Further, these variables were considerable interconnected with other binary variables, making the problem very difficult to solve and decompose. Several intervals were tested and either very bad integer solutions or infeasible ones were obtained for tested instances.

Our developed algorithm presents a different strategy in which the set of integer variables are partitioned into non-divisible blocks. Each block is defined by a strategy set vector (SB) that can be seen as a permutation of the integer variables set. In the first iteration, only the variables in the first block are considered as integer, while all the remaining ones are relaxed in the model. Then, the current sub-model is solved for the entire horizon planning. As it is difficult to find a feasible solution to the current MILP being solved due to the excessive number of constraints, the repairing algorithm developed by Fischetti e Lodi (2008) is employed in the solution process. The repairing procedure is a hybrid algorithm that uses the feasibility pump method to provide initial solutions to the local branching. More specifically, the original current MILP being solved is augmented with artificial variables. The augmented model is then solved iteratively to reduce the infeasibility by driving the values of the artificial variables to zero. The objective is not only to repair solutions, but also to repair infeasible MILP models. At the next iterations of the algorithm, the subsequent variables in a block are defined as integer (but not fixed) and incorporated in the current sub-model, obeying the position of the variable in the block, defining our strategy as a relax-and-define (RaD) algorithm. The solutions obtained in previous iterations are used as initial ones for the current sub-model. In the last iteration, all variables are considered as integer in the current sub-model, and solved to find the best possible solution. The number of iterations is equal to the cardinality of the integer variables set (IS). As a consequence, our RaD strategy solves a smaller number of larger sub-models rather than a larger number of smaller sub-models used by the traditional RaF. Further, the values of integer variables are not fixed in the interactions, but included in a MILP solution pool.

Before presenting the algorithm, it is necessary to introduce additional notation. Let SB_k be the strategy vector k, consisting of indexes to each integer variables in IS, B_s be the set of fixed integer variables at iteration s, ordered by a FIFO strategy. The total number of iterations is denoted by $|IS|$. Let SP be the a MIP solution pool, t_s be the execution time limit of iteration s, h_i be an index to the variable in i-th position of vector SB_k, and S be a vector of MIP solutions. Algorithm 3 outlines the developed routine.

Algorithm 3: Relax-and-define algorithm

1. $B_s \leftarrow \emptyset$

2. $SP \leftarrow$ relaxation heuristic solution

3. For $s = 1, \ldots, |IS|$

- **3.1.** $h_s \leftarrow$ index to the variable in the s-th position of vector SB_k
- **3.2.** $B_s \leftarrow B_{s-1} \cup \{h_s\}$
- **3.3.** Set current MILP as model (1) – (45) with the integral constraints related to variables in B_s
- **3.4.** Solve current MILP using the repair algorithm by Fischetti e Lodi (2008) with initial solutions in SP within time limit t_s
- **3.5.** $SP \leftarrow$ feasible solutions of the current MILP

4. Return best solution in SP

The algorithm has some customizable parameters, which according to the instance of the problem, can be altered so that the algorithm might improve the performance. Among these parameters, we highlight the relaxation strategies and the size of parameters t_s. Particularly for the latter, it should be observed that the heuristic priority at each iteration is to obtain several feasible solutions for the current model being solved rather than intermediary optimal ones. The majority of the solutions, if not all, will be infeasible for the complete model. In general, as more solutions in P_s, more effective is the repair routine in offering more promising intermediary solutions towards good final ones in P_{s+1}, considering a threshold in the memory usage. In order to deal with the usual quality-efficiency trade-off, we use t_s as a control parameter. Depending on the ability of the MILP solver to obtain good solutions, t_s can be increased or decreased. The parametrization is discussed with more details in Section 2.6.

Table 1 presents the definition strategies considered to solve our problem, e.g, the different orders in which the decision variables are defined as integers in each iteration of the solution process. The number of strategies is defined by the permutation of the integer variables in set vector SB. This process requires some experimentation, but it is facilitated by the resources of current MILP solvers in terms of quickly defining infeasible solutions and difficult constraints to be solved, among others. Further, a pruning process can be performed by an analysis of the MILP formulation. Particularly, we used the connection among the variables to determine their position in vector SB. All strategies considered to solve our problem uses variables x_{ijv} as the first variable to be fixed as integer, since the flow in an arc defines the values of almost all remaining integer and continuous variables in our model. Note that variable yd_{ijpv} is always positioned after its dominant variable y_{ijpv}. Strategy A uses variable g_{pv} as the next one to become integer, since both variables define the values of all remaining ones in the formulation. Strategies B and C are similar, but altering the position of variable aux_{vw} in the vector. Strategy D and E are variations of strategy A, altering the position of variable g_{pv}.

Tabela 1 – Definition block strategies

$Strategy$	Strategy vector (SB_k)
A	$[x_{ijv}, g_{pv}, y_{ijpv}, yd_{ijpv}, aux_{vw}]$
B	$[x_{ijv}, g_{pv}, aux_{vw}, y_{ijpv}, yd_{ijpv}]$
C	$[x_{ijv}, aux_{vw}, g_{pv}, y_{ijpv}, yd_{ijpv}]$
D	$[x_{ijv}, y_{ijpv}, g_{pv}, yd_{ijpv}, aux_{vw}]$
E	$[x_{ijv}, y_{ijpv}, yd_{ijpv}, g_{pv}, aux_{vw}]$

Figure 2 illustrates the RaD workflow using strategy A. At iteration 1 ($s = 1$), only the variables in the first block (x_{ijv}) were defined as integer, while the remaining ones are relaxed. When $s = 2$, the first block of variables are set as integer for all subsequent iterations, and variables g_{pv} are defined as integer. When a variable is defined as integer at any iteration, this condition is maintained for all remaining iterations. This process is repeated until $s = 5$, when the last block of variables (aux_{vw}) receive the integrality constraints, restoring the original complete model. It is expected that the matheuristic provides a feasible solution to the problem. Each iteration s is executed within a time limit t_s.

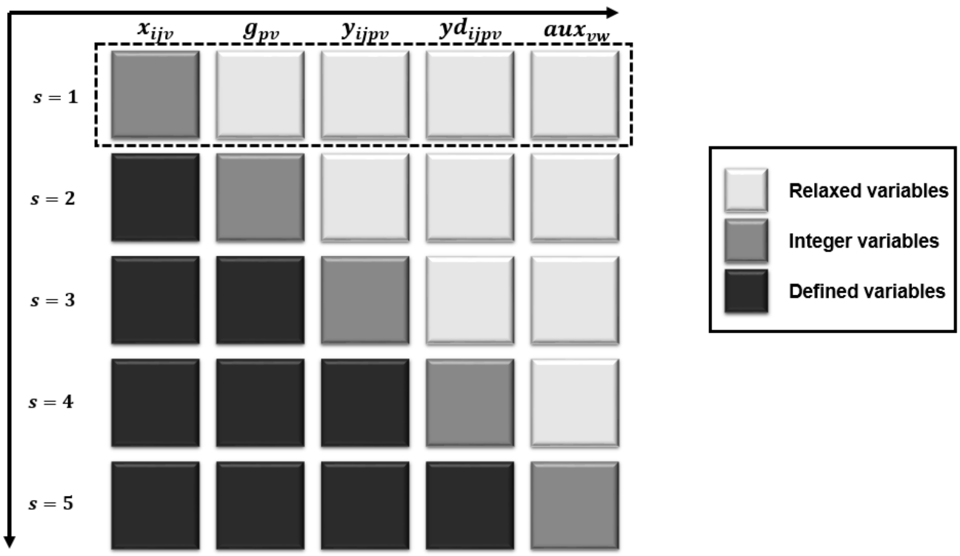


Figura 2 – RaD phase diagram, strategy A

2.5.3 Improvement phase

After the RaD step is finalized, a post-processing phase is applied in order to improve the best solution found by the algorithm. The improvement phase is composed of a two-step procedure. First, the improvement phase polishes the solution from the RaD algorithm, solving the whole problem (model (1)–(49)) using the polishing algorithm by Rothberg (2007). Next, the same problem is re-solved employing the repair algorithm by Fischetti e Lodi (2008). The solution pool generated in the polishing step is used as input of the repair heuristics. It is important to notice that if the polishing step is not able to find any new solution, the polishing heuristic is not activated and the matheuristic finishes its processing.

The polishing algorithm is based on Rothberg (2007), an evolutionary approach in which the usual operations of crossover and mutation are built within a large-neighborhood search framework. The solutions of the polishing algorithm are then integrated into the MILP search tree, and the solutions found by the MILP solver are used in the evolutionary algorithms, characterizing a benefiting integration of information during the search solution process.

Considering p as the best solution from the RaD step, the improvement phase can be outlined by Algorithm 4.

Algorithm 4: Improvement phase algorithm

1. $SP \leftarrow p$

2. Set MILP as model (1)–(49)

3. Solve the MILP using the polish routine by Rothberg (2007) with SP as initial solution within time limit t_i

4. $SP \leftarrow SP \cup$ MILP solutions obtained in polishing

5. If $|SP| \geq 2$

  5.1. Solve the MILP using the repair algorithm by Fischetti e Lodi (2008) with initial solutions in SP within time limit t_i

6. Return best solution in SP

Based on experimentation, we observed that the polishing routine can have a significant effect in improving integer solutions, but sometimes requires a considerable amount of time to fulfill this task. We decided to limit the execution of each step of the improvement phase to a limit time of $t_i = max_{s=1,\ldots,|IS|} t_s$, in order to find a good compromise between quality and efficiency.

2.5.4 Implementation

The matheuristic was implemented in C++, using IBM ILOG CPLEX 12.8.0 to solve the MILP models. Also, it uses several inherent capabilities of contemporary MILP solvers, such as parallel branch-and-cut, multiple default heuristics, non-traditional tree-of-trees search, solution improvement, symmetry detection, and cutting planes. CPLEX parameter names are used to specifically identify the capabilities utilized. Although the denominations are specific, similar or analogous parameters are available in other powerful MILP solvers, such as Gurobi or SAS.

Tabela 2 – CPLEX parameters controlling cuts

Parameter	Value	Meaning
`Cliques`	2	Generate clique cuts moderately
`Covers`	2	Generate cover cuts moderately
`DisjCuts`	2	Generate disjunctive cuts moderately
`FlowCovers`	1	Generate flow cover cuts moderately
`FlowPaths`	1	Generate flow path cuts moderately
`GUBCovers`	2	Generate generalized upper bound (GUB) cuts moderately
`ImplBd`	1	Generate implied bound cuts moderately
`MIRCuts`	1	Generate a moderate number of MIR cuts (mixed integer rounding cuts)
`MCFCuts`	1	Generate a moderate number of MCF cuts (multi-commodity flow cuts)
`ZeroHalfCuts`	1	Generate zero-half cuts moderately

Parameters `Repair.Tries` and `SubMIPNodeLim` were employed in the RaD algorithm to attempt to repair infeasible MIP solutions in SP. The former modifies the number of attempts by the repair heuristic, while the latter modifies the depth of the repair. Both parameters are stepped up, with the former being changed from *default* 1 to 1000, and the latter from 500 to 10000, respectively. Parameters `PolishAfterIntSol` and `PolishTime` were used in the improvement phase. In addition, parameter `Probe` was activated at its maximum level of 3 in each step of the algorithm. *Probing* is a technique that analyzes the logical implications of the problem by fixing each binary variable to 0 or 1, thus increasing the probability of finding a viable solution for the subproblem in question. Parameter `Emphasis_MIP` was also used to modify the emphasis of the optimization engine. CPLEX default is 2, so the optimizer processes an "equilibrium" between optimality and feasibility. In the matheuristic, this value is changed to 1, which prioritizes the viability of the solution. Since the matheuristic has limited ability to see how an initial decision affects the subsequent intervals, it may not be advantageous to solve each subproblem at optimality (UGGEN et al., 2013).

Given the success of using cutting planes for solving the TSP-DL Rakke et al. (2012), Battarra et al. (2014), several CPLEX parameters related with *cutting planes* were activated with different values than the defaults, as presented in Table 2. Although aggressive values would increase the contribution of the generated cuts in the process of finding a good solution, moderate values offer better compromise between efficiency and solution quality.

2.6 EXPERIMENTS

In this section, we describe the experiments carried out using real cases provided by the chemical company. Intentionally, we used instances with a wide range in the problem dimensions (ports and products), representing the seasonality involved in the fertilizer business (see Table 5). Experiments were carried out using real-life planning cases to evaluate the performance of the optimization approach. The used instances are real plans executed by the company from 2013–2019. Computations were performed using an AMD Ryzen 5 2600 six-core computer with 3.4Ghz and 16 GB RAM. Our objective was demonstrate to company managers that planning can be accomplished using easily available and cost-effective computational resources.

2.6.1 Matheuristic parameters tuning

Before evaluating the matheuristic, we carried out experiments to select the best set of parameters. For economy's sake, we only present results related to some of the most important parameters. Table 3 presents a comparison of the solution quality in five runs of each strategy vector proposed in Table 3 for instance 5 (a middle size one in Table 5). Each run uses random values of $\forall s,\ t_s \in [150, 200]$. Average and standard deviation of the objective functions for each strategy are also presented. The best values found for each run are presented in bold. Strategies A or C obtained the best values for all tested instances. These are not surprising results, given the similarity between the two strategies and the natural dependency of all remaining binary variables with x_{ijv}, and g_{pv}. Although strategy A has not always obtained the best value, it presented the most overall robust behavior, with the best compromise between average values and standard deviation. Figure 3 illustrates the solution method process of each strategy on the same instance 5 for five iterations of the RaD algorithm, and two improvement steps. As expected, the solution process is quite disturbed by the introduction of relevant new integer variables for each strategy in the sub-problems. Observe that the post-processing phase was not relevant for almost all strategies for this instance. In general, we noticed that this stage is more relevant for large size instances. Based on the obtained results, we selected strategy A as the most appropriate choice, and limit further experimentation to this strategy vector.

Tabela 3 – Results for instance 5 for different strategy vectors

	Objective Function						
SB_k	Run 1	Run 2	Run 3	Run 4	Run 5	Average	SD
A	2182262	2089586	2183361	2147773	2105837	2141764	43069
B	2204890	2239481	2232796	2144956	2282726	2220970	50826
C	2254921	2168180	2123997	2123997	2233216	2180862	60938
D	2182262	2105837	2315293	2183671	2147773	2186967	78452
E	2238943	2222102	2286424	2213727	2182684	2228776	38151

Another very important parameter in the matheuristic is the size of t_s. A small value can avoid the matheuristic to find a feasible solution, while a very large value can guarantee a very good solution, but unnecessarily increasing the computational time. In order to properly tune this parameter, we tested several values in four different size instances: 4 (small), 5 (medium), 6 (large), and 14 (extra-large). Table 4 presents the best solutions for different values of parameter t_s for the four instances. The total computational time in seconds (CPU) is expressed as $7t_s$.

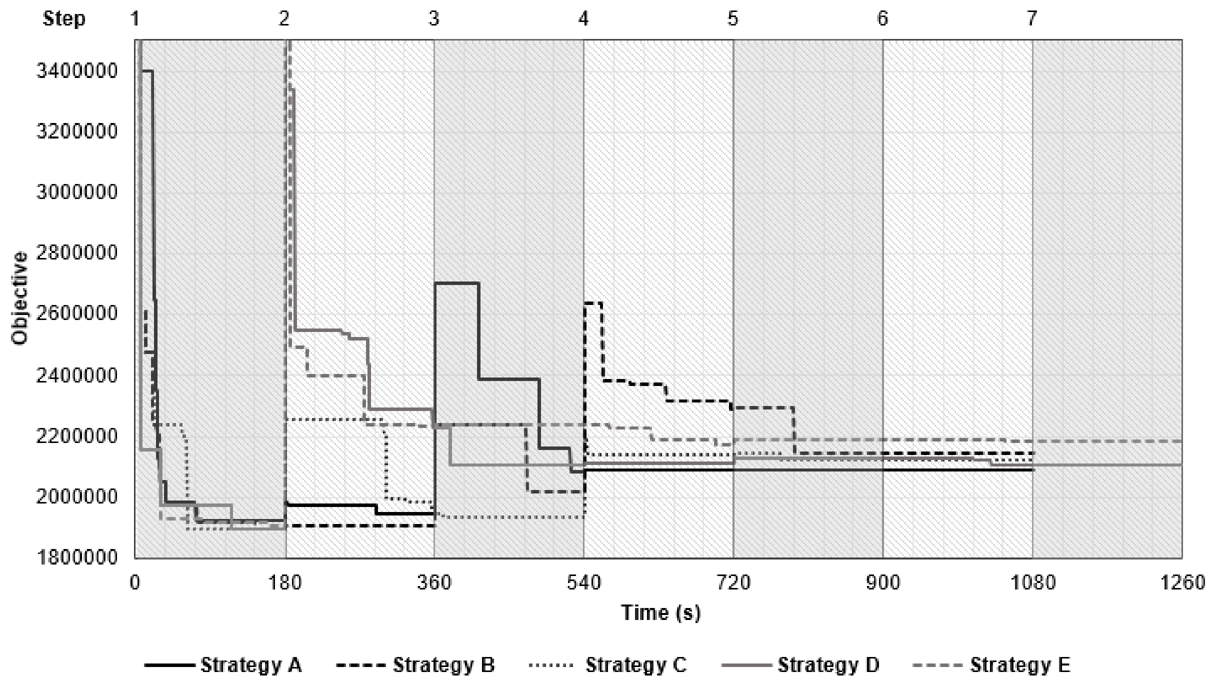


Figura 3 – Definition strategy behavior by step of the RaD algorithm, instance 5

Tabela 4 – Impact of parameter t_s on different instance types

Instance	Instance Type	t_s(s)	CPU(s)	Solution
4	Small	10	70	1216330
		20	140	1206560
		30	210	1206560
		40	280	1206560
5	Medium	120	840	2144956
		180	1260	2123997
		240	1680	2118370
		300	2100	2105837
6	Large	200	1400	3651044
		300	2100	3587072
		400	2800	3578946
		500	3500	3578946
14	Extra-large	400	2800	–
		500	3500	4771420
		600	4200	2568760
		700	4900	2423750

The results in Table 4 ratify the expected behavior. Larger values of t_s led to better solutions. However, there is a clear upper threshold in the value of t_s for each instance. Beyond this value, the marginal contribution in the objective function is small, considering the additional CPU time. Although the choice of an appropriate value is highly dependent on the instance being solved, the obtained results were helpful in selecting useful initial balanced values of parameter t_s, towards good compromising solution in terms of efficiency and efficacy. As small instances are relatively easy to solve, it is possible to set $t_s \leq 20s$. For medium instances, a balanced t_s can be set to above 180s, with a relative deviation of less than 0.9% in comparison with the best solution with $t_s = 300s$. For large instances, a balanced value of 300s seems quite sound. Finally, for extra-large instances, a minimum value of 600s is recommended, but depending of the instance it would be necessary to increase this value, given the difficulty of solving the intermediary complex sub-problems during the execution of the RaD algorithm. Nonetheless, it should be noted that the above values are reference values, requiring validation by experimentation.

2.6.2 Real Instances

Table 5 presents a comparison of the solution obtained from the company, the MILP solver with default parameters (CPLEX), the MILP solver with the parameters defined in the Table 2 (P-CPLEX), and the matheuristic. Each instance is defined in terms of the number of origin ports (OP)/delivery ports (DP)/products to be transported (P). In the table, we compare the quality and efficiency of the obtained solutions, in terms of final costs (Solution), CPU time in seconds to obtain the presented solution (CPU), and the number of vessels (V). The number and characteristics of the vessels, and the costs computation follow the data made available by the company. Further, a vessel should visit, at most, two pickups and three deliveries ports, following an operational policy of the company.

The company solution was the planning costs computed by the logistics department. However, the managers were not only interested in minimizing costs, but also in delivering good customer services. For some historic cases, the cargo delivery was considered as the most important criteria to be met, causing significant increase in the final cost. Due to commercial confidentiality, the company did not disclosure the information of when this fact happened. As a consequence, the comparison between the company's solutions and the optimization methods must be viewed very carefully.

We limited the execution of CPLEX to six hours for instances 1–23. For instance 24, a time limit of twelve hours was considered. The matheuristic was executed with strategy A and values of t_s following the recommendations from the previous section in accordance with the instance size. Note that in the CPLEX and P-CPLEX cases, "–" indicates that they fail to produce any feasible solution within the time limit considered.

The optimization approach offered solutions with smaller total costs than the logistics department of the company. CPLEX and P-CPLEX were not able to find solution for five instances (12, 18, 19, 20, and 23) and four instances (18, 19, 22, and 23), respectively, within 2 hours. Moreover, P-CPLEX was not able to solve instance 24 within 12 hours. The matheuristic proved to be the most robust solution method, being capable of offering average better solutions for all instances in less than 6300s. As a consequence, we focus our qualitative comparison in the matheuristic. The reduction in the total costs is dependent on the instance. For instances 3 and 8,

Tabela 5 – Company-optimization methods comparison

		Company		CPLEX			P-CPLEX			Matheuristic		
Inst	OP-DP-P	Solution	V	Solution	CPU	V	Solution	CPU	V	Solution	CPU	V
1	2-4-7	2577470	2	1206770	281	2	1206770	454	2	1206770	70	2
2	2-5-5	1513310	4	1035600	848	4	1035600	932	4	1035600	70	4
3	2-5-6	2086000	3	827910	84	2	827910	100	2	827910	70	2
4	2-5-6	3244746	3	1575446	11132	3	1575446	10848	3	1575446	1260	3
5	3-3-8	2619014	3	1279264	110	3	1279264	217	3	1279264	70	3
6	3-5-6	1075550	3	998740	138	2	998740	144	2	998740	70	2
7	3-5-7	1701770	3	1206560	2490	3	1206560	2212	3	1206560	140	3
8	4-5-6	5311625	3	2177327	21600	3	2131213	21600	3	2123997	1260	3
9	5-7-6	5343392	4	4046932	21600	3	4046932	21600	3	3578946	2100	3
10	4-4-6	5756646	4	2225366	21600	4	2215006	21600	4	2215006	2100	4
11	4-4-6	2592598	4	1618622	21600	4	1618622	21600	4	1681913	2100	4
12	3-6-8	3657060	4	-	21600	-	2344300	21600	4	2391940	2100	4
13	5-5-7	3780400	4	3120770	21600	4	2150800	21600	4	2153570	2100	4
14	5-5-8	6470280	4	5703900	21600	4	4102690	21600	4	3710430	2100	4
15	5-5-7	5109720	5	3250960	21600	4	3786290	21600	4	2732850	2100	4
16	5-7-5	3389492	4	2361882	21600	4	2347459	21600	4	2299431	2100	4
17	5-7-9	4088320	4	2749840	21600	3	2135550	21600	3	1964790	2100	3
18	5-8-10	5250760	4	-	21600	-	-	21600	-	2439760	2100	3
19	6-9-10	4470320	4	-	21600	-	-	21600	-	2568760	4200	4
20	6-5-9	5297050	4	-	21600	-	3449212	21600	4	3314918	2100	4
21	4-5-8	3068690	4	1783480	21600	4	1726020	21600	4	1688320	1260	4
22	6-7-11	8848112	6	7201920	21600	6	-	21600	-	4661604	2100	6
23	7-7-10	9090452	7	-	21600	-	-	21600	-	5501930	2100	7
24	6-8-10	9655370	7	6704940	43200	7	-	43200	-	3810510	6300	7

the reduction was quite high, while for instance 6 the reduction was around 7%. In six instances, the matheuristic reduced the number of vessels. The decrease in total costs might be explained by several reasons, as follows: (i) better distribution of cargo/routing; (ii) use of fewer vessels; (iii) reduced number of inversions of the geographical orientation of vessels' route; or (iv) the company's decision to prioritize customer service rather costs. Although we cannot objectively measure the matheuristic impact in terms of improving the solution due to different objectives prioritization for some instances, the matheuristic was more efficient than the company's current method. The company requires several days to obtain a feasible solution for each plan using EXCEL, while the optimization approach can find better solutions in minutes, for small and medium instances, and in hours, for large and extra-large ones.

If we only compare the optimization approaches, we can observe that there was no significant differences in the final solution for the small to medium instances. The differences began to appear to the medium instances onwards. Solutions from CPLEX were dominated by the remaining methods. In general, the use of cutting planes in P-CPLEX allowed better solutions than CPLEX. Instance 13 is a good example. However, P-CPLEX was not able to find a feasible solution within 12h for instance 24. Given the complexity of the models, it is difficult to justify this observed behavior. It seems that P-CPLEX probably generated excessive cuts in some nodes of the branch-and-bound during the solution process of instance 24, losing overall efficiency. Nevertheless, it is possible to affirm that the simple use of cutting plane parameters does not guarantee a robust solution process for this specific problem. The matheuristic obtained very competitive quality solutions in comparison with CPLEX and P-CPLEX for all instances. The matheuristic solved all 24 instances, and obtained better or equal solutions in 21 instances, specially for the large and very large ones. The matheuristic was able to offer solutions with an average relative deviation reduction of 10.8% and 3.1% in comparison with CPLEX and P-CPLEX, respectively, considering the instances solved by these two optimization methods. CPLEX, P-CPLEX and the matheuristic found the same number of vessels, with the exception of the instances not solved by one of the two first approaches. The reduction of the total costs by the matheuristic was obtained by a better usage of the vessels' capacity, lower cost routes, and lower utilization of port resources. Succintly, the greater the complexity of the instance, larger the difference in solution between CPLEX/P-CPLEX and the matheuristic. This fact is exemplified by instance 24, a very difficult one to solve, in which the heuristic reduced the total costs in 43.1% in comparison with the solution obtained by CPLEX. All computed costs were reduced by the matheuristic in comparison with CPLEX, but the savings were mainly due to the capability of the matheurisitc to find, very quickly, vessel routes that reduced the inversion of the geographical orientation costs as indicated in Table 6.

Tabela 6 – Solution costs of instance 24

Solution Costs ($)	CPLEX	Matheuristic
Usage of the ports	1228500	1094500
Vessel chartering	1440500	1261800
Demurrage Cost	1150000	1030000
Inversion of orientation	2880000	420000
Early or late arrival	5940	4210

Results in Table 5 shows that the matheuristic is more efficient than CPLEX and P-CPLEX, regardless of the instance dimension. The matheuristic decreased the required CPU time by average speed factors of 9.5 and 9.8 in comparison with CPLEX and P-CPLEX, respectively. However, better results could be obtained by the matheuristic if a loss in efficiency was tolerated, with a possible decrease in the solution quality. We could not find a correlation between the instance dimensions and the CPU time reductions. Further experimentation may be needed.

Overall, the developed matheuristic offered a very competitive approach to solve the problem, independently of the instance dimension. Although the matheuristic did not offer the best solution for all tested instances, it obtained near optimal solutions very fast. For large instances, it was the most effective and efficient method by far. The results were presented to the logistics department staff of the company and the schedulers responded positively to the optimization approach.

The efficiency and quality of the solutions by the matheuristic caused a very good impression. The possibility of shortening the planning time, from months to days, and to obtain better solutions can lead to very desirable effects, as follows: (i) to reduce the overall planing time, allowing better estimations of the demand of each products by the units' managers based on historical and current orders. As a result, the plan would better match supply and demand of each unit, simultaneously reducing stocks and increasing the service level (specially for premium products); and (ii) to offer an effective negotiation tool for decreasing costs with vessel chartering. Both issues are very important in the current strategy of consolidating the sustainability of the company in Brazil by offering customized fertilizers for customers, rather than simple commodities.

2.6.3 Solving a more general problem

Towards better evaluating the optimization approach, we decided to carry out additional experiments, validating the developed matheuristic using a diverse setting. We considered a more generic cargo routing and scheduling problem (GCRSP), without some of the peculiarities of the fertilizer company, as discussed in the last paragraph of Section 2.4. The three optimization methods were applied to solve model (1)–(28), (31), (32), (36)–(49). In addition to the restrictions (29)–(30) and (33)–(35), decision variables aux_{vw} were also disregarded. We used the same CPLEX and P-CPLEX CPU time limitations as in the previous experiments, six hours for instances 1–23, and twelve hours for instance 24. We also used strategy A and the same t_s values for running the matheuristic. As the cardinality of the integer variables set is smaller for the more general problem, the matheuristic solved the problem with six iterations. Table 7, with a similar layout of Table 5, compares the performance of the three optimization methods for the new problem, using the same real-life planning instances of the previous experiments.

Tabela 7 – Comparison of the optimization methods for the GCRSP

		CPLEX			P-CPLEX			Matheuristic		
Inst	OP-DP-P	Solution	CPU	V	Solution	CPU	V	Solution	CPU	V
1	2-4-7	1206330	10	2	1206330	11	2	1206330	2	2
2	2-5-5	1035600	901	4	1035600	932	4	1035600	60	4
3	2-5-6	795570	112	2	795570	125	2	795570	60	2
4	2-5-6	1575336	12086	3	1575336	12409	3	1575336	360	3
5	3-3-8	1279264	60	2	1279264	60	2	1279264	60	2
6	3-5-6	998740	120	2	998740	131	2	998740	60	2
7	3-5-7	1205990	4812	3	1205990	4257	3	1205990	120	3
8	4-5-6	2063252	21600	3	2142267	21600	3	2096902	1260	3
9	5-7-6	3971404	21600	3	3896134	21600	3	3578946	1800	3
10	4-4-6	2161936	21600	4	2161936	21600	4	2161936	1800	4
11	4-4-6	1592063	21600	4	1592063	21600	4	1623734	1800	4
12	3-6-8	3556860	21600	4	2344300	21600	4	2050900	1800	4
13	5-5-7	2137020	21600	4	2137020	21600	4	2094670	1800	4
14	5-5-8	3771470	21600	4	3972390	21600	4	3154250	1800	4
15	5-5-7	3541250	21600	4	3765360	21600	4	2356310	1800	4
16	5-7-5	2248891	21600	4	2302556	21600	4	2189441	1800	4
17	5-7-9	1610440	21600	3	1577120	21600	3	1696610	1800	4
18	5-8-10	3251900	21600	4	3138110	21600	4	2069070	1800	4
19	6-9-10	3614220	21600	4	3491380	21600	4	2189720	1800	4
20	6-5-9	2975949	21600	4	2989081	21600	4	2963128	1800	4
21	4-5-8	1695690	21600	4	1704610	21600	4	1679530	1800	4
22	6-7-11	6898418	21600	7	7210425	21600	4	3919573	1800	7
23	7-7-10	6289491	21600	7	-	21600	-	4410214	1800	7
24	6-8-10	5891770	43200	7	-	43200	-	3858310	4260	7

As expected, the solutions for the GCRSP (Table 7) generally have lower costs than the solutions presented in Table 5, independently of the optimization approach. The decrease in the final solution was of 7.3% and 6.5%, on average, for CPLEX and the matheuristic, respectively. However, for some instances, this difference can be as large as 40% for CPLEX (instance 17), and 19% for the matheuristic (instance 23). The influence of the peculiarities of the rel-life problem in the final solution is dependent on the instance. In general, but not always, as large the instance, greater the influence. However, this behavior could not be observed concerning CPU time for CPLEX and P-CPLEX. Some instances were solved quicker for the real-life problem than for the GCRSP by these two solution methods. It seems that the deleted constraints can speed up the solver to find worse solutions faster, for some instances. We were no able to find a pattern for this behavior. Surprinsigly, for instance 24, the matheuristic found a slightly better solution for the real-life problem than for the GCRSP. A possible explanation for this counter intuitive result is the CPU times used for solving this instance (6300s for the real-life problem versus 4260s for the GCRSP).

Comparing the optimization approaches, the behavior of the results for the GCRSP was quite similar with the ones obtained in the specific problem of the company. There is no significant differences in the total solution costs for the small to medium instances among the three solving methods. CPLEX and the matheuristic were able to solve all instances given the CPU time limitations used. Although P-CPLEX has obtained better results than CPLEX for some instances, the former was not able to solve the two largest instances (23 and 24). It seems that P-CPLEX can only be applied to small–medium instances, lacking robustness to solve all possible real-world problems within a reasonable computation time. For large instances, the matheuristic obtained better solutions than CPLEX. The matheuristic equaled or outperformed CPLEX and P-CPLEX solutions in 21 instances. The matheuristic was able to offer solutions with maximum relative deviations of around 5% and 7% in comparison with CPLEX and P-CPLEX, respectively. As the instance dimensions increased, the matheuristic was able to improve the solution quality. If we consider the instances with more than 5 origins or destinations ports, the matheuristic improved on average 19.8% in comparison with CPLEX solutions. Simultaneously, the matheuristic is more efficient than both CPLEX and P-CPLEX, regardless of the instance dimension. The matheuritic decreased the required CPU time by an overall average factor of 13 in comparison with CPLEX and P-CPLEX. With the exception of instance 17, CPLEX, P-CPLEX and the matheuristic found the same number of vessels. It seems the value of t_s was too small for the matheuristic to solve the problem with 3 vessels as carried out by CPLEX and P-CPLEX. Overall, the set of presented results have proved that the matheuristic is a robust method to solve the heterogeneous fleet with dedicated compartments, multi-product, flexible cargo sizes, split load deep-sea cargo routing and scheduling problem, considering different settings.

2.7 CONCLUSIONS

This paper introduces an optimization approach for solving a real-life shipping cargo routing and scheduling problem faced by a chemical company in its raw material supply for Brazilian mixing units. The problem can be characterized as a multi-product pickup and delivery problem with heterogeneous fleet, dedicated compartments, TWs, and draft limits. Feasible routes should comply with several constraints, some peculiar to the particular real-life problem. Although our optimization approach was developed based on a case study, it can be easily implemented in analogous maritime problem faced by companies transporting large quantities of bulk multi-products using tramp shipping as pointed out in Section 2.6.3.

We extended previous MILP formulations developed for the tramp cargo routing and scheduling (ARNESEN et al., 2017; MALAGUTI et al., 2018), introducing heterogeneous fleet and multi-products. CPLEX, including controlled cuts set to moderate, was applied for solving small to medium instances, obtaining relatively good solutions within a reasonable time. For large real-life instances, we developed a matheuristic approach, integrating a relax-and-define strategy with several in-built cutting plan heuristics within CPLEX. The matheuristic was evaluated using real-life instances, proving to be quite appropriate to solve the maritime problem, since it outperformed CLEX results in almost all instances tested, both in efficiency and efficacy. Particularly, the matheuristic significantly obtained better solutions, much quicker, than human schedulers and CPLEX, specially for large and extra-large instances, demonstrating its ability to cope with different planning contexts. Similar results were obtained when the optimization approaches were applied to a more general problem, without the specific peculiarities of the company context.

The research team is proceeding to develop a decision support systems with a friendly graphical interface for the effective use of the optimization approach by the company. Further, new algorithms and models are being considered to enhance the optimization approach. One idea is to tighten the formulation by introducing different families of valid inequalities developed by solving VRPs (ROPKE; CORDEAU, 2009). Further, a multi-objective formulation, considering simultaneously total costs and planning makespan minimization, is a special request by company managers.

3 MULTI-OBJECTIVE OPTIMIZATION OF THE MARITIME CARGO ROUTING AND SCHEDULING PROBLEM

ABSTRACT

This paper addresses the multi-objective maritime cargo routing and scheduling problem (MO-CRSP), in which the delivery of bulk products from pickup to delivery ports is served by a heterogeneous fleet of vessels. A mixed integer linear programming (MILP) model is formulated to simultaneously minimize total operation costs, the scheduling makespan, and delays in selected deliveries. The model accounts for several real features, such as time windows, capacity of the vessel's compartments, and ports requirements. A fuzzy weighted max–min method was applied to solve the problem. Two heuristics were developed to effectively handle the complex generated MILP models during the solution process. Experiments were conducted to evaluate the optimization approach using real life instances provided by a fertilizer company. Finally, a case study shows that the developed model and algorithmic framework are flexible and effective in coping with real problems, incorporating specific business rules from different companies.

3.1 INTRODUCTION

We introduce a new problem class in the context of maritime transportation, the multi-objective, multi-commodity heterogeneous fleet cargo routing and scheduling problem with time windows, draft limits, and split loads (MO-m-CRSPTW-DL-SL) which generalizes the CRSP characterized by Christiansen et al. (2013). In the MO-m-CRSTW-DL-SL setting, a heterogeneous fleet of vessels operates highly constrained routes to load multiple bulk products from a set of pickup ports and unload them in a set of delivery ports, fulfilling a deterministic demand from a set of customers. All cargo should firstly be collected and only then delivered within specified time windows in each port in a route. Early or late arrivals/departures of a vessel in a port are very costly, and therefore the synchronization of vehicle scheduling and port operating time windows is an important characteristic of the problem. Also, delays in deliveries may cause disruptions in the companies' manufacturing process, jeopardizing the productivity of customers. The nature of the products prevents them from being mixed, they must be transported in dedicated compartments of the vessels. Further, the routes are constrained by several operational requirements such as draft limits and berth utilization of the ports. The solution of the problem specifies (i) the sequence of ports to be visited by each vessel; (ii) the amount of each product to be loaded/unloaded in each port by a vessel; and (iii) the arrivals and departures schedule of the vessels in/from ports. However, as the vessel master is solely responsible for allocating the products in the compartments (CHRISTIANSEN et al., 2011; STANZANI et al., 2018), due to the stability and structural strength of each vessel, and sea conditions on the route, this task is partially addressed in the route planning, only guaranteeing that the transported diversity and amounts of products respect the number and nominal capacity of the ships' compartments, respectively. Furthermore, cargo allocation is a very difficult subproblem to solve in the context of the CRSP (HVATTUM et al., 2009). The overall plan should simultaneously minimize total transportation costs, scheduling makespan, and delays in some deliveries.

This study is motivated by the problem faced by fertilizer companies in Brazil that manage their raw material logistics planning. The companies use charter vessels to execute the transportation of the required raw materials from European ports to fulfill the demand for mixing units in Brazil. The planning is executed by experienced managers using spreadsheets, without the help of any optimizing tool. The planning process starts well ahead of its execution due to the difficulty of finding an acceptable, feasible solution by the logistics team. The increasing introduction of customized products to specific clients led the companies to optimize their raw material logistics towards simultaneously offering better service levels and obtaining higher profit margins, decreasing the operational costs. The new market environment has fomented the use of multi-objective optimization approaches.

The MO-m-CRSPTW-DL-SL is a maritime variant of the multi-objective pickup and delivery problem with time windows (MO-PDPTW) (DUMAS et al., 1991), a vehicle routing problem (VRP). Although the single objective of minimizing the cost is still dominant in the VRP literature (BRAEKERS et al., 2016), the problem is multi-objective in nature (JOZEFOWIEZ et al., 2008). In real life, decision-makers (DMs) consider additional objectives beyond costs, such as the optimization of the number of customer visits, the minimization of total lengths, and optimization of the makespan. There is crescent literature in the MO-VRP, and also in the MO-m-PDPTW. In a succinct analysis, the multi-objective approach is formulated by introducing extensions or adaptations to the single-objective modeling. The MO-VRP has been solved using (JOZEFOWIEZ et al., 2008): (i) scalar methods, mainly weighted aggregation using local search algorithms (PAQUETE; STÜTZLE, 2003), specific heuristics (ZOGRAFOS; ANDROUTSOPOULOS, 2004), and genetic algorithms (OMBUKI et al., 2006); (ii) Pareto dominance methods, mainly using multi-objective evolutionary (WANG et al., 2016; BRAVO et al., 2019), and hybrid algorithms (ZHANG et al., 2020); and (iii) non-scalar and non-Pareto algorithms, which includes ant colony systems (ZHANG et al., 2019), and particle swarm (ZOU et al., 2013) optimization.

However, peculiar characteristics of the CRSP setting it apart from the m-PDPTW (SANTOS et al., 2020), such as: (i) ports and vessels have rigorous capacities and draft limits that must be regarded; (ii) there is no central depot; and (ii) partial loading/unloading of products in the same vessel. So, it is unclear whether the approaches specifically developed for the the more generic MO-PDPTW/MO-VRPTW can be directly applied to the MO-m-CRSPTW-DL-SL.

In the domain of shipping routing and scheduling, the CRSP is a very important research topic, being a typical problem of bulk products transported by tramp or industrial shipping (CHRISTIANSEN et al., 2013). The CRSP is focused on the cargo to be load/unload in pickup/delivery ports, respectively, to fulfill the fixed demand of customers, respecting the time windows imposed by contractual deadlines (AL-KHAYYAL; HWANG, 2007). Given the diversity of real world applications and case studies described in the literature, the CRSP has been applied to several contexts. The problem was considered an m-PDPTW (FAGERHOLT, 2001) or a traveling salesman problem with time windows (FAGERHOLT; CHRISTIANSEN, 2000b) and draft limits (ARNESEN et al., 2017), incorporating real features such as split loads (KORSVIK et al., 2011), flexible cargoes (KORSVIK; FAGERHOLT, 2010), and dedicated compartments (FAGERHOLT; CHRISTIANSEN, 2000a). The problem was modeled as a single-objective multi-commodity arc-flow and solved using column generation (BRØNMO et al., 2010), branch-and-cut (MALAGUTI et al., 2018), tabu search (TROTTIER; CORDEAU, 2019), and neighborhood search (KORSVIK et al., 2011). More recently, Santos et al. (2020) modeled and solved the CRSP, simultaneously considering the following real features: heterogeneous fleet, multiple products, time windows, draft limitation, flexible cargoes, and split loads. A matheuristic, based on a modified relax-and-fix algorithm, was developed to find good solutions towards minimizing the total transportation costs with efficiency.

Studies in using multi-objective in the maritime routing and scheduling problem are still scarce. In a review by Mansouri et al. (2015) about the consideration of multi-objective decisions in sustainable maritime shipping, no explicit multi-objective optimization (MOO) approach was cited. Multiple objectives were mainly considered as constraints. Chan et al. (2014) developed a dynamic scheduling of oil tankers with the splitting of cargo at pickup and delivery ports, using a multi-objective ant colony-based approach. The developed algorithm proved very efficient in comparison with a non-dominated sorting genetic algorithm II (NSGA II) towards finding good solutions for instances with dozens of pickup and delivery ports with a heterogeneous fleet of oil tankers. Although containing interesting ideas, the problem considered by Chan et al. (2014) is much less restrained than the problem tackled in this paper, neglecting time windows, port drafts, and dedicated compartments. The problem only considers a single product. Recently, MOO approach for planning liner shipping service considering uncertain port times were introduced by Song et al. (2015). The problem was formulated as a stochastic nonlinear programming model, considering three objectives, as follows: (i) annual total vessel operating costs; (ii) average schedule unreliability; and (iii) CO_2 emissions. The model was solved using an NSGA II algorithm and applied to a container shipping service route. De et al. (2017) developed a bi-objective model addressing the sustainable ship routing and scheduling with time windows and draft restrictions, maximizing the overall profit incurred of providing shipping operations within a planning horizon, and minimizing the total carbon emission incurred by the ship fleet. The model was solved by combining NSGA-II and multi-objective particle swarm optimization (MOPSO). However, both problems are directed to container ships, which are large ocean vessels that operate, in general, as line service, transporting goods using regular transit routes on fixed schedules. Further, cargoes are aggregated in containers, without considering the products individually. To the best of our knowledge, no previous research work uses MOO for handling the maritime CRSP.

In this paper, we develop a multi-objective optimization approach for the m-MO-CRSPTW-DL-SL, considering three objectives, extending the single-objective formulation presented by Santos et al. (2020). The main objective of the optimization approach is to find good solutions, as close as possible to the Pareto front, towards supporting the decision-making process of human planners. These objectives include: total costs, makespan, and delays in selected deliveries. A fuzzy weighted max–min multi-objective method has been developed to solve the problem. Towards finding solutions for the complex mixed integer linear program (MILP) models involved in the weighted max-min, two methods were introduced, using mathematical programming technologies offered by contemporary commercial MILP solvers. The proposed algorithms were compared using real-world planning instances kindly provided by a fertilizer company. Further, the MOO approach was applied to a case study, illustrating the effectiveness and the ability of the modeling approach to incorporate several additional requirements. The major contribution of this paper is to introduce an MOO approach for the maritime CRSP, not only capable of incorporating the several aspects of the problem, but also to integrate operational peculiarities and business rules of different companies and chartering modalities.

This paper is organized as follows. The problem is described and formulated in Section 4.3. Section 3.3 introduces the solution method, detailing the algorithms developed to solve the several MILP models generated during the solution process. In Section 3.4, computational experiments to evaluate the performance of the algorithms are presented. Section 4.7 describes a real-case study carried out using data from a multinational fertilizer company's Brazilian branch. In the concluding Section 4.8, an overview of the results is presented and future research are elaborated.

3.2 PROBLEM DESCRIPTION AND FORMULATION

We consider in this study a multi-objective short-term deep-sea CRS planning. The plan is mainly used by chemical companies for the chartering process to transport raw materials from Europe to Brazil. Cargoes are transported by dry bulk tramp shipping, using contracts of affreightment (COA), in which a charterer agrees to pay freight, daily, for a shipowner to carry goods in the vessel for a specified time. The COA specifies a period, known as laytime, for loading and unloading the cargo. If laytime is exceeded, the charterer must pay a daily charge, known as demurrage. The contract also specifies the laydays, the period within which the shipowner has to make the vessel fully operative to the charterer at the place and time agreed in the charter party. The shipowner pays the fuel costs and crew costs during the term of the COA. The vessel may be delivered at an agreed port following a ballast voyage, which costs are paid by the charterer to the shipowner as a “ballast bonus”. Laycan and ballast voyages are not addressed in our cargo route and schedule planning. The plan is used for internal consumption, mainly to negotiate product deliveries to end customers. Vessels' locations are unknown during this planning phase. Both aspects become relevant in the next contracting phase of the chartering. In general, handymax or handysize bulk carriers are used, presenting a capacity of approximately 35,000–60,000 deadweight tonnage (DWT). Each vessel contains multiple compartments that can be used to transport different products. Because the supplies and the demands of products in some ports are lower than the vessel capacity, a route may include the pickup and delivery of products in several ports. Each pickup port may have a high supply for one or more products, whereas one or more delivery ports may have a high demand for the same product in different or equal quantities. Furthermore, most origin and destination ports have severe draft restrictions, limiting vessel usage based on its load condition. We assume that the Port Authority has defined the draft limits for each terminal in a port, which are expressed in terms of the maximum weight a vessel can safely carry on board.

There are restrictions on the minimum load per vessel and the minimum amount per product transported by a vessel to avoid underutilization. We assume that all vessels, regardless of size, travel at a cruise speed of fifteen knots. As a consequence, the sailing times between ports are solely determined by the distance within them. The speed in each port is also constant and regulated by the Port Authority. The vessel draft after each load and unload operation depends highly on its physical dimensions, cargo weighting, hydrostatic curves, and sea conditions. For the sake of simplicity, we assume that the vessel draft is determined by its current cargo, following (ARNESEN et al., 2017).

Feasible vessel routes must meet a number of requirements and constraints. All cargo must be collected before any delivery can occur, without the possibility of transshipment in any port. A vessel route in the CRSP has a rigid structure $P, P, \ldots, P, D, D, \ldots, D$, where a delivery port (denoted by D) cannot appear before any pickup port (denoted by P) in the sequence. Furthermore, as presented in (FAGERHOLT; RONEN, 2013), our problem involves flexible cargo sizes and split loads. Another critical requirement in a route is the delivery of a product close to the date specified by each mixing plant in order to fulfill its production schedule. As a result, time windows are imposed for cargo loading and unloading at each port in order to comply with negotiated delivery dates of final products to clients by the commercial department. When designing a route, the following times should be considered: loading/unloading time for each cargo, compartment washing time after delivering a cargo, waiting time for the product to be released for land transport in accordance with customs legislation, and waiting time for a vessel to dock at a terminal. The load/unload times at each port are dependent on the amount and type of cargo handled and the capacity and speeds of available equipment (ship-loaders and conveyor belts). As a consequence, the same quantities of raw materials present a reasonable variability of load/unload times in different ports, the fertilizer companies define average values, based on historical data, for each port, regardless of the quantities moved. Since it is a pre-contract planning of the real transportation, we used these values for simplification, reducing the number of required parameters to run the model.

Our problem consists of defining: (i) each vessel's route and schedule; and (ii) which product, and the respective amount, is to load/unload into which vessel at which port. The overall plan should simultaneously minimize three objectives. The first objective is related to the completion time of the last delivery by a vessel in the planning horizon, known as makespan. As deep-sea transportation is prone to various disruptions caused by weather conditions and port problems, the main objective of minimizing makespan in the CRSP is to better integrate the supply of raw materials and the production master plan in the mixing units. Furthermore, by reducing the variability of deliveries of the same raw material by different vessels, possible delays can be reduced to all demands involved in the planning. The second objective refers to the total costs faced by the deep-sea operations. Although costs and makespan are rarely considered as separated objectives in the general MO-VRP literature (JOZEFOWIEZ et al., 2008), since they are some way related (minimizing the makespan also reduces the freight costs), this is not the case in the specific context of the maritime CRSP. The minimization of costs often implies in vessels visiting more ports, frequently increasing the makespan of a plan. The reduction in costs can significantly increase the makespan in the CRSP for some instances, as demonstrated in our experiments. They are considered independent in the preference by all contacted managers/schedulers. They always prefer minimum cost, independent on the value of the makespan, and vice-versa. In summary, managers of the fertilizer companies approach costs and makespan as a classic trade-off in the real world, justifying their choice in the multi-objective formulation of the CRSP. To minimize the delay of some deliveries is the third objective. This objective has become highly relevant since Brazilian fertilizer companies are focusing on producing premium products rather than commodities to increase profit from operations. These products are customized according to the specific customers' needs. Possible delays in the delivery of final products can have a very negative effect on the long-term relationship with these customers. Nowadays, to minimize delay is a very important strategic objective, strictly linked to the sustainability of the fertilizer companies.

The MO-m-CRSPTW-DL-SL is modeled using the network and formulation by (SANTOS et al., 2020) as a starting point, but considering modifications to account for multi-objective, delays in deliveries, and multi-vessel utilization with overlapping docking times. Table 4.4 presents the mathematical notation following (SANTOS et al., 2020). Sets and parameters are expressed as upper case letters, while variables and indexes use lower case letters.

The problem is defined on a directed graph $G = (N, A)$, where set $N = N^V \cup \{s\} \cup \{f\}$ is the set of nodes and $A = \{(i,j) | (i \in N^P, j \in N^V) \land (i \in N^D, j \in N^D) \land (i \in \{s\}, j \in N^P) \land (i \in N^D, j \in \{f\})\}$ is the set of arcs, where N^V is the set of pickup and delivery ports, N^P is the set of pickup ports, and N^D is the set of delivery ports. Dummy nodes s and f represent artificial start and end depots, respectively. All out-going arcs from s and in-going arcs to f are travelled without costs and with null time, since laycan and ballast voyages are unconsidered. There are no arcs in graph G, connecting delivery to pickup ports. Each elementary path in the graph from s to f, in which the constraints related to TWs, draft, and capacity are respected, corresponds to a feasible vessel route. Fig. 4 illustrates feasible routes of two vessels associated with 3 pickup and 2 delivery ports. In the MO-m-CRSPTW-DL-SL,

- pickup ports $i \in N^P$ has a supply (Q_{ip}) of product p, and a demand (-Q_{ip}) in the delivery ports $i \in N^D$;
- pickups and deliveries are performed by a heterogeneous fleet of charter vessels that differ on dimensions, capacity, and number of compartments;

- products cannot be mixed in the vessels' compartments, but feasible bulkheads can be used to divide the nominal cargo compartments;
- vessel v must transport cargo within the interval $[R_v^V, \sum_{c \in C(v)} Y^V cv]$;
- the number of vessels being simultaneously loading/unloading cargo in port j is limited by parameter U_j^N. If U_j^N is set to zero, only a vessel can be served anytime in port j;
- although transshipment is strictly forbidden, partial loading/unloading of a vessel compartment by the same product is allowed;
- products can be loaded in pickup ports $j \in N^P$ in the TW interval $[S_j^T, F_j^T]$, and unloaded in delivery ports $j \in N^D$ in the interval $[A_j^T, L_j^T]$;
- products $p \in P$ are expected to be delivered in port $j \in N^D$ in time D_{jp}^T.

Based on the network and notation presented, the MO-m-CRSPTW-DL-SL can be formulated as a MILP, as follows:

$$\min Z_1 = \alpha \tag{50}$$

$$\begin{aligned}\min Z_2 = & \sum_{v \in V} C_v^V \left(\sum_{i \in N^D} t_{ifv}^A + \sum_{j \in N^P} t_{sjv}^A \right) + \sum_{i \in N} \sum_{j \in N^V} \sum_{v \in V} P_j^C x_{ijv} \\ & + \sum_{i \in N^V} \sum_{j \in N^D} \sum_{p \in P} \sum_{v \in V} H^C t_{ijpv}^E + \sum_{i \in N} \sum_{j \in N^V} \sum_{v \in V} W_v^C E_j^T x_{ijv}\end{aligned} \tag{51}$$

$$\min Z_3 = \sum_{i \in N^V} \sum_{j \in N'} \sum_{p \in P'} \sum_{v \in V} t_{ijpv}^L \tag{52}$$

Sets	
N^V	set of pickup and delivery ports
N^P	set of pickup ports
N^D	set of delivery ports
N	set of all ports, including the dummy start (s) and finish (f) ports
P	set of products
V	set of vessels
$C(v)$	set of compartments of vessel v
Parameters	
Q_{ip}	stock level of product p in port i
D^T_{jp}	expected date of arrival of product p in port j
A^T_j	earliest starting time for the delivery of products in port j
L^T_j	latest starting time in days for the delivery of products in port j
O^T_j	loading and unloading time in port j
E^T_j	estimated queuing and clearance times in port j
S^T_j	estimated earliest starting time for the pickup of a cargo in port j
F^T_j	estimated latest starting time for the pickup of a cargo in port j
L^T_{ij}	sailing time of vessel v between ports i and j
P^N_j	draft limit in ton of port j
B^N_j	maximum number of berths in port $j \in N^D$
U^N_j	maximum number of vessels being simultaneously served in of port j
K^P_v	minimum amount of product p to be loaded in vessel v
R^V_v	minimum amount of cargo in compartments of vessel v
W^C_v	demurrage cost of vessel v
P^C_j	cost of using port j facilities
C^V_v	freight price of vessel v
H^C	daily cost for disrespecting the time window of a pickup/delivery
Y^V_{cv}	nominal capacity of compartment c of vessel v
Decision variables	
l_{ijpcv}	amount of product p transported using arc (i,j) by vessel v at compartment c
t^A_{ijv}	arrival time of vessel v using arc (i,j)
t^D_{ijv}	departure time of vessel v using arc (i,j)
t^E_{ijpv}	early arrival or delay of vessel v carrying product p using arc (i,j)
t^L_{ijpv}	arrival delay of vessel v at port j carrying product p using arc (i,j)
x_{ijv}	binary variable that indicates if arc (i,j) is used by vessel v
y_{ijpv}	binary variable that indicates if arc (i,j) is used by vessel v to transport product p
y^D_{ijpv}	binary variable that indicates if product p is unloaded off vessel v in port j, using arc (i,j)
z_{vwj}	binary variable that indicates if vessel v starts to be served in port j while vessel w has already been served in the same port

The objective function Z_1 minimizes the scheduling makespan (α), computed by constraints (53). Objective function Z_2 minimizes the total transportation costs. Objective function Z_3 minimizes the delay of the delivery of a set of products $P' \subseteq P$ for a set of ports $N' \subseteq N^D$ connected to special clients.
Subject to:

$$\alpha \geq t^A_{ifv} \qquad \forall i \in N^D,\ \forall v \in V \tag{53}$$

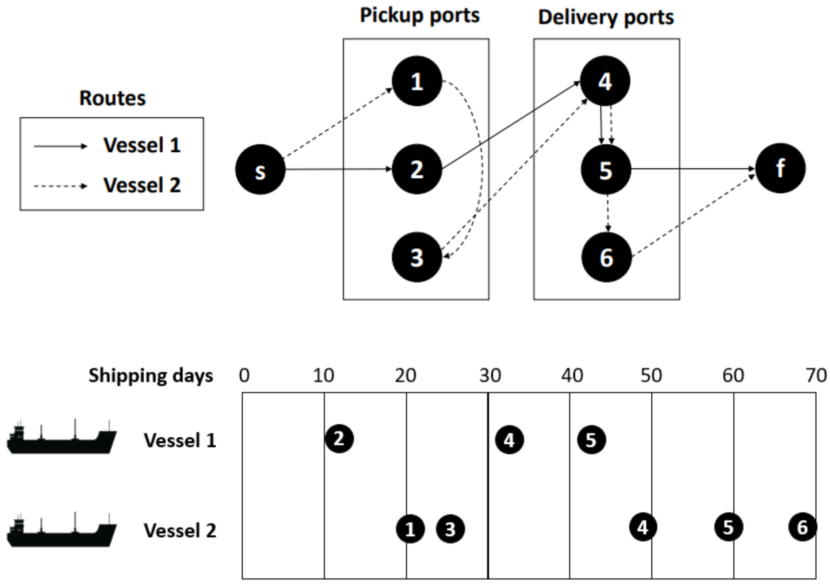


Figura 4 – Illustration of two vessel routes

Supply-demand constraints

$$Q_{ip} - \sum_{j \in N^V} \sum_{v \in V} \sum_{c \in C(v)} l_{ijpcv} + \sum_{h \in N^V} \sum_{v \in V} \sum_{c \in C(v)} l_{hipcv} \geq 0 \qquad \forall i \in N^V,\ \forall p \in P \tag{54}$$

Constraints (115) ensure that the supply and demand of each product are respected in the pickup and delivery ports.

Ship load constraints

$$Q_{ip} - \sum_{j \in N^V} l_{ijpcv} + \sum_{h \in N^V} l_{hipcv} \geq 0 \qquad \forall i \in N^V, \forall p \in P, \forall v \in V, \forall c \in C(v) \tag{55}$$

$$\sum_{h \in N^P} l_{hipcv} - \sum_{j \in N^V} l_{ijpcv} \geq 0 \qquad \forall i \in N^P, \forall p \in P, \forall v \in V, \forall c \in C(v) \tag{56}$$

$$\sum_{i \in N^P \cup \{s\}} l_{ijpcv} - \sum_{k \in N^V} l_{jkpcv} \leq 0 \qquad \forall j \in N^P,\ \forall p \in P, \forall v \in V, \forall c \in C(v) \tag{57}$$

$$\sum_{i \in N^P} \sum_{j \in N^D} \sum_{p \in P} \sum_{c \in C(v)} l_{ijpcv} \leq \sum_{c \in C(v)} Y^V_{cv} \qquad \forall v \in V \tag{58}$$

$$\sum_{i \in N^P} \sum_{j \in N^D} \sum_{p \in P} \sum_{c \in C(v)} l_{ijpcv} \geq \sum_{j \in N^V} R^V_v x_{sjv} \qquad \forall v \in V \tag{59}$$

$$\sum_{c \in C(v)} l_{ijpcv} \geq K^P_v y_{ijpv} \qquad \forall i \in N^P,\ \forall j \in N^V,\ \forall p \in P,\ \forall v \in V \tag{60}$$

$$l_{ijpcv} \leq Y^V_{cv} y_{ijpv} \qquad \forall i, j \in N^V, \forall p \in P, \forall v \in V, \forall c \in C(v) \tag{61}$$

$$\sum_{i \in N^P} \sum_{j \in N^V} \sum_{p \in P} y_{ijpv} \leq |C(v)| \qquad \forall v \in V \tag{62}$$

Constraints (116) assure that only stocked products in a pickup port can be loaded in a vessel. Constraints (117) satisfy the demand of any product at each delivery port. Constraints (118) prohibit that a vessel discharge cargo in a pickup port. Constraints (119) respect the vessel capacity. Constraints (59) avoid the under utilization of vessels. Constraints (60) limit the minimum amount of a product being transported by a vessel. Constraints (120) guarantee that the capacity of each compartment of a vessel is respected. Constraints (121) limit the number of products transported by a vessel to the number of its compartments.

Flow constraints

$$\sum_{j \in N} x_{ijv} \leq 1 \qquad \forall i \in N,\ \forall v \in V \tag{63}$$

$$\sum_{i \in N} x_{ijv} \leq 1 \qquad \forall j \in N,\ \forall v \in V \tag{64}$$

$$\sum_{j \in N} x_{ijv} - \sum_{h \in N} x_{hiv} = 0 \qquad \forall i \in N,\ \forall v \in V \tag{65}$$

$$\sum_{i \in N^P} \sum_{j \in N^D} x_{ijv} \leq 1 \qquad \forall v \in V \tag{66}$$

$$x_{ijv} = 0 \qquad \forall i \in N^D,\ \forall j \in N^P,\ \forall v \in V \tag{67}$$

$$x_{fsv} = 0 \qquad \forall v \in V \tag{68}$$

$$x_{fjv} = 0 \qquad \forall j \in N^V,\ \forall v \in V \tag{69}$$

$$x_{isv} = 0 \qquad \forall j \in N^V,\ \forall v \in V \tag{70}$$

$$\sum_{p \in P} l_{ijpcv} - Y^V_{cv} x_{ijv} \leq 0 \qquad \forall i, j \in N, \forall v \in V, \forall c \in C(v) \tag{71}$$

Constraints (133) and (134) impose that a vessel cannot re-use any arc in the network. Constraints (135) are conservative flow constraints. Constraints (136) and (137) guarantee that no delivery can occur before all collects have been carries out. Constraints (138), (139) and (140) delimit that nodes s and f are super source and super sink nodes, respectively. Constraints (141) guarantee that a cargo can only be transported by an effectively used vessel and respecting the capacity of each vessel compartment, logically connecting continuous flow and binary variables concerning the vessel utilization.

Time Constraints

$$t^A_{ijv} \geq S^T_j x_{ijv} \qquad \forall i \in N,\ \forall j \in N^P,\ \forall v \in V \tag{72}$$

$$t^D_{ijv} \leq F^T_j x_{ijv} \qquad \forall i \in N,\ \forall j \in N^P,\ \forall v \in V \tag{73}$$

$$t^D_{ijv} \geq t^A_{ijv} + E^T_j + O^T_j + M(x_{ijv} - 1) \qquad \forall i \in N,\ \forall j \in N^V,\ \forall v \in V \tag{74}$$

$$t^A_{ijv} \geq t^D_{hiv} + L^T_{ij} + M(x_{ijv} - 1) \qquad \forall h \in N,\ \forall i \in N^V,\ \forall j \in N^V,\ \forall v \in V \tag{75}$$

$$t^E_{ijpv} \geq \mid D^T_{jp} - E^T_j - t^D_{ijv} \mid + M(y^D_{ijpv} - 1) \qquad \forall i \in N^V,\ \forall j \in N^D,\ \forall p \in P,\ \forall v \in V \tag{76}$$

$$t^A_{ifv} \geq t^D_{hiv} + M(x_{ifv} - 1) \qquad \forall i, h \in N^V,\ \forall v \in V \tag{77}$$

$$t^A_{siv} \leq t^A_{ijv} - S^T_j - O^T_i - E^T_i - M(x_{ijv} - 1) \qquad \forall i, j \in N^V,\ \forall p \in P,\ \forall v \in V \tag{78}$$

$$t^A_{ijv} + E^T_j - M(x_{ijv} - 1) + Mz_{vwj} \geq t^D_{kjw} + M(x_{kjw} - 1) \qquad \forall i, k \in N,\ \forall j \in N^V,\ \forall v, w \in V, v \neq w \tag{79}$$

$$t^D_{ijv} \leq D^T_{jp} + L^T_j - E^T_j + M(-y^D_{ijpv} + 1) \qquad \forall i \in N^V,\ \forall j \in N^D,\ \forall p \in P,\ \forall v \in V \tag{80}$$

$$t^D_{ijv} \geq D^T_{jp} - A^T_j - E^T_j - M(y^D_{ijpv} - 1) \qquad \forall i \in N^V,\ \forall j \in N^D,\ \forall p \in P,\ \forall v \in V \tag{81}$$

$$t^L_{ijpv} \geq t^A_{ijv} + E^T_j - D^T_{jp} + M(y^D_{ijpv} - 1) \qquad \forall i \in N^V,\ \forall j \in N^{'}\ \forall p \in P^{'},\ \forall v \in V \tag{82}$$

where M represents a very large number.

Constraints (144) and (145) link the TW to the arrival and departure times of each vessel in/from each port, respectively. Constraints (146) specify when each vessel departs from a port. Constraints (147) determine the arrival time of each vessel from port i in port j. Constraints (148) account for a vessel's early arrival or late departure in each port. The starting and ending route times of each vessel are defined by constraints (149) and (150). The arrival time of each vessel in a port is determined by constraints 151. The departure time of each vessel from a port is determined by constraints (152) and (153). Constraints (154) are used to compute the delivery delay of product p in port j of vessel v using arc (i, j).

Port related constraints

$$\sum_{v \in V} \sum_{w \in V | w \neq v} z_{vwj} \leq B_j^N \qquad \forall j \in N^V \tag{83}$$

$$\sum_{v \in V} \sum_{w \in V | w \neq v} z_{vwj} \leq U_j^N \qquad \forall j \in N^V \tag{84}$$

$$\sum_{j \in N} \sum_{p \in P} \sum_{c \in C(v)} l_{ijpcv} \leq P_i^N \qquad \forall i \in N^V,\ \forall v \in V \tag{85}$$

$$\sum_{i \in N} \sum_{p \in P} \sum_{c \in C(v)} l_{ijpcv} \leq P_j^N \qquad \forall j \in N^V,\ \forall v \in V \tag{86}$$

Constraints (125) guarantee that the berth capacity of port j is respected. Constraints (126) restrain the number of vessels being simultaneously served in port j to parameter U_j^N. Constraints (127) and (128) assure that the port draft is respected when both entering or leaving a port, respectively.

Routing Constraints

$$y_{ijpv} - x_{ijv} \leq 0 \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{87}$$

$$t_{ijv}^D - Mx_{ijv} \leq 0 \qquad \forall i, j \in N,\ \forall v \in V \tag{88}$$

$$t_{ijv}^A - Mx_{ijv} \leq 0 \qquad \forall i, j \in N,\ \forall v \in V \tag{89}$$

$$y_{ijpv} - D_{jp}^T y_{ijpv}^D \leq 0 \qquad \forall i \in N^V,\ \forall j \in N^D,\ \forall p \in P,\ \forall v \in V \tag{90}$$

Constraints (129) state that each vessel can only transport a product using arc (i, j) if: (i) the vessel follows the path of arc (i, j), and (ii) the vessel is transporting the product in one of its compartments. Constraints (130) and (131) assure that arrival/departure times and vessel routes are compatible. Finally, constraints (132) ensure that a product is only unloaded from a vessel in a port if the product is being transported by the vessel on its route.

Domain of the variables

$$t_{ijv}^D, t_{ijv}^A, l_{ijpcv}, t_{ijpv}^E \geq 0 \qquad \forall i, j \in N, \forall p \in P, \forall v \in V, \forall c \in C(v) \tag{91}$$

$$x_{ijv}, y_{ijpv}, y_{ijpv}^D \in \{0, 1\} \qquad \forall i, j \in N,\ \forall p \in P,\ \forall v \in V \tag{92}$$

$$z_{vwj} \in \{0, 1\} \qquad \forall v, w_{v \neq w} \in V,\ \forall j \in N^V \tag{93}$$

As a VRP variant, the MO-m-CRSPTW-DL-SL is also a NP-hard problem. As a planning problem, our goal is to find non-dominant solutions, with acceptable values for the three objectives, for model (114) – (158). Further, the decision making process should be capable of interacting with planners, since the importance of each objective is highly context dependent.

3.3 SOLUTION METHOD

The general multi-objective model can be formulated, as follows:

$$\max Z_1, Z_2, \ldots, Z_k \tag{94}$$

$$\min Z_{k+1}, Z_{k+2}, \ldots, Z_l \tag{95}$$

st

$$x \in X \tag{96}$$

where $Z_1, Z_2, \ldots, Z_k$ are the positive objectives for maximization, $Z_{k+1}, Z_{k+2}, \ldots, Z_l$ are the negative objectives for minimization, and X is the set of feasible solutions.

The main objective in MOO is to choose non-dominant solutions, based on different levels of trade-off among the different objectives, from the Pareto front. In the planning of dry bulk cargoes, the main objective is to support the logistics team in finding the most preferred Pareto optimal solution according to the company's preferences and needs. The underlying assumption is that a good solution to the problem must be identified as implementable in practice and in accordance with each planning context. For instance, if some special order of a customer is involved, the level of service becomes more important. Otherwise, the cost is the most important objective. The human DMs play an important role in the planning process.

Scalar techniques and Pareto methods are very popular in handling the MO-VRP (JOZEFOWIEZ et al., 2008). The former is a set of *a priori* methods, in which decisions are made (based on DM) before searching a solution, while the latter is *a posteriori* method, in which a search is performed before making decisions by the DM. Both methods have well known strengths and weaknesses (EHRGOTT, 2008). Although Pareto methods, especially evolutionary algorithms, are becoming very popular to solve MOO (ZITZLER et al., 2000), they are not well suited to our specific problem, since they generate excessive "unnecessary" solutions, making the decision process for the planners a little bit confusing. Scalar methods can find solutions of interest to DMs. A scalar technique is more suited to solve the real MO-m-CRSPTW-DL-SL. There are several ways of converting the MOO to a single-objective program described in the literature (EHRGOTT, 2008), being the weighted sum the simplest and most used. However, literature in MOO addresses several drawbacks of this method in depicting the Pareto optimal set, and the proper scaling or normalization of the objectives (DAS; DENNIS, 1997; MARLER; ARORA, 2010). Based on these deficiencies, we choose another scalar technique, the fuzzy weighted max-min method, as introduced by Lin (2004). The key advantage of the weighted max-min method is that it can provide almost all the Pareto optimal points, even for a nonconvex Pareto front. It is relatively well suited for generating, with variation in the weights, a Pareto front. However, this method requires the minimization of individual single-objective optimization problems to determine the utopia point, which can be computationally expensive (CHANG, 2014), particularly for mixed integer multi-objective problems. Considering this drawback, special algorithms were customized/developed to overcome this difficulty.

In several real problems, a fuzzy perspective is assumed by the DMs. Membership functions $\mu_{Z_j}(x)$ are defined for each objective, and the solution achieves all objectives given a certain tolerance limit under the constraints. The problem consists of finding a solution for the reformulated formulation (AMID et al., 2011):

$$\tilde{Z}_i \geq\sim Z_i^o \qquad i = 1, \dots, k \tag{97}$$

$$\tilde{Z}_j \leq\sim Z_j^o \qquad j = k+1, \dots, l \tag{98}$$

st

$$g_s(x) = \sum_{i=1}^{n} a_{si} x_i \leq b_s \qquad \forall s \tag{99}$$

$$x_i \geq 0 \qquad \forall i \tag{100}$$

where Z_k^o and Z_l^o are the levels that the decision maker wants to reach, a_{si} and b_s are crisp values, and symbol $\sim$ indicates the fuzzy environment.

To solve this problem, Lin (2004) expanded the max-min operator approach (ZIMMERMANN, 1978), by proposing a weighted max-min model, in which the decision-maker provides relative weights ($\theta_k, k = 1, 2, \dots, l$) for the l fuzzy objectives with corresponding membership functions. This model finds an optimal feasible solution such that the ratio of the levels achieved is as close to the ratio of the weights as possible. This model can be stated as follows (AMID et al., 2011):

$$\max \lambda \tag{101}$$

st

$$\theta_k \lambda \leq f_{\mu_{Z_k}}(x) \qquad k = 1, 2, \dots, l \tag{102}$$

$$\sum_{k=1}^{l} \theta_k = 1 \tag{103}$$

$$\theta_k \geq 0 \qquad k = 1, 2, \dots, l \tag{104}$$

$$g_s(x) \leq b_s \qquad \forall s \tag{105}$$

$$x_i \geq 0 \qquad i = 1, 2, \dots, m \tag{106}$$

where the fuzzy membership function for maximization objectives ($\mu_{Z_k}(x)$) and for minimization ones ($\mu_{Z_l}(x)$) are as follows, respectively:

$$\mu_{Z_l}(x) = \begin{cases} 1, & Z_l \leq Z_l^- \\ 0, & Z_l \geq Z_l^+ \\ f_{\mu_{Z_l}} = \frac{Z_l(x) - Z_l^-}{Z_l^+ - Z_l^-}, & Z_l^- \leq Z_l(x) \leq Z_l^+ \end{cases} \tag{107}$$

$$\mu_{Z_k}(x) = \begin{cases} 0, & Z_k \leq Z_k^- \\ 1, & Z_k \geq Z_k^+ \\ f_{\mu_{Z_k}} = \frac{Z_k^+ - Z_k(x)}{Z_k^+ - Z_k^-}, & Z_k^- \leq Z_k(x) \leq Z_k^+ \end{cases} \tag{108}$$

where Z_k^+ and Z_l^- are the optimal single objective functions (individual maximum and minimum solutions) of positive objective Z_k and negative objective Z_l, respectively, and Z_k^- and Z_l^+ are the minimum and maximum values (worst solutions) of objectives Z_k and Z_l, respectively (AMID et al., 2011).

Algorithm 5 presents the customized version of the algorithm by Amid et al. (2011) to solve the fuzzy weighted max-min model for the MO-m-CRSPTW-DL-SL. The single-objective formulations to find the best solution and nadir points for each objective can be solved using the very efficient matheuristic by Santos et al. (2020). The algorithm solves several complex MILP formulations, mainly in Step 6. Model (101) – (104), (53) – (158) is a very difficult MILP to solve. Unfortunately, this model is solved several times, considering several weight combinations during a real decision making process. Using the traditional branch and bound algorithm implemented in powerful MILP solvers when applied to the maritime CRSP leads to low quality solutions and requires excessive computational time for instances with more than 6 products and 9 ports, that is incompatible with the solution of real-world problems as demonstrated in Santos et al. (2020). Another possibility, considering the limited number of ports, is to use a path flow based solution method, such as column generation or a previous enumeration of all possible feasible routes. However, both methods have difficult problems to overcome in practice. On one hand, the use of column generation involves the solution of a very complex NP-hard resource constrained shortest path as the dual problem (FEILLET et al., 2004), considering the number of the different constraints of the CRSP. On the other hand, to elicit several feasible routes for the path model requires the development of an effective and efficient algorithm, considering simultaneously the voyage sequences, use of the compartments, and the hard constraints. These issues and the tight due deadline to solve real cases have motivated the development of the two algorithms described next.

3.3.1 CPLEX-based algorithm

The developed algorithm has a major objective to improve and accelerate solutions obtained by the well-known CPLEX solver. Using the traditional branch-and-bound was inefficient in dealing with the weighted max-min model for real case instances during initial experimentation. The branch-and-bound got stuck in a node of the search tree, avoiding finding Pareto front solutions. Instead of developing a heuristic method, we used the contemporary validated resources offered by CPLEX to improve the solution search process.

The algorithm uses several inherent capabilities of the well-known IBM ILOG CPLEX 12.9, such as non-traditional tree-of-trees search, multiple default heuristics, solution improvement, symmetry detection, and cutting planes. Further, we used the same cutting planes routines

Algorithm 5: Fuzzy MOO solution method for the MO-m-CRSPTW-DL-SL, based on Amid et al. (2011)

1. Define model (114) – (158) as the MOLP to be solved.

2. Solve the MOLP as a single objective problem for each objective $i, i = 1, \ldots, k$, using the matheuristic developed by Santos et al. (2020). As this is the best value for each objective, set Z_i^+ as the upper bound of the i-th objective.

3. Solve the MOLP as a single objective, changing the optimization direction of each objective $j = k + 1, \ldots, l$, using the matheuristic developed by Santos et al. (2020). As this is the nadir point for each objective, set Z_j^- as the lower bound of the j-th objective.

4. For each objective $i = 1, \ldots, k$ find the membership function by using (108).

5. For each objective $j = k, \ldots, l$ find the membership function by using (107).

6. For each weight combination $(\theta_1, \theta_2, \ldots, \theta_l)$ do:

6.1. Formulate the problem as a multi-objective weighted max-min model (101) – (104), (53) – (158).

6.2. Solve the model of Step 6.1, using Algorithms 6 or 7, so as to find a non-dominant solution for the weight combination in analysis.

as in Santos et al. (2020), given the success of branch-and-cut algorithms when applied to the VRP and its variants (BRAEKERS et al., 2016). We call P-CPLEX the solver with implemented cutting planes routines by CPLEX. It should be noted that several powerful commercial MILP solvers have analogous routines to perform the same cutting planes.

Algorithm 6 outlines the developed algorithm. In the first step, the main objective is to obtain a good feasible integer solution to the problem, by applying P-CPLEX. Step 2 entails polishing the solution obtained in Step 1 using the algorithm proposed by Rothberg (2007), an evolutionary approach in which crossover and mutation operations are built within a MILP branch-and-bound. The solutions of the evolutionary approach are incorporated into the MILP search tree, and the solutions obtained by the MILP solver are utilized in the evolutionary algorithms, indicating a beneficial integration of information during the search solution process. In Step 3, the repairing algorithm developed by Fischetti e Lodi (2008) is employed, a hybrid algorithm that employs the feasibility pump approach to solve local branching at the beginning. The current MILP that is being solved is extended with artificial variables. This augmented model is then solved iteratively to reduce the number of infeasible solutions equaling by zero the values of the artificial variables. The goal is not to repair solutions only, but also infeasible MILP models. Although these two routines are time consuming and therefore should be used with parsimony, they effectively enhance the solution quality.

Step 2 of Algorithm 6 is implemented using the CPLEX parameter `PolishAfterIntSol`, while Step 3 is implemented using `Repair.Tries`. Time parameters t_1 and t_2 depend on the

Algorithm 6: CPLEX based algorithm (PC++)

1. Solve the current MILP model, using P-CPLEX until time limit ts_1. Create set SP with all feasible solutions found by the solver.

2. Solve the MILP using the polish routine by Rothberg (2007) with SP as initial solutions within time limit t_2. Set SP as the solution pool.

3. If $|SP| \geq 2$ then solve the MILP using the repair algorithm by Fischetti e Lodi (2008) with initial solutions in SP within time limit t_2. Keep SP as the solution pool.

4. Return the best feasible solution in SP. If no feasible solution is obtained, the problem cannot be solved considering parameters ts_1 and ts_2.

---

problem dimensions in terms of the number of products and ports.

3.3.2 Matheurisitc

A second algorithm was developed to solve the weighted max-min model of Step 6 in Algorithm 5 as a possible alternative to generate the Pareto front. The developed algorithm is a slightly modified version of the matheuristic presented in Santos et al. (2020). The algorithm has two interconnected phases. The first phase is inspired by a fix-and-relax solution strategy, and called relax-and-define step. In our algorithm, each set of integer variables are reduced into non-divisible blocks. Each block is defined by a strategy set vector (SB), which is a permutation of the set of integer variables. Only the variables in the first block are treated as integers in the first iteration, while the others are relaxed in the model. Due to the difficulty in finding a feasible solution to the current MILP, the repairing algorithm proposed by Fischetti e Lodi (2008) is used. For the entire horizon planning, the current sub-model is solved within the time limit t_s. The following variables in vector SB are defined as integers (but not fixed) and incorporated into the current sub-model at the next iteration of the algorithm, satisfying the position of the variable in vector SB. The solutions obtained in previous iterations are used as the starting point for the current sub-model. In the previous iteration, all variables are treated as integer in the current sub-model. The sub-model is solved to find the best solution. The number of iterations equals the cardinality of the set of integer variables. It should be noticed that the values of integer variables are not fixed in the interactions, and are instead included in a MILP solution pool.

After the first phase is finalized, an improvement phase is applied in the best solution obtained so far. The improvement phase uses Steps 2–4 of Algorithm 6, but with a time limit of t_s, following the limit used in each iteration step of the previous phase. Algorithm 7 outlines the developed routine.

Algorithm 7: Matheuristic algorithm

1. Relax and define. Let SP be a solution pool and IV the current set of integer variables. Set $SP \leftarrow \emptyset$ and $IV_s \leftarrow \emptyset$. For each $s = 1, ..., |SB|$ do

1.1. Define a variable in position s in vector SB as an integer. Insert the variable in set IV.

1.2. Consider the weighted max-min model (101) – (104), (53) – (132) with only the variables in set IV constrained as an integer. Relax the remaining integer variables.

1.3. Solve the MILP model in Step 1.2 using the repair algorithm by Fischetti e Lodi (2008) with initial solutions in SP within time limit t_s. Insert all feasible solutions in solution pool SP.

2. Improvement. Apply Steps 2 and 3 of Algorithm 6 within a time limit t_s for each step. Return the best feasible solution.

3.4 COMPUTATIONAL EXPERIMENTS

There are some measures that compare the performance of a designed algorithm with the true Pareto front. They are based on the distance between the obtained Pareto front by the algorithm and the true Pareto front. Unfortunately, these metrics can be applied only to test suite functions (VELDHUIZEN; LAMONT, 1998). These functions present special mathematical properties that allow the computational generation of the true Pareto front. Veldhuizen e Lamont (1998) presents several examples of these functions. Unfortunately, the true Pareto front of the MO-CRSP is unknown, and therefore there is no way of defining how far the Pareto solutions obtained by the developed algorithms from the true Pareto front. For this situation, we can only evaluate the performance of the two developed algorithms, comparing their performance using well-known convergence and diversity measures defined in Zitzler et al. (2000).

The objective of the computational experiments is to compare the effectiveness of the different developed algorithms to obtain the Pareto front. The algorithms were tested using sixteen real planning cases provided by a fertilizer company. The tested instances present different dimensions, in terms of the number of products to be transported, and pickup and delivery ports. The time windows of the deliveries and pickups were also included in the instance definition. Further, the company has made available vessel's options to transport the goods, based on its chartering database. The transportation costs were computed following the logistics department of the company. All computations were performed on an AMD Ryzen 5 2600 six-core computer, with 3.4 GHz and 16 GB RAM. Based on analysis and experimentation, big-M values were constrained to 1,000, strengthening the formulation. The trade-off frontiers were compared using the following performance metrics:

Set Coverage Metric ($\mathcal{C}$-metric): Introduced by Zitzler et al. (2000), it is a very common metric to compare two sets of non-dominated solutions (denoted as X and Y). The function C maps the ordered pair (X, Y) to the interval [0,1], as follows:

$$\mathcal{C}(X,Y) := \frac{|\{\mathbf{y} \in Y; \exists \mathbf{x} \in X : \mathbf{x} \succeq \mathbf{y}\}|}{|Y|}$$

The value $\mathcal{C}(X,Y) = 1$ means that all solutions in Y are dominated by or equal to solutions in X. If $\mathcal{C}(X,Y) = 0$, none of the solutions in Y are covered by the set X. Both $\mathcal{C}(X,Y)$ and $\mathcal{C}(Y,X)$ have to be considered since solutions in X and Y might not Pareto dominate each other.

Hypervolume (HV): It measures the size of the objective space covered by an approximation set (ZITZLER; THIELE, 1999), using a reference point to calculate the covered space. The reference point, in general, is defined from the given set having the values of its coordinates higher than the largest ones seen in the set. If an approximation set dominates another set, the HV of the former will be greater than the HV of the latter. This measure is widely accepted since it simultaneously considers accuracy, diversity, and cardinality.

Table 8 presents the values of the performance metrics found by PC++ and Mat to sixteen real-life instances. Each instance is defined in terms of the number of used origin ports (OP)-delivery ports (DP)-products to be transported (P). OP refers to the historical records of product offerings in the ports actually used by the plan. The companies did not have records of the offers of unused ports, reducing the complexity of the route planning. In the table, we compare the $\mathcal{C}$-metric and HV measures for each method using 21 solutions generated by different combinations of θ_1 θ_2, and θ_3, varying between 0 and 1 by 0.05. We also present the CPU time per weight combination required for each instance in seconds (CPU), following the experiments carried out in Santos et al. (2020). Both algorithms were run with the same total CPU time for all instances. For PC++, $t_1 = 2t_2$, while for the matheuristic, t_s =CPU/6. Also, vector $SB = [x_{ijv}, y_{ijpv}, y^D_{ijpv}]$ was used to solve the matheuristic.

Tabela 8 – Values of the performance metrics found by PC++ and Mat

		$\mathcal{C}$-metric		HV		
Instance	OP-DP-P	$\mathcal{C}$-metric(PC++,Mat)	$\mathcal{C}$-metric(Mat,PC++)	PC++	Mat	CPU
1	2-4-7	0.4286	0.524	2.500	2.4621	300
2	2-5-5	0.5238	0.4762	0.3974	0.3974	300
3	2-5-6	0.5238	0.5238	0.8532	0.7709	300
4	3-3-8	0.5714	0.2857	1.1980	1.1958	300
5	3-5-6	0.6190	0.5238	1.9790	1.9225	300
6	3-5-7	0.2857	0.3333	2.3583	2.3857	300
7	4-5-6	0.1905	0.3810	4.2101	4.6261	900
8	5-7-6	0.3333	0.1905	3.0501	2.8091	900
9	4-4-6	0.1429	0.1905	4.9758	4.4073	900
10	3-6-8	0.3333	0.4762	3.8189	2.8545	900
11	5-5-7	0.5714	0.5238	2.3504	1.5543	900
12	5-5-8	0.2857	0.1905	4.5490	4.6044	900
13	5-5-7	0.2857	0.0952	2.7310	2.7030	1800
14	5-7-5	0.1905	0.0952	3.7730	2.8060	1800
15	5-5-7	0.2381	0.2381	5.3109	5.3422	1800
16	5-7-5	0.1905	0.1905	4.3121	4.2367	1800
Average	–	0.3571	0.3273	2.9917	2.8173	–

The results obtained in Table 8 show that the algorithms present a similar behavior regarding the two performance metrics. It seems both algorithms obtained similar Pareto fronts. Regarding $\mathcal{C}$-metric, the algorithms tied in terms of the number of instances in which one exceeded (with a higher value of $\mathcal{C}$) the other. For two instances, they presented the same values of $\mathcal{C}$. Algorithm PC++ obtained better or equal HV values for ten instances, evidencing a slightly general better coverage of the solutions in the Pareto front than algorithm Mat. A justification for the behavior of both algorithms is the highly constrained nature of the problem, leading to a few feasible solutions. But, it is possible to note that PC++ presented, in general, slightly better values of both metrics for larger instances, with over five origin and destinations ports.

Figure 5 illustrates the convergence and diversity of the projected solutions obtained by PC++ (denoted as $\circ$) and Mat (denoted as $\times$) for Instance 7, a medium-sized one, at $Z_1 - Z_2, Z_1 - Z_3$, and $Z_2 - Z_3$ planes, considering 21 weight combinations. Given the number of constraints and the integrality of some variables, the number of different solutions was small. The same solution was shared by different weight combinations. Both methods could not find non-dominated solutions for all weight combinations, given both the complexity of the MILP and the CPU time used to solve the instances. The methods obtained very similar distribution and diversity of solutions, ratifying the results in Table 8. In summary, both algorithms are very competitive towards finding solution for Step 6 of Algorithm 5. However, considering the somewhat superior performance of the PC++ algorithm for the largest instances, we adopted this algorithm as the solution method to be used in the additional experiments of this study.

3.5 REAL APPLICATION

We consider the short-term planning of the cargo routing and scheduling problem from a large fertilizer company, involving the deep-sea transportation of up to 12 fertilizers stored in warehouses in North Africa and Europe to a maximum of 24 mixing units closely located to Brazilian destination ports. The plan is carried out by the logistics department. The company applies EXCEL to define the plan, using data from its enterprise resource planning (ERP). Due to the complexity of the plan, the process starts 90 days before its actual implementation. The units are responsible for estimating the monthly consumption of each raw material. Based on this information and the experience of the logistics department, the plan is elaborated and provided in the company's ERP for the raw material acquisition process and vessel chartering by the central administration located outside Brazil. This case study was dealt with in Santos et al. (2020), but considering a single-objective approach, the minimization of costs. However, during this research project, we realise this is only one of the several objectives used to elaborate a plan by the logistics department, motivating the development of this MOO.

In the real case study, some additional operational requirements arise to the more generic model introduced earlier. These requirements are specific to the analyzed company. They are described and formulated as follows. Due to coordination issues, up to 6 origin ports can be visited per plan, while the mix plants in Brazil can use up to 13 delivery ports per plan to receive

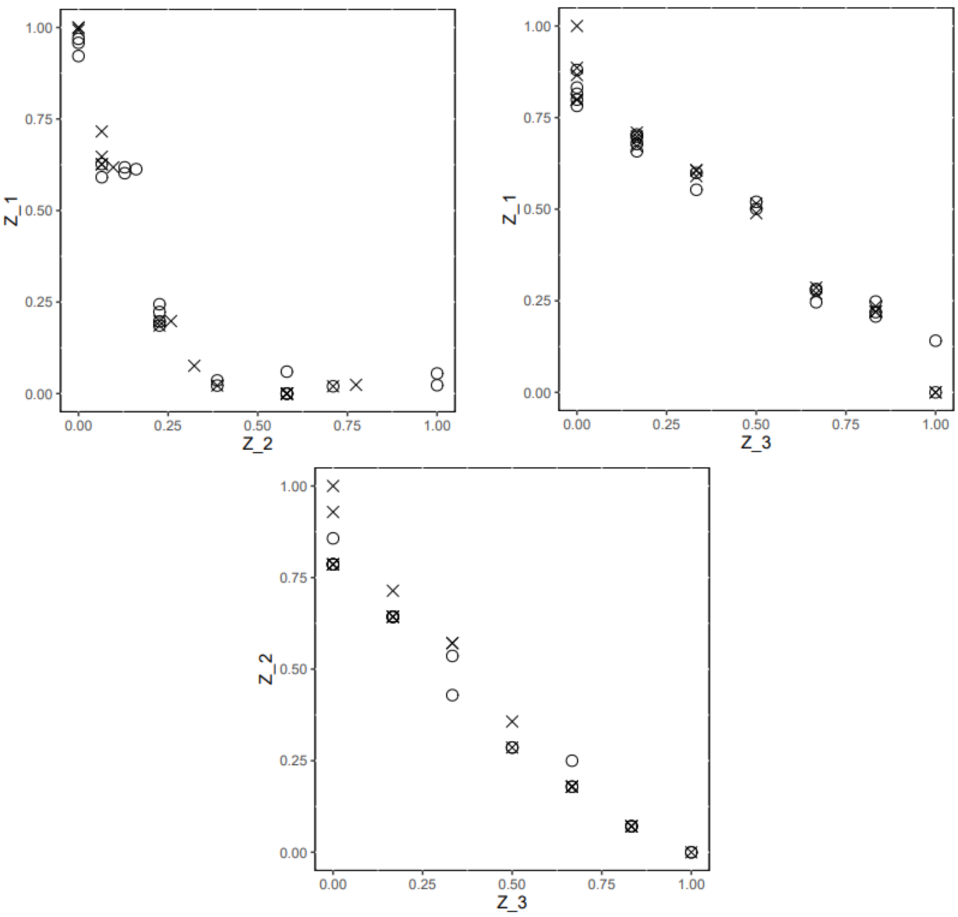


Figura 5 – Solutions obtained by PC++ (denoted as $\circ$) and the matheuristic (denoted as $\times$) on the selected instance 7 for (a) $Z_1 - Z_2$, (b) $Z_1 - Z_3$, and (c) $Z_2 - Z_3$ planes, respectively

their demanded raw materials. Although some destinations' ports may have several berths, only one berth can be used by the chartered vessels at any time, avoiding competition in using the port's resources by vessels serving the same route plan. To avoid a low usage of the vessels, there are restrictions on the minimum load of 20 kt per vessel, and the value of 4 kt per product transported by a vessel.

Further, feasible routes should comply with additional requisites and constraints. A vessel should visit, at most, two pickup ports and three delivery ports. Due to the large amounts transported by the company per year, the chartering is often negotiated with only two or three shipowners. These shipowners offer considerable discounts in the freight rates to secure the client. However, the COA with these shipowners includes an additional cost if there is a change in the geographical orientation in the delivery sequence of a vessel route. By adopting an orientation in the contract (in general, north to south), the company must pay a high fee each time the next port in a route sequence is located further north to the previous port. The fee is computed based on the vessel load when departing the latter port.

To model the problem, considering the operational peculiarities of this real case, it is necessary to define additional parameters as follows. Let W_v^V be the maximum number of pickup visits of vessel v, Y_v^V be the maximum number of delivery visits of vessel v, γ_v^C be the fee for geographical orientation change of loaded vessel v, and β^V_{ij} be a binary matrix that indicates a geographical orientation between ports $i, j \in N^D$. Further, binary decision variable a_{vw} replaces variable z_{vwj}, indicating if vessels v and w are simultaneously using the same port. The real case can be formulated as follows:

$$
\begin{aligned}
&(114), (52) && &&\\
&\min Z_2^{'} = Z_2 + \sum_{i \in N^V} \sum_{j \in N^V} \sum_{v \in V} \beta^V_{ij}\, \gamma^C_v\, M^V_v\, x_{ijv} && &&\\
&\text{st} && &&\\
&(114) - (150), (152) - (154), (127) - (158) && &&\\
&t^A_{ijv} + E^T_j - M(x_{ijv} - 1) + M a_{vw} \geq t^D_{kjw} && \forall i, k \in N, \forall j \in N^V, &&\\
&+ M(x_{kjw} - 1) && \forall v, w \in V, v \neq w && (109)\\
&t^A_{kjw} + E^T_j - M(x_{kjw} - 1) + M(1 - a_{vw}) \geq t^D_{ijv} && \forall i, k \in N, \forall j \in N^V, &&\\
&+ M(x_{ijv} - 1) && \forall v, w \in V, v \neq w && (110)\\
&\sum_{i \in N^P} \sum_{j \in N^P} x_{ijv} \leq W^V_v - 1 && \forall v \in V && (111)\\
&\sum_{i \in N^D} \sum_{j \in N^D} x_{ijv} \leq Y^V_v - 1 && \forall v \in V && (112)\\
&a_{vw} \in \{0, 1\} && \forall v, w | v \neq w \in V && (113)
\end{aligned}
$$

Objective function $Z_2^{'}$ refers to the minimization of total costs, adding the penalty for a change of orientation of vessel routes, as computed by the company. Constraints (109) and (110) ensure that two or more vessels cannot be at the same port simultaneously. Constraints (142) and (143) limit the number of pickup and delivery ports, respectively, that each vessel can visit in its route. The domain of the additional decision variable is defined by constraints (113).

The model was applied to a medium size instance involving deliveries of 8 raw-materials to fulfill the demand of mixing units located close to 5 delivery ports in Brazil. Due to the seasonality of the crops, the demand for a plan can be concentrated in only a few ports. Delivery port 3 is closely located to a very important customer. The pre-programmed delivery of demanded products in this port is an important strategic issue for the company, being represented in objective Z_3. The demands and supplies of products and the TWs at the corresponding ports were informed by the fertilizer company. A list of possible vessels, with information in dimensions, compartments and drafts, were also kindly provided. We briefly report the decision making process, using the optimization approach.

Table 9 presents the weight combinations (characterized as runs) and the corresponding values of the three objective functions. The runs were numbered in the same order as the weight combinations were interactively defined by the DMs. The optimization approach found non-dominant solutions for all combinations of weights, proving the robustness of the developed solution process. The first three runs aimed to find the best solutions of each objective function, separately. They are assumed as the best individual solutions and used as the minimum values for each objective. In the next six runs, the DMs wanted to better understand the trade-off magnitude between two of the three objectives. The solutions in Runs 1, 4, and 7 show that the trade-off between costs and makespan is manageable. The introduction of the change of direction cost seems to reduce the trade-off between costs and makespan, by significantly reducing the number of possible sequences in a vessel route, mainly ones with excessive makespan and low cost. However, considering the objective delay increased the complexity of the decision process, since the trade-offs between costs and delay, and makespan and delay are difficult to compromise (see Runs 3, 5, 6, 7, 8, and 10). The perception of the trade-off issue among the three objectives was consolidated with the results of Run 11, in which all objectives have the same weights. If we carefully analyze the solutions until Run 11, we can raise two important observations: (i) good solutions for the makespan can be obtained with high weights of objective cost (see Runs 1, 4, and 7); and (ii) the decrease by one in the objective delay results in a considerable increase in costs, but without considerable change in the makespan, as demonstrated comparing solutions of Runs 2–4, and Runs 8–11. So far, the DMs found solutions of Runs 6 and 8 as good ones, capable of obtaining reasonable costs and acceptable delays in the selected deliveries. The first one prioritizes delay, while the second prioritizes costs.

Tabela 9 – Results of the case study

	Weight			Solution		
Run	θ_1	θ_2	θ_3	Z_1	$Z_2^{'}$	Z_3
1	1	0	0	67	1702760	8
2	0	1	0	66	2301260	10
3	0	0	1	95	2968620	0
4	0.5	0.5	0	66	1882890	9
5	0	0.5	0.5	76	2795860	3
6	0.5	0	0.5	87	2769750	1
7	0.75	0.25	0	67	1702760	8
8	0.75	0	0.25	75	2336380	5
9	0.25	0.75	0	66	2301260	10
10	0.25	0	0.75	98	2838590	0
11	0.33	0.33	0.33	75	2764170	4
12	0.6	0.2	0.2	78	2719640	4
13	0.2	0.2	0.6	85	2944100	1
14	0.4	0.2	0.4	79	2588470	3
15	0.7	0.1	0.2	75	2336380	5
16	0.2	0.1	0.7	85	2944100	1
17	0.6	0.1	0.3	80	2534750	3
18	0.45	0.1	0.45	87	2769750	1

Based on the two previous conclusions, the DMs decided to find solutions considering all three objectives, but always assigning to objective makespan the smallest weights. Given the high number of constraints of the problem and the integrality of several variables, some solutions (see Runs 15, 16 and 18) were equal to previously obtained ones with similar combination of weights, limiting the range of possible adopted solutions. There was a consensus it could be counter-productive to consider additional weight combinations. From Runs 12–18, the solution of Run 17 stands out. Although similar to the solution of Run 14, Run 17 presents a better cost values, with the same delay value. The unitary worsening in the makespan was considered acceptable, given the decrease in the total costs. The solution of Run 17 has joined solutions from Runs 6 and 8 as the set of preferable solutions by the DMs. The solution of Run 17 offers a mid-term solution between the best cost solution of Run 8 and the best delay of Run 6, making it a good solution alternative.

We also compared the optimization approach with the logistics department solving process. Solutions of Run 6 and Run 17 reduced the total logistics costs, makespan, and delay by about 34%, 17%, and 90%, and 40%, 13%, and 70%, respectively, in comparison with the company's solution. On one hand, the improvement in the cost by the optimization approach was direct consequence of obtaining routes without any changes in the direction north-south by loaded vessels, while 75% of the routes presented a change of direction in the company's solution. On the other hand, the improvements in makespan and delays were consequence of more homogeneous vessel's routes in terms of the number of visited ports. The company's solving process is based on obtaining a viable solution in terms of all hard constraints. After a solution is obtained, an improvement step is applied where minimization of delays comes first, next costs, and finally makespan. In both stages, the staff faces difficulties in finding/improving the initial solution due to the complexity of simultaneously dealing with the operations requirement and the trade-off of the objectives. In general, the plans generated by the company disrespect soft constraints, such as the change of direction, in order to both respect the hard constraints, and to obtain reasonable values for the objectives. Due to confidentiality issues, solutions cannot be detailed in this paper.

A sensitivity analysis is performed on the best indicated solutions – Runs 6, 8, and 17, by decreasing the cost of inverting the direction of a loaded vessel during its route. Santos et al. (2020) experimentally determined this is a very influential term in defining different cost solutions. The majority of the remaining cost parameters are fixed by port and regulatory authorities. The DMs believe that a discount can be obtained in the applied penalty through a negotiation process with the more frequently used charter companies. Table 10 presents the new solutions, considering discounts of 10%, 15%, and 20% in the current value.

As we can note from Table 10, the total costs have decreased for all discounts considered, showing the relevance of this cost term. For a 10% discount, Runs 6, 8, and 17 reduced, on average, the total cost by 2.73%, 3.54%, and 8.38%, respectively. In terms of makespan, Runs 6 and 8 obtained the same number of days, while Run 17 was able to find a makespan with a reduction of 5% (around 4 days). Concerning delay, all solutions found the same values obtained in Table 9. The 15% discount presented a similar pattern, with the total cost being reduced by 6.59%, 5.80%, and 12.51% for Runs 6, 8, and 17, respectively, while the makespan and delay remained the same as obtained in the 10% category. Finally, the 20% discount also presented reductions in total cost by 9.01%, 6.84%, and 13.58%, respectively. Solution of Run 17 was the most impacted in terms of cost reduction while maintaining acceptable values for makespan and delay. If a 20% discount is obtained, the solution of Run 6 obtains zero delays with an acceptable value of cost, being recommendable in the case delay is the number one priority. Solution of Run 8 was discarded, given its similar cost results with Run 17, but with worse delay values. Solutions of Runs 6 and 17 were used by the DMs during the chartering process.

Overall, the company managers become very satisfied with the optimization method, being a scientific alternative to the current trial and error method based on Excel. There was a consensus among the managers that the developed optimization approach can capture the conflicting nature of decreasing costs and improving the service level to customers. The flexibility of the optimization approach, allowing the incorporation of specific peculiarities of different operations, and the efficiency and efficacy of the optimization approach of dealing with difficult strategic trade-offs were highly praised by the managers.

Tabela 10 – Results considering reduction on penalties of changing geographic orientation

Penalty		Weight			Solution		
Discount	Run	θ_1	θ_2	θ_3	Z_1	Z_2'	Z_3
10%	6	0.5	0	0.5	87	2694120	1
	8	0.75	0	0.25	75	2253630	5
	17	0.6	0.1	0.3	76	2322230	3
15%	6	0.5	0	0.5	87	2587160	1
	8	0.75	0	0.25	75	2200900	5
	17	0.6	0.1	0.3	75	2217520	3
20%	6	0.5	0	0.5	90	2520290	0
	8	0.75	0	0.25	75	2176520	5
	17	0.6	0.1	0.3	75	2190530	3

3.6 CONCLUSION

This paper introduces a multi-objective approach for the maritime CRSP, a variant of the multi-product, heterogeneous fleet pickup and delivery problem with time windows and draft limits. The problem is formulated as a MILP model in which the constraints represent the several real-life requirements of sea transportation. The solution of the problem specifies an operational plan, consisting of (i) the sequence of ports to be visited by each vessel; (ii) the amount of each product to be loaded/unloaded in a port by a vessel; and (iii) the arrivals and departures schedule of the vessels in/from ports. The overall plan should simultaneously minimize total transportation costs, scheduling makespan, and delays in some deliveries.

A weighted max-min fuzzy solution approach was developed to solve the multi-objective formulation. Considering the complexity of the MILP models generated during the solution process, two heuristics were developed based on several in-built cutting plan heuristics within contemporary MILP solvers. The first one uses developed polishing and repairing algorithms to improve and accelerate the solution search of the CPLEX solver. The second algorithm is a matheuristic that integrates the first developed algorithm with a modified relax-and-fix strategy. The former heuristic presented a slightly better behaviour towards obtaining the Pareto front in experiments using real-world instances from a Brazilian company. Finally, our approach was applied to a real case. Given the peculiarities of the real case, new constraints were added to the more generic formulation. From the findings of this real application, it is clear the flexibility and effectiveness of the optimization approach in handling diverse real-life problems.

Future research can be directed in two main directions. First, the proposed model can be improved by expanding its capabilities. Some of the deterministic assumptions could be relaxed, introducing stochastic travel times, loading/unloading times, and demand. Pollution objectives, such as CO_2 emissions, could be considered to reflect environmental costs. Second, evolutionary multi-objective optimization-based algorithms should be studied to improve the efficiency of the solution process.

4 THE SEGREGATED STORAGE MULTI-SHIP ROUTING AND SCHEDULING PROBLEM

ABSTRACT

This paper addresses the segregated storage multi-ship routing and scheduling problem with several products, heterogeneous fleet, time windows, split load, and draft limits in deep-sea transportation. The study is motivated by the real raw material supply problem of fertilizer companies operating in Brazil. First, bulk grain products are collected in European ports, then delivered to several Brazilian ones close to mixing plants. The main objective is to minimize the total logistics costs, respecting the several requisites of the problem, such as the capacity of ships' compartments, delivery delays, and change of direction in a ship's route in the Brazilian coast. Considering the complexity of the generated mixed integer linear programming models for real-life problems, a Lagrangian-based solution method was developed. Based on experimentation using 24 real-life instances, and a real case study, we attest to the effectiveness and relative efficiency of the optimization approach.

4.1 INTRODUCTION

This research work studies the ship routing and scheduling problem (SRSP) arising from the inbound logistics of fertilizer companies operating in Brazil. Brazil is the fourth-largest fertilizer importer in the world, accounting for around 8% of global fertilizer utilization (**??**). Most products come from Russia and Belarus. The logistic problem consists of transporting dry fertilizers by chartered ships. The products are first collected from ports in Europe and only then delivered to ports in Brazil, strategically located near mixing units.

According to the classification of Christiansen et al. (2013), the problem can be categorized as a cargo routing and scheduling problem (CRSP). The CRSP is a typical problem involving tramp shipping. This paper deals explicitly with a multi-commodity, heterogeneous multi-fleet CRSP with time windows, draft limits, split load, and segregated storage. Further, the problem involves several operational requisites concerning bulk grain transportation, such as: (i) respect the capacity of each ship compartment, (ii) the number of vessels that can be simultaneously served in a port, (iii) limits in the visited pickup and delivery ports in a route, and (iii) penalties for route inversion, from north to south or south to north, in the Brazilian coast. The main objective is to find ship routes and schedules with the lowest logistic costs, which include: (i) the quantity of products loaded/unloaded in each ship's compartment in each port; (iii) the routes of each ship; and (iv) the ship schedules in/from visited ports in its route.

In some previous studies (SANTOS et al., 2020; SANTOS; BORENSTEIN, 2022), the authors have formulated and solved the single and multiple objective CRSP, respectively. They considered the multi-product, heterogeneous fleet, time windows, dedicated compartments, and split load variant (m-CRSP-TW-DC-SL). The problem was formulated as a mixed-integer linear programming (MILP) model and solved by a matheuristic method. Although the matheuristic can efficiently solve real-world plans involving dozens of ports and products, the COVID-19 pandemic and the Ukraine war have significantly changed some requisites of the problem. Since the end of the pandemic peak and the beginning of the war, the increase in both the demand for bulk products and the oil prices have raised the shipping freight by more than 170% compared with pre-pandemic values[1]. Further, the war has inflicted severe supply chain disruptions, exacerbating the port congestion and crew crisis caused by the prolonged pandemic. The sanctions imposed on one side of the conflict resulted in severe problems in international freight with the disruption of trade with the belligerent countries and the loss of vessels[2]. Particularly, charter companies are facing unexpected challenges such as availability and cost of fuel and crew.

Before the conflict, companies carried out CRS planning before the chartering process to estimate costs, dates, and quantities to be collected and delivered in each port and quantities of ships to be contracted. However, with this new crisis scenario, the planning should be as detailed as possible, considering all the real features to guarantee the timing supply of raw materials at a reasonable cost in the mixing units. Premium products must reach customers at the proper use time on the plantation. Delays lead to essential productivity losses for the end customer and result in a painful customer loss for the fertilizer companies. The developed modeling approaches by Santos et al. (2020) and Santos e Borenstein (2022) obtain solutions that do not implicitly consider the following two important problem requisites: (i) segregated storage for partially loaded cargo in a ship compartment; and (ii) ship stability conditions concerning the load of cargo in multi-compartments. However, considering both requisites in the problem might significantly change the solution by an increase in the number of required ships to fulfill the demand. This paper expands the previous modeling approaches for the CRSP by simultaneously considering segregated storage and ship stability so that an optimization approach can be used both in the new international commerce scenario of our particular problem and in real-world problems with similar specifications. We refer to this problem as the heterogeneous CRSP with time windows, segregated storage, and split load (m-CRSP-TW-SS-SL) that integrates the tank allocation problem (TAP) (HVATTUM et al., 2009) to the maritime multi-ship CRSP. Considering all resulting requisites and constraints, this integration results in a complex problem to solve.

This study presents a Lagrangian relaxation (LR)-based solution method for the m-CRSP-TW-SS-SL. LR is widely used in solving hard integer programming problems (FISHER, 1981). The central concept is to decompose the problem into two different types of constraints: “hard” and “soft”. The hard constraints are incorporated into the objective function so that they are penalized by the corresponding Lagrangian multipliers. The resulting relaxed problem should be easier to solve, offering reasonable bounds for the original problem. The LR-based method is developed using an expanded formulation of the MILP presented in (SANTOS; BORENSTEIN, 2022), incorporating segregated storage and ship stability constraints. Experiments were

[1] https://unctad.org/news/war-ukraine-raises-global-shipping-costs-stifles-trade
[2] https://ifa-forwarding.net/blog/sea-freight-in-europe/impact-of-the-ukraine-conflict-on-maritime-shipping/

conducted in real-world instances, and in a case study in one of the largest fertilizer companies in Brazil. The results show that our Lagrangian approach effectively and efficiently solves the m-CRSP-TW-SS-SL.

This paper is structured as follows. Section 4.2 provides a concise literature review of previous related work. The problem is described in Section 4.3. Section 4.4 describes the mathematical notation and the extended MILP formulation. The LR-based method developed to solve the problem is described in detail in Section 4.5. Section 4.6 presents computational experiments to evaluate the performance of the solution method, while a real-life planning carried out by a fertilizer company is described in Section 4.7. An overview of the results and some directions for future research are given in the last section.

4.2 PREVIOUS RELATED WORK

The ship routing and scheduling problem and its variants have gained increasing attention since the seventies (RONEN, 1983; CHRISTIANSEN et al., 2013). The problem was initially studied as a single-vessel scheduling problem, where the routes were previously defined (APPELGREN, 1969; RONEN, 1993). Time windows (FAGERHOLT; CHRISTIANSEN, 2000b) and multi-ship routes (FAGERHOLT, 2001) were considered soon after, enlarging the real-world application of the developed optimization approaches. In natural development, real attributes were being increasingly incorporated, such as draft limits (MALAGUTI et al., 2018), flexible cargoes (KORSVIK; FAGERHOLT, 2010), and split loads (KORSVIK et al., 2011), substantially increasing the complexity of the mathematical formulation involved. As this study's main contribution concerns dealing with segregated storage in the CRSP, we focus our literature review on integrating cargo allocation into the CRSP.

The segregated storage CRSP is associated with the multi-compartment vehicle routing problem (McsRP) (OSTERMEIER et al., 2021). However, the deep-sea problem has characteristics that differentiate it from the usual truck-based McsRP, such as draft restrictions, cargo size flexibility, split loads, and long travel times. These attributes demand specific solution approaches for the maritime problem (OSTERMEIER et al., 2021). Segregated storage in marine transportation was introduced in Barbucha e Filipowicz (1997), where several simple optimization models were developed to maximize profit while satisfying the various requirements imposed by proper segregation. The problem was expanded and called tank allocation problem (TAP) by Hvattum et al. (2009). The authors introduced an embracing MILP formulation for the TAP and its variant, showing the computational intractability of the problem. Numerical experiments were conducted for single-ship instances using a commercial MILP solver. Vilhelmsen et al. (2017) developed a heuristic framework to solve the tactical TAP in a concise amount of time. The paper reports an apparent increase in the solution efficiency compared to the optimal method by Hvattum et al. (2009). However, all these works specifically dealt with cargo allocation, disregarding ship routes and the load/unload operations associated with cargo.

Al-Khayyal e Hwang (2007) is the first study to consider the integration of cargo allocation to an inventory routing and scheduling problem involving multi-commodity and heterogeneous fleet, following the developments of the McsRP (OSTERMEIER et al., 2021). However, no constraints related to segregation storage are included in the developed mathematical model. Fagerholt e Christiansen (2000a) introduce a combined ship scheduling and allocation problem (SSAP), in which vessels are equipped with flexible bulkheads that can easily increase or decrease the nominal capacity of the compartment. The problem is breakdown into two sequential and iterative processes. First, a list of candidate schedules is generated. Next, a subsequent set partitioning problem is applied to allocate the ships to the previously generated schedules. Fagerholt e Christiansen (2000b) detail how the generation of candidate schedule is solved as a traveling salesman problem with allocation, time windows, and precedence constraints, using a forward dynamic programming algorithm. However, ship stability and draft limits in ports are not considered. Kobayashi e Kubo (2010) consider a similar multi-vehicle pickup and delivery problem with time windows and allocation constraints. They decompose the problem into a TAP and a routing problem, both of which are solved as set partitioning problems. The main goal is to find feasible routes for the ships in the fleet to minimize the overall cost. However, cargoes may be rejected or carried out by spot charters, allowing the use of simple procedures to allocate cargo in the compartments for each ship.

Santos et al. (2020) and Santos e Borenstein (2022) consider dedicated compartments to the multiple commodity and heterogeneous fleet CRSP, as previously mentioned. However, a solution is either obtained by using a set of ships with compartment and capacity attributes to respect this requisite or by using the strategy of ships equipped with flexible bulkheads, following Fagerholt e Christiansen (2000a). Both requisites are guaranteed by constraints on the values of binary variables related to the products to be loaded in each compartment. However, choosing a proper set of ships depends on historical data of previous plans and the logistic staff's experience in deep-sea transportation of the involved products.

The above papers, focusing on combining the TAP and the ship routing and scheduling problem, do not consider explicit set partitioning constraints in their modeling approaches to guarantee segregated storage. Neo et al. (2006) is the forerunner of incorporating set partitioning and ship stability constraints into a previous routing and cargo allocation MILP developed by Jetlund e Karimi (2004). The model is solved using CPLEX for a case study involving a single parcel tanker with 10 compartments, 10 possible cargoes, and 5 pickup ports. Wang et al. (2018) and Ladage et al. (2021) expanded the work by Neo et al. (2006), incorporating draft requirements and cargo incompatibilities for tankers. In Ladage et al. (2021), cargo swaps between compartments are allowed. Both papers use heuristic methods to cope with the problem's complexity. Wang et al. (2018) decompose the problem into two phases: (i) the generation of feasible routes and (ii) the solution of a TAP problem for each route generated until a feasible solution is found. A neighborhood search-based heuristic is developed by Ladage et al. (2021). Although the ideas in these two articles are interesting, they restrict the modeling to a single ship. Furthermore, they do not allow the same product to be loaded in a ship compartment in different pickup ports.

Compared with previous work (SANTOS et al., 2020; WANG et al., 2018; LADAGE et al., 2021), the main contributions of our paper is to formulate and solve a CRSP variant that simultaneously considers heterogeneous fleet, multi-products, time-windows, segregated storage, split load, and draft limits. This is the first study to simultaneously consider all these real features in the deep-sea CRSP.

4.3 PROBLEM STATEMENT

Table 11 summarizes the problem structure, in terms of characteristics, assumptions, and constraints. The problem is encountered in the raw-material supply logistics of the largest fertilizer companies in Brazil, responsible for around 73% of the market share. The problem to be studied is the short-term CRS planning of bulk grain products. The plan aims at minimizing the transportation costs, determining: (i) the ships that will be used, (ii) each ship route and schedule; and (iii) the quantity of each product to be loaded/unloaded in/from ship's compartment at each port. Fig. 6 illustrates feasible ships' routes, schedules, and cargo allocation to compartments.

Tabela 11 – Problem characteristics and assumptions

Operations

- Bulk dry products
- Contracts of affreightment (COA)
- Many pickup and delivery ports
- No return cargo

Ships

- Heterogeneous fleet, with different dimensions, capacities, and number of compartments
- The draft of a ship is defined by its current load
- All ships, independent of their dimensions, travel at the same speed between ports

Ports

- Pickup and delivery ports may be visited by several ships
- Load and unload times at each port are previously known
- Draft limits are expressed by the maximum cargo which a ship can carry to enter or leave a port
- Ports can handle several products
- There are limits on the number of ships simultaneously berthed in a port
- Time windows $[A_j^T, L_j^T]$ are imposed for the pickup/delivery of products in port j

Ship Routes

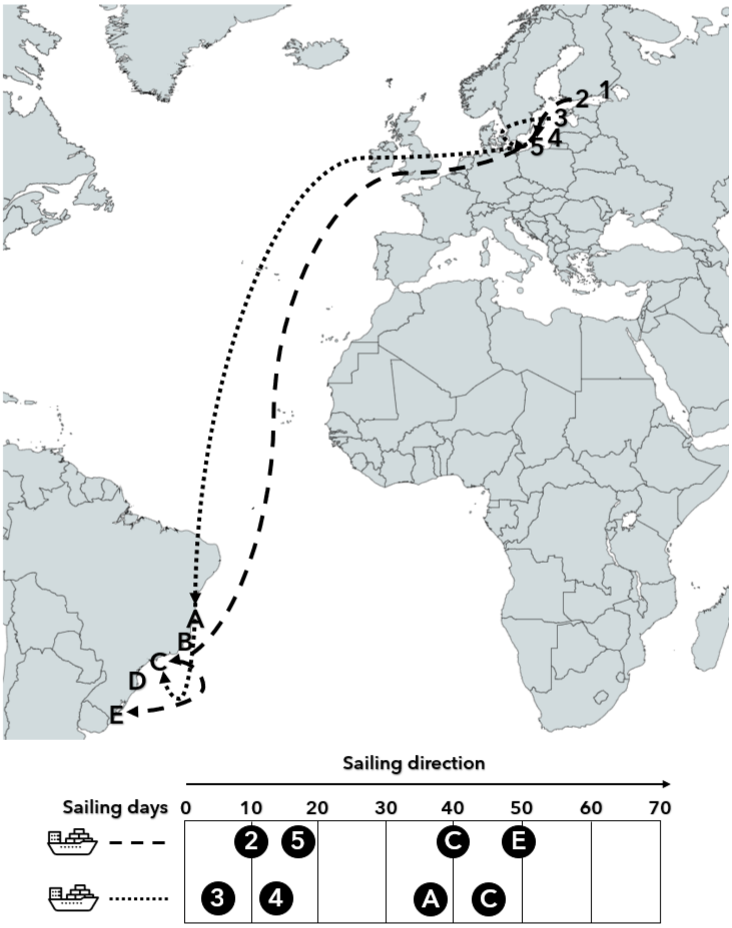
- Sequences where a pickup port always appears before any delivery port
- Ships only visit a port once
- Limits on the number of ports for a specific route

Shipment

- Flexible cargo sizes
- Split load
- Segregated storage
- No transshipment allowed

Costs

- Chartering
- Demurrage
- Use of port facilities
- Inversion of the geographical orientation north-south or south-north by a ship during delivery
- Fee for non-compliance with time windows in ports



Sailing direction

Comp.	Port 2	Products	Port 5	Products	Port C	Products	Port E	Products
1	+5500	1		1	-5500	1		
2	+2000	2	+5000	2		2	-7000	2
3	+12000	4		4	-12000	4		
4			+5000	5	-3000	5	-2000	5
5			+10000	6		6	-10000	6

Sailing direction

Comp.	Port 3	Products	Port 4	Products	Port A	Products	Port C	Products
1	+4000	1	+2500	1	-6500	1		
2	+8000	2		2	-5000	2	-3000	2
3	+6000	4		4		4	-6000	4
4	+7500	5		5	-3000	5	-4500	5
5	+12000	6		6	-6000	6	-6000	6

Figura 6 – Example of ships' routes, schedules, and cargo allocation

4.4 PROBLEM FORMULATION

We used the underlying network and formulation by Santos e Borenstein (2022) as a basis to develop an extended MILP formulation, in which modifications to account for segregated storage allocation of products to ships' compartments and ship stability were incorporated. The mathematical notation following (SANTOS; BORENSTEIN, 2022) is presented below. Sets and parameters are expressed in upper case letters, while lower case letters are used for variables and indexes.

Sets	
N^V	set of pickup and delivery ports
N^P	set of pickup ports
N^D	set of delivery ports
N	set of all ports, including the artificial start (s) and finish (f) ports
P	set of products
V	set of ships
V^*	set of ships used in a solution of the problem
P^*	set of products transported in a solution of the problem
$C(v)$	set of compartments of ship v
Parameters	
Q_{jp}	quantity of product p in port j
D^T_{jp}	arrival due time of product p in port j
O^T_j	estimated time for loading/unloading in port j
E^T_j	estimated waiting times in port j
L^T_{ij}	travel distance from port i to port j
P^N_j	draft of port j in ton
B^N_j	berths in port j
U^N_j	limit of ships simultaneously berthed in port j
M^s_s	nominal capacity of ship s
W^C_s	demurrage rate of ship s
P^C_j	port j facility costs
C^S_s	daily freight cost of ship s
H^C	fee for not respecting the loading/unloading time windows of a port
Y^S_{cs}	capacity of compartment c of ship s
R^S_{cs}	longitudinal distance of compartment c from ship s's centre of flotation
K^S_{cs}	lateral distance from compartment c from ship s's centre of flotation
W^S_s	allowed number of pickup visits of ship s
Y^S_s	allowed number of delivery visits of ship s
α	maximum absolute permissible moments causing trim of a ship
β	maximum absolute permissible moments causing heel of a ship
γ_s	fee for route inversion of loaded ship s in Brazilian coast
θ_{ij}	binary matrix that specifies a geographical orientation between ports i, j
$\mathcal{M}$	very large number
Decision variables	
l_{pijcs}	quantity of product p transported within compartment c of ship s using arc (i, j)
t^A_{ijs}	time of arrival of ship s at port j, coming from port i
t^D_{ijs}	time departure of ship s from port j, coming from port i
t^E_{pijs}	early arrival or delay of ship s carrying product p using arc (i, j)
t^L_{pijs}	arrival delay of ship s at port j carrying product p using arc (i, j)
x_{ijs}	binary variables indicating if arc (i, j) is used by ship s
y_{pijcs}	binary variables indicating if product p transported by ship s uses arc (i, j)
y^D_{pijs}	binary variables indicating if product p is unloaded of ship s in port j
g_{pcs}	binary variables indicating if product p is loaded in compartment c of ship s
z_{swj}	binary variables indicating if ships s and w can be simultaneously berthed in port j
s_{ps}	quantity of product p that cannot be transported in ship s due to segregated storage

The problem can be represented on a directed graph $\mathcal{G} = (N, A)$, where N is the set of nodes, and $A = \{(i, j) | (i \in N^D, j \in N^D) \wedge (i \in N^P, j \in N^s) \wedge (i \in N^D, j \in \{f\}) \wedge (i \in \{e\}, j \in N^P)\}$ is the set of arcs. The graph has no arcs between delivery and pickup ports, respecting the routing structure in Table 11. An elementary path from e to f is a feasible route for a ship if it respects all flow, shipload, time, port, and routing constraints of the problem.

Based on the formulation presented in Santos e Borenstein (2022), the mathematical formulation of the m-CRSP-TW-SS-PL is presented below:

$$\min \sum_{s\in S} C_s^S \left(\sum_{i\in N^D} t_{ifv}^A - \sum_{j\in N^P} t_{ejs}^A \right) + \sum_{i\in N} \sum_{j\in N^s} \sum_{s\in S} P_j^C x_{ijs} \tag{114}$$
$$+ \sum_{p\in P} \sum_{i\in N^S} \sum_{j\in N^D} \sum_{s\in S} H^C t_{ijps}^E + \sum_{i\in N} \sum_{j\in N^S} \sum_{s\in S} W_s^C E_j^T x_{ijs}$$
$$+ \sum_{i\in N^D} \sum_{j\in N^D} \sum_{s\in S} \theta_{ij}^s \gamma_s M_s^S x_{ijs}$$

subject to

$$Q_{ip} + \sum_{h\in N^s} \sum_{s\in S} \sum_{c\in C(s)} l_{phics} - \sum_{j\in N^S} \sum_{s\in S} \sum_{c\in C(s)} l_{pijcs} \geq 0 \qquad \forall p\in P, \forall i \in N^s, \tag{115}$$

$$Q_{ip} + \sum_{h\in N^P} l_{phics} - \sum_{j\in N^S} l_{pijcs} \geq 0 \qquad \forall p\in P, \forall i\in N^P, \forall s\in S, \forall c\in C(s) \tag{116}$$

$$\sum_{h\in N^S} l_{phics} - \sum_{j\in N^D} l_{pijcs} \geq 0 \qquad \forall p\in P, \forall i\in N^D, \forall s\in S, \forall c\in C(s) \tag{117}$$

$$\sum_{i\in N^P\cup\{e\}} l_{pijcs} - \sum_{k\in N^S} l_{pjkcs} \leq 0 \qquad \forall p\in P, \forall j\in N^P, \forall s\in S, \forall c\in C(s) \tag{118}$$

$$\sum_{p\in P} \sum_{i\in N^P} \sum_{j\in N^S} \sum_{c\in C(s)} l_{pijcs} \leq M_s^S \qquad \forall s\in S \tag{119}$$

$$l_{pijcs} \leq Y_{cs}^S g_{pcs} \qquad \forall p\in P, \forall i\in N^P, j\in N^S, \forall s\in S, \forall c\in C(s) \tag{120}$$

$$\sum_{p\in P} g_{pcs} \leq 1 \qquad \forall s\in S, \forall c\in C(s) \tag{121}$$

$$-\alpha \leq \sum_{p\in P} \sum_{i\in N^S} \sum_{j\in N^S} R_{cs}^S l_{pijcs} \leq \alpha \qquad \forall s\in S, \forall c\in C(s) \tag{122}$$

$$-\beta \leq \sum_{p\in P} \sum_{i\in N^s} \sum_{j\in N^S} K_{cs}^S l_{pijcs} \leq \beta \qquad \forall s\in S, \forall c\in C(s) \tag{123}$$

$$y_{pijs} - \sum_{c\in C(s)} g_{pcs} \leq 0 \qquad \forall p\in P, \forall i\in N^S, \forall j\in N^D, \forall s\in S \tag{124}$$

$$\sum_{s\in S} \sum_{w\in S|w\neq s} z_{swj} \leq B_j^N \qquad \forall j\in N^s \tag{125}$$

$$\sum_{s\in S} \sum_{w\in S|w\neq s} z_{swj} \leq U_j^N \qquad \forall j\in N^s \tag{126}$$

$$\sum_{p\in P} \sum_{j\in N} \sum_{c\in C(s)} l_{pijcs} \leq P_i^N \qquad \forall i\in N^S,\ \forall s\in S \tag{127}$$

$$\sum_{p\in P} \sum_{i\in N} \sum_{c\in C(s)} l_{pijcs} \leq P_j^N \qquad \forall j\in N^s,\ \forall s\in S \tag{128}$$

$$y_{pijs} - x_{ijs} \leq 0 \qquad \forall p\in P, \forall i,j\in N, \forall s\in S \tag{129}$$

$$t^D_{ijs} - \mathcal{M}x_{ijs} \leq 0 \qquad \forall i,j \in N,\ \forall s \in S \qquad (130)$$

$$t^A_{ijs} - \mathcal{M}x_{ijs} \leq 0 \qquad \forall i,j \in N,\ \forall s \in S \qquad (131)$$

$$y_{pijs} - D^T_{jp}y^D_{pijs} \leq 0 \qquad \forall p \in P, \forall i \in N^S,\ \forall j \in N^D, \forall s \in S \qquad (132)$$

$$\sum_{j \in N} x_{ijs} \leq 1 \qquad \forall i \in N,\ \forall s \in S \qquad (133)$$

$$\sum_{i \in N} x_{ijs} \leq 1 \qquad \forall j \in N, \forall s \in S \qquad (134)$$

$$\sum_{j \in N} x_{ijs} - \sum_{h \in N} x_{hiv} = 0 \qquad \forall i \in N,\ \forall s \in S \qquad (135)$$

$$\sum_{i \in N^P} \sum_{j \in N^D} x_{ijs} \leq 1 \qquad \forall s \in S \qquad (136)$$

$$x_{ijs} = 0 \qquad \forall i \in N^D,\ \forall j \in N^P,\ \forall s \in S \qquad (137)$$

$$x_{fes} = 0 \qquad \forall s \in S \qquad (138)$$

$$x_{fjs} = 0 \qquad \forall j \in N^S,\ \forall s \in S \qquad (139)$$

$$x_{ies} = 0 \qquad \forall j \in N^S,\ \forall s \in S \qquad (140)$$

$$\sum_{p \in P} l_{pijcs} - Y^S_{cs}x_{ijs} \leq 0 \qquad \forall i,j \in N, \forall s \in S, \forall c \in C(s) \qquad (141)$$

$$\sum_{i \in N^P} \sum_{j \in N^P} x_{ijs} \leq W^S_s - 1 \qquad \forall s \in S \qquad (142)$$

$$\sum_{i \in N^D} \sum_{j \in N^D} x_{ijs} \leq Y^S_s - 1 \qquad \forall s \in S \qquad (143)$$

$$t^A_{ijs} \geq A^T_j x_{ijs} \qquad \forall i \in N, \forall j \in N^s,\ \forall s \in S \qquad (144)$$

$$t^D_{ijs} \leq L^T_j x_{ijs} \qquad \forall i \in N, \forall j \in N^s,\ \forall s \in S \qquad (145)$$

$$t^D_{ijs} \geq t^A_{ijs} + E^T_j + O^T_j + \mathcal{M}(x_{ijs} - 1) \qquad \forall i \in N,\ \forall j \in N^S . \forall s \in S \qquad (146)$$

$$t^A_{ijs} \geq t^D_{hiv} + L^T_{ij} + \mathcal{M}(x_{ijs} - 1) \qquad \forall h \in N,\ \forall i \in N^S,\ \forall j \in N^S,\ \forall s \in S \qquad (147)$$

$$t^E_{pijs} \geq \mid D^T_{jp} - E^T_j - t^D_{ijs} \mid + \mathcal{M}(y^D_{pijs} - 1) \qquad \forall i \in N^S,\ \forall j \in N^D,\ \forall p \in P,\ \forall s \in S \qquad (148)$$

$$t^A_{ifs} \geq t^D_{his} + \mathcal{M}(x_{ifs} - 1) \qquad \forall i,h \in N^S,\ \forall s \in S \qquad (149)$$

$$t^A_{eis} \leq t^A_{ijs} - A^T_j - O^T_i - E^T_i - \mathcal{M}(x_{ijs} - 1) \qquad \forall i,j \in N^S,\ \forall p \in P,\ \forall s \in S \qquad (150)$$

$$t^A_{ijs} + E^T_j - \mathcal{M}(x_{ijs} - 1) + Mz_{swj} \geq t^D_{kjw} + \mathcal{M}(x_{kjw} - 1) \qquad \forall i,k \in N,\ \forall j \in N^S,\ \forall s,w \in S, s \neq w \qquad (151)$$

$$t^D_{ijs} \leq D^T_{jp} + L^T_j - E^T_j + \mathcal{M}(-y^D_{pijs} + 1) \qquad \forall i \in N^s,\ \forall j \in N^D,\ \forall p \in P,\ \forall s \in S \qquad (152)$$

$$t^D_{ijs} \geq D^T_{jp} - A^T_j - E^T_j - \mathcal{M}(y^D_{pijs} - 1) \qquad \forall i \in N^s,$$

$$t^L_{pijs} \geq t^A_{ijs} + E^T_j - D^T_{jp} + \mathcal{M}(y^D_{pijs} - 1) \qquad \forall p \in P, \forall i, j \in N^S, \forall s \in S \tag{154}$$

$$l_{pijcs}, t^D_{ijs}, t^A_{ijs}, t^E_{ijps} \geq 0 \qquad \forall p \in P, \forall i, j \in N, \forall s \in S, \forall c \in C(s) \tag{155}$$

$$x_{ijs}, y_{pijs}, y^D_{pijs} \in \{0, 1\} \qquad \forall p \in P, \forall i, j \in N, \forall s \in S \tag{156}$$

$$g_{pcs} \in \{0, 1\} \qquad \forall p \in P, \ \forall s \in S, \forall c \in C(s) \tag{157}$$

$$z_{swj} \in \{0, 1\} \qquad \forall s, w | s \neq w \in S, \forall j \in N^s \tag{158}$$

The objective function (114) minimizes the total transportation costs, including freight costs, costs of using port facilities, penalties for disrespecting the time window of each port, ships' demurrage rates, and penalties for route inversion on the Brazilian coast. Constraints (115) guarantee that the quantities loaded/unloaded of product p in ship s are compatible with the supply/demand of port j. Constraints (116) assure that a product can only be loaded in a ship if a corresponding stock of this product exist in a pickup port. Constraints (117) satisfy product demand at delivery ports. Constraints (118) prevent transshipment in a pickup port. Constraints (119) and (120) guarantee that the nominal capacity of ships and compartments are satisfied, respectively.

Constraint set (121) guarantees segregated ship compartment storage. Constraints (122) and (123) assure that the moments causing trim and heel, respectively, of a ship, have to be less than their respective limits to ensure stability (NEO et al., 2006). Constraint set (124) guarantees that a product is transported in ship s using arc (i, j) if loaded in at least one of the ship's compartments.

Constraints (125) respect the berth capacity of port j. Constraint set (126) limits the number of ships simultaneously served in port j. Constraints (127) and (128) are port draft-related constraints. Constraints (129) establish connections between a ship's loading and its route, while constraints (130) and (131) establish connections between the times of a ship and its route. Constraint set (132) connects the cargo allocated to a ship when arriving to a port with the unloading of a product in the port.

Constraints (133) and (134) guarantee that a ship only uses once an arc in the graph. Constraints (135) are flow conservative ones. Constraints (136) and (137) ensure that products are first collected and only then delivered in any route. Constraint sets (138), (139) and (140) define nodes e and f as super source and super sink nodes. Constraint set (141) assures that the nominal capacity of ships' compartments are respected. Constraints (142)–(143) limit the number of pickup and delivery ports to be visited by a ship in its route.

Constraints (144) and (145) connect sailing times of ship s to the time windows of port j. Constraints (146) and (147) compute the departure and arrival times of ship s from/in port j, respectively. Constraint set (148) computes the deviation from the expected delivery time of product p by ship s in port j. Constraints (149) and (150) compute the starting and ending times of a ship's routes, respectively. Constraint set (151) determines if ship w is already being served in port j when ship s arrives at this port. The departure time from port i to port j by ship s is computed by constraints (152) and (153). Constraints (154) determine the late delivery of product p in port j. Constraints (155)–(158) define the domain of the variables.

4.5 SOLUTION METHOD

Commercial MILP solvers are restricted to tiny instances for the m-CRSP-TW-DL-PL (SANTOS et al., 2020). As our problem is an extension of this CRSP variant, we developed a heuristic approach for solving model (114)–(158). Several heuristic methods were employed to solve the SRSP in different environments and decision levels, such as cutting plane algorithm (MALAGUTI et al., 2018), column generation (COCCOLA et al., 2015), multi-start heuristic (YAMASHITA et al., 2019), metaheuristics (KORSVIK; FAGERHOLT, 2010; TROTTIER; CORDEAU, 2019), and Lagrangian relaxation (SHEN et al., 2011). We recommend Christiansen et al. (2013) for a detailed review of solution methods applied to the SRSP and its variants.

It should be noted that the above MILP formulation without constraints (121)–(123) becomes a similar problem addressed by Santos e Borenstein (2022). The Lagrangian relaxation is a natural strategy as the model can be efficiently solved without the complicated dedicated compartment constraints (121) (SANTOS et al., 2020). We use multipliers λ_{cs} for each compartment c in ship s to relax constraints (121). The LR problem is formulated as follows:

$$LR(\lambda) = \min Z + \sum_{s \in S} \sum_{c \in C(s)} \lambda_{cs} (\sum_{p \in P} g_{pcs} - 1) \tag{159}$$

$$\text{st} \tag{160}$$

$$(122) - (124), (115) - (119), (133) - (158) \tag{161}$$

$$\sum_{p \in P} \sum_{i \in N^P} \sum_{j \in N^S} l_{pijcs} \leq Y^S_{cs} \qquad \forall s \in S, \forall c \in C(s) \tag{162}$$

Without constraints (121), the constraint set (120) allows the capacity of compartments to be disrespected and replaced by constraints (162). Using constraints (162), different products can be transported simultaneously in one compartment while maintaining the nominal capacity of each compartment.

Algorithm 8 summarizes the overall Lagrangian relaxation method developed.

Algorithm 8: Solution approach

Step 1 (Lagrangian relaxation): Use Lagrangian relaxation on model (114)–(158), relaxing constraints (121). Solve the Lagrangian subproblem $LR(\lambda)$, using the relax and define algorithm introduced in (SANTOS et al., 2020).

Step 2 (Primal Solution): Based on the solution of Step 1, apply the heuristic described in Algorithm 9 to obtain a solution for the primal problem.

Step 3 (Subgradient optimization): Update the Lagrangian multipliers using a subgradient search.

Step 4 (Halt Condition): Repeat Steps 1—3 until a solution with a required accuracy is achieved or a prescribed time-limit is surpassed.

4.5.1 Solving the Lagrangian Dual

The Lagrangian dual $L(\lambda) = \max_{\lambda \geq 0} LR(\lambda)$ is solved by the subgradient optimization introduced by (HELD; KARP, 1971). To each Lagrange multiplier λ_{cs}, a subgradient vector associated is defined by: $v_{cs} = 1 - \sum_{p \in P} g_{pcs}, s \in S, c \in C(s)$. A sequence $\lambda^0, \lambda^1, \ldots$ of Lagrangian multiplier vectors are generated during the subgradient optimization. The multipliers are updated, as follows: $\lambda_{cs}^{k+1} : k \geq 1 = \max[0; \lambda_{cs}^k + \frac{\alpha^k (UB - L(\lambda^k)}{||v_{cs}(\lambda^k)||^2}], s \in S, c \in C(s)$, where UB is an upper bound in the primal problem, $LR(\lambda^k)$ is the lower bound obtained by solving the dual Lagrangian problem for the multiplier λ^k, and $0 \geq \alpha^k \leq 2$ is a given step-size parameter. Following Fisher (1981), α^0 is set to 2, being halved whenever the upper bound UB has not decreased in some fixed number of iterations.

Each model $LR(\lambda^k)$ is solved by the relax-and-define (RaD) algorithm proposed by (SANTOS et al., 2020). In this algorithm, the set of binary variables is partitioned into blocks. Only variables in the first block are considered binary in the first iteration. The resulting MILP is solved for the whole horizon planning. At the next iterations of the algorithm, the variable in the next block is defined as binary in the MILP of the previous step. The resulting MILP is again solved for the whole horizon planning. The process is repeated until all variables are defined as binary. Each iteration uses the best solutions of previous ones as initial solutions for the current MILP. At each iteration, we incorporate the polishing algorithm developed by Rothberg (2007) in the commercial MILP solver towards improving the solution process. Next, the repairing algorithm developed by Fischetti e Lodi (2008) is applied to the pool of solutions obtained considering all variables as binary. The best solution obtained in this repairing step is returned.

4.5.2 Primal heuristic

Obtaining a primal solution for our problem is challenging due to the several constraints in the problem, especially when the constraint set (121) is considered. However, the solutions of the dual Lagrangian problems can be used to find a suitable primal solution. The solution of the Lagrangian relaxation allows that a ship compartments can be loaded with more than one product. To obtain a feasible primal solution, we need to fix this unwanted situation, if it occurs for any ship.

To solve the primal feasible solutions, we employ a strategy to introduce new ships to the current Lagrangian dual solution to comply with the segregated storage rather than trying to adjust the cargo of already used ones, and only then consider the utilization of new ships. The strategy is justified by the difficulty of finding feasible solutions adjusting ships' capacity, considering the interconnection of the several constraints of the original problem. Therefore, the primal heuristic is based on the following two interconnected phases, as follows:

Phase 1: In this phase, we guarantee that all utilized ships in the dual solution obey constraint set (121), solving a segregated storage problem per ship;

Phase 2: In the second phase, we serve the uncovered demands resulting from the first phase by using extra ships.

The first phase is solved using the segregated storage model in Evans e Tsubakitani (1993) for each effectively used ship $s \in S^*$, and product quantities l^*_{ijpcs} transported in the current dual solution, indicated by the use of the superscript *. Two additional decision variables need to be defined, u_{pcs} as the quantity of product $p \in P^*$ to be transported in compartment c of ship $s \in S^*$, and s_{ps} as the amount of product $p \in P^*$ that cannot be transported in ship $s \in S^*$, due to the segregated storage constraints (121). The segregated storage problem for each $s \in S^*$ can be formulated as follows:

$$\max \sum_{p \in P^*} \sum_{c \in C(s)} u_{pcs} \tag{163}$$

st

$$\sum_{c \in C(s)} u_{pcs} + s_{ps} = L_{ps} \qquad \forall p \in P^* \tag{164}$$

$$\sum u_{ijpcs} - Y^s_{cs} g_{pcs} \leq 0 \qquad \forall p \in P^*, \forall c \in C(s) \tag{165}$$

$$\sum_{p \in P^*} g_{pcs} \leq 1 \qquad \forall c \in C(s) \tag{166}$$

$$u_{pcs}, s_{ps} \geq 0 \qquad \forall p \in P^*, \forall c \in C(s) \tag{167}$$

where $L_{ps} = \sum_{i \in N^P} \sum_{j \in N^P} l^*_{pijcs}, p \in P^*, c \in C(s)$. The objective is to load the maximum amount of products $p \in P^*$ in ship $s \in S^*$ without changing its route and schedule. Constraints (164) ensure that the availabilities of products from the dual solution are respected. Constraints (165) respect the capacity of each compartment of ship s. The solution of the segregated storage problem guarantees that ship $s \in S^*$ carries quantities u_{pcs} of a single product p in each of its compartment $c \in C(s)$, respecting constraint set (121).

The problem now is how to allocate the amounts $s_{ps} > 0, p \in P^*, s \in S^*$ no longer delivered since they cannot be transported in ship $s \in S^*$, due to constraints (121). This problem is addressed in the second step. Instead of trying to develop a specialized algorithm for solving the problem, we used an intuitive idea of assigning the amounts $s_{ps} > 0, p \in P^*$ of ship s to a new ship s', with the same route used by ship s to collect and delivery products p. In order to obtain the primal solution, we need to incorporate the transportation costs related to new ships into the solution of the current LR problem. The complete process is outlined in Algorithm 9.

Due to time limitations to solve the problem, we designed two different methods to define the route and schedule of ship s' in Step 2.4. One efficient way is to copy the ship route s, adjusting the schedule to respect all constraints of the problem. An alternative is to consider the same ports in the route of ship s, but focusing only on the supply/demand of products p'. The schedule is then adjusted in the same way as in the previous methods. Although the latter method offers lower upper bounds, both yielded the same results and number of iterations when applied

Algorithm 9: Primal heuristic

Step 1: Consider the solution of the current Lagrangian dual problem, in terms of products transported, l^*_{pijcs}, routes and schedules of ships $s \in S^*$. Set variable UB as the objective function of the current $LR(\lambda^k)$.

Step 2: For each ship $s \in S^*$ do

Step 2.1: Solve the segregated problem (163)–(167) considering product quantities transported in the dual solution, l^*_{pijcs}.

Step 2.2: If all cargo assigned to ship s respect the shipload constraints (e.g., $s_{ps} = 0, \forall p \in P^*$, go to Step 2. Otherwise, go to Step 2.3.

Step 2.3: Define set $P' = \{p \in P^* | s_{ps} > 0\}$.

Step 2.3: Segregate storage cargo $s_{ps} > 0, \forall p \in P'$ in a new ship s' with capacity $M^s_{v'} \geq \sum_{p \in P'} s_{ps}$ and number of compartments $|C(s')| \geq |P'|$ by solving a segregated storage problem.

Step 2.4: Define the route and schedule of ship s' based on the route and schedule of ship s.

Step 2.5: Compute the transportation costs associated with ship $s', TC_{s'}$, using the formula in Equation (114). Set $UB \leftarrow UB + TC_{s'}$

Step 3: Return a primal solution with the value of UB.

to real and randomly generated instances. For efficiency's sake, the first method was adopted. Nevertheless, we are developing a heuristic method towards improving the upper bounds to improve the solution search.

It should be noted that the primal heuristic is quite efficient since the number of segregated storage problems, in general, decrease as the Lagrangian multipliers are updated, using both the LR solution (as lower bound) and the primal solution (as upper bound). However, the convergence of the whole solution process can be slow, especially for large problems, in terms of products and ports. The bounds of the primal heuristic can be quite high in the first iterations of Algorithm 8. The focus of our primal heuristic is on the feasibility of the solution toward updating the Lagrangian multipliers.

4.6 NUMERICAL EXPERIMENTS

This section describes the experiments carried out to evaluate the optimization approach. The algorithms were coded in C++, using Visual Studio 2022 compiler. To solve the MILP formulations within the optimization approach, we used CPLEX 12.8 with several cut-related parameters active, which values are listed in Santos et al. (2020). We refer to this parameterized variant as P-CPLEX. The experiments were carried out on a six-core AMD Ryzen 5 2600 3.4 GHz CPU and 16 GB RAM. A fertilizer company kindly provided real-life instances from 2013–2019. The instance data include the logistics costs, the set of available ships, and the supply and demand quantities of products in the pickup and delivery ports, respectively. Furthermore, the company informed the final chartering costs. Table 12 presents each instance's characteristics by the number of pickup (PP), delivery ports (DP), products in demand (P), and the number of ships in set s ($|V|$). The table also presents the number of constraints and variables of model (114)–(158) of each instance, offering an idea of the dimensions of each problem. However, it should be noted that the complexity of the model is not only a function of the number of variables and constraints but also of the attributes of ships in set s and the ports index set N^s, as well the prescribed time windows and expected time of delivery of products to ports. Several different ships were considered, with a median size of 35,000 DWT and 4 to 5 compartments. Product demand ranges from 80,000t to 200,000t.

Tabela 12 – Instances' characteristics

					Model (114)–(158)		
Instance	OP	DP	P	$\|V\|$	Total Variables	Binary Variables	Constraints
1	2	4	7	4	7853	2525	44986
2	2	5	5	3	7341	2049	36159
3	2	5	6	3	8681	2409	40957
4	2	5	6	3	11056	3216	59133
5	3	3	8	4	8492	3020	52082
6	3	5	6	3	10566	2898	51632
7	3	5	7	5	13660	4444	82980
8	4	5	6	4	14022	4600	103539
9	5	7	6	4	28582	9550	257554
10	4	4	6	4	14022	4600	99781
11	4	4	6	4	14022	4600	99781
12	3	6	8	6	24147	7785	154551
13	5	5	7	5	25270	7670	171640
14	5	5	8	5	32160	8660	185170
15	5	5	7	6	25270	7670	171640
16	5	7	5	4	27752	8212	179844
17	5	7	9	5	47665	13825	324085
18	5	8	10	5	61735	17795	374983
19	6	8	10	5	77805	24930	678936
20	6	5	9	5	63545	20797	482440
21	4	5	8	4	23462	8916	194475
22	6	7	11	7	84060	26262	715211
23	7	7	10	7	99840	28496	801080
24	6	8	10	7	97062	28110	795672

We follow the parameters defined in Santos et al. (2020) to execute the RaD algorithm. The variables are considered as an integer in the following order $x_{ijs}, g_{pcs}, y_{pijs}, y^D_{pijs}, z_{swj}$ for all executions of the RaD algorithm. As the initial λ might have a significant effect on the efficiency of the solution process (CAPRARA et al., 1999), we define, based on experimentation, that our Lagrangian solution works very well with $\lambda^0_{cs} = 10,000, s \in S, c \in C(s)$.

Table 13 compares the quality of our optimization approach, LRM, with the real executed plan in terms of final costs (Solution), the number of ships (S), and the number of route inversions (RI) (north to south or south to north) by ships in the Brazilian cost. We included this last performance criterion in the table because it impacts logistics costs and deliveries within the desired time windows, as evidenced in Santos et al. (2020). Further, we compare the efficacy and efficiency of the Lagrangian approach with possible alternative optimization techniques, running model (114)–(158) using P-CPLEX and RaD. Both methods were run within a 6h limit. An additional column with the running time in seconds (CPU) is inserted in the table for each optimization approach. If an optimization approach cannot find a feasible solution within the CPU time limit, it is indicated by a "–". Table 13 also presents the initial Lower Bound (LB) for each instance, computed by the LR problem solved by the RaD algorithm, without considering segregated storage constraints.

Tabela 13 – Solution methods comparison

		Company			LRM				P-CPLEX		RaD	
Instance	LB	Solution	S	RI	Solution	S	RI	CPU	Solution	CPU	Solution	CPU
1	1206330	3152660	3	4	1392850	2	0	900	1406300	21600	1395320	21600
2	1035600	2774190	4	2	1164240	4	0	600	1184480	21600	1163900	21600
3	795570	1906450	3	2	937450	3	0	300	942400	21600	937450	21600
4	1575336	3450412	3	4	1722464	3	0	600	2211328	21600	1762496	21600
5	1279264	3010396	3	4	1353830	3	0	900	1422790	21600	1361200	21600
6	998740	2295640	3	4	1359070	3	1	900	1620450	21600	1593660	21600
7	1205990	2440570	5	3	1487050	5	0	1200	1702660	21600	1572910	21600
8	2096902	4112966	4	3	2670454	4	2	2100	4100810	21600	2966322	21600
9	3578946	8103942	4	7	4156718	4	2	5400	-	21600	-	21600
10	2161936	5211984	4	6	2663214	4	1	3600	-	21600	3399678	21600
11	1623734	4721258	4	6	1916256	4	0	7200	2225242	21600	2198456	21600
12	2050900	8402950	6	8	3566800	6	6	20000	-	21600	-	21600
13	2094670	5120470	5	4	2649620	5	1	8100	-	21600	-	21600
14	3154250	5624250	5	4	4187300	5	3	7200	-	21600	-	21600
15	2356310	6611860	6	5	3010780	6	2	7200	-	21600	-	21600
16	2189441	4212340	4	6	2974217	4	5	9000	-	21600	3810620	21600
17	1696610	3774270	5	4	2253500	5	1	9000	-	21600	-	21600
18	2069070	4522410	5	5	2795940	5	2	9000	-	21600	-	21600
19	2189720	6513390	5	5	3130520	5	3	20000	-	21600	-	21600
20	2963128	7190086	5	6	3626458	5	3	4500	-	21600	-	21600
21	1679530	3862170	4	6	1969390	4	0	6300	3102600	21600	2567210	21600
22	3919573	10566748	6	8	4762912	6	4	20000	-	21600	-	21600
23	4410214	11996254	7	7	5596070	7	4	20000	-	21600	-	21600
24	3858310	9863612	7	8	4958622	7	4	20000	-	21600	-	21600

The results in Table 13 show that the inclusion of segregated storage significantly affects the SRSP. Let us compare the solutions obtained by LB (without segregated storage) with the corresponding ones by the Company, LRM, P-CPLEX, and RaD. It is possible to observe considerable increases in the transportation costs. If we compare the LB solutions with the ones obtained by the Company and the LRM (both have solutions for all instances), the values increased on average 148% and 27%, respectively. These results ratify the importance of considering segregated storage (when this condition is a problem requisite) in the deep-sea SRSP.

Let us first compare the performance of the optimizing methods. Considering the CPU time limit imposed, P-CPLEX and RaD could only find feasible solutions for 10 and 12 instances, respectively. In general, they have solved instances with up to 9 ports in total, which can be categorized as small or medium. Although P-CPLEX and RaD can generate feasible solutions for some minor instances (1–7) in 3 hours, a good solution was only obtained when the CPU limit was increased to 6 hours. If we only consider the instances solved by either P-CPLEX or RaD, LRM has found better or equal solutions than the two remaining methods for all instances. LRM has significantly decreased the objective function compared with P-CPLEX and RaD, on average, in 14.4% and 9.4%, respectively. The optimization approaches have obtained similar solutions for tiny instances (up to seven ports and 6 products), especially LRM and RaD. However, LRM was significantly more efficient than P-CPLEX and RaD in obtaining good-quality solutions. LRM has solved all instances requiring, on average, 7666s, while P-CPLEX and RaD have required 21600s for solving small and medium size instances. When limiting the comparison to the instances solved by P-CLEX or RaD, LRM has solved them on average in 2100s and 2800s, being 10 and 7.7 times faster than P-CPLEX and RaD, respectively. In summary, LRM outperformed P-CPLEX and RaD both in quality and efficiency.

In comparison with the company's costs, the optimization approach provided solutions with lower total costs. On average, the LRM reduced the total costs by around 47.66%. Naturally, the reduction in total costs is dependent on the instance dimensions, number of products and ports, and the informed time window. However, this reduction should be carefully approached. The values informed by the company incorporate several real unexpected events that were not considered in our optimization approach, such as crew-related problems and extreme weather conditions. They can significantly increase the costs involved in transportation.

Nevertheless, the significant decreases in the objective function might be explained by the following three primary reasons: (i) the use of fewer vessels; (ii) the smaller number of route inversions by ships on the Brazilian coast due to a better distribution of cargo in the ships; or (iii) the company prioritized customer service over cost. Unfortunately, given the need for more information concerning the third item, we can only analyze the first two items. Company and LRM had the same number of used ships for all evaluated dimensions. Although there were differences in the ships' characteristics in the solutions, they were of little contribution to the obtained total cost differences, considering how the total costs were computed. In this way, the significant reduction in the number of route inversions by ships on the Brazilian coast assumed a more relevant role in cost reduction. LRM reduced by about 66.66%, on average, the number of route inversions, representing a decrease of 3 RI per plan, when comparing the company's and LRM's plans. In general, there was a proportionality between the reduction in the route inversions and costs. The average relation between route inversion reduction and total reduction cost was equal to 1.39, indicating that the reduction of the total costs also increased as the route inversion increased. However, there were some exceptions. In eight instances, the values were smaller than one. This effect can be explained by how the route inversion is accounted for in the total cost, considering the ship's load at the time of route inversion, justifying the direct lack of proportion between RI and cost reduction in some instances tested. We used instances 8, 12, and 19 to validate this argument. These instances presented a slight percentage reduction in route inversions, but significant cost reductions. In all three instances, route inversions occurred when ships were at more than 80% capacity, justifying that route inversions significantly reduced costs. In summary, the LRM outperformed the company's final costs in all evaluated instances, defining cargo routing and scheduling plans at a significantly lower cost while respecting all operational requirements of the problem, including port drafts, time windows, berth utilization, segregated stowage, and ship's stability.

4.7 CASE STUDY

This section describes the short-term planning carried out in mid-August/2022 for a fertilizer company operating in Brazil. The logistics department managers applied the developed optimization approach, given the results obtained in the previous section. Since most of the products came from Russia and Belarus, direct belligerents in the Ukraine war, there was an unprecedented necessity to evaluate different scenarios very quickly since several parameters changed values during the execution of the plan, a consequence of the uncertainties related to the conflict. The analysts were not directly involved with the negotiation processes with the contacted deep-sea chartering companies. The company managers provided all data and information to run the optimization approach.

The case study involved five pickup ports in Eastern Europe and five delivery ports on the Brazilian coast. The pickup ports are primarily located in Eastern Europe: 1 - St. Petersburg, RUS; 2 - Muuga, EST; 3 - Ventspils, LVA; 4 - Klaipeda, LTU; and 5 - Gdansk, POL. The delivery ports are in Southern Brazil: 1 - Aracruz; 2 - Vitória; 3 - Santos; 4 - São Francisco; and 5 - Rio Grande. Seven products were involved, as follows: (i) Muriate of potash; (ii) Sulphate of potash; (iii) Kainit; (iv) Sylvanite; (v) Potash + sodium admixture; (vi) Potash + magnesium admixture; and (vii) Sodium.

As usual, the company's main objective was to minimize transportation costs, respecting the time windows of each delivery port. Due to the Ukraine war, the main difference to previous planning processes was the limitation in the number and size of available vessels. During the analysis, as usual, all considered ships had five compartments with deep-sea handysize or handymax bulk carriers. Due to confidentiality issues, the ships were grouped into three categories based on their real load capacity, namely 30,000, 35,000, and 40,000 DWT, with associated daily freights of 1,000, 1,150, and 1,300 USD/day.

Table 14 presents a subset of the analyzed scenarios during planning. This table shows the number of ships in each category that composes set s. In the subsequent columns, we show the results obtained by the optimization approach in terms of the best solution (in US$), the actual fleet in terms of the three categories, the minimum utilization of a ship in the used fleet in percentage (MI), the number of route inversions (RI), and the maximum delay in days (MD) of a ship in any port of its route. These later performance measures are becoming increasingly relevant, as the company has directed its production towards premium products with customized specifications for large customers. A CPU time limit of 6300s was fixed for each scenario.

The first three scenarios analyzed consider that all ships are of the same category. Ships with 35k DWT presented the best results, closely followed by 40k DWT ships. Both ship categories could transport all products with five ships without route inversion. Although ships with 30k DWT are the cheapest ones to hire, it is necessary to hire an additional ship to transport all demanded products with segregated storage. Furthermore, one additional route inversion penalizes the total costs, as well a delay of 2 days in a shipping route. Scenario 2 presents the best total costs, but it introduces a small delay of one day in a port. Next, scenarios considering

Tabela 14 – Case study results for different scenarios

	# Ships per DWT			Optimization Approach					
Scenario	30k	35k	40k	Solution	30k	35k	40k	RI	MD
1	–	–	6	1815780	–	–	5	0	0
2	–	6	–	1767210	–	5	–	0	1
3	6	–	–	2128840	6	–	–	1	2
4	2	2	2	2788910	2	2	2	2	0
5	3	–	3	2181580	3	–	3	1	0
6	3	3	–	1794010	2	3	–	0	1
7	–	3	3	1890350	–	3	3	0	1
8	4	–	2	2059750	3	–	2	0	0
9	2	–	4	2053360	1	–	4	0	0
10	4	2	–	1874340	3	2	–	0	2
11	2	4	–	2167310	2	4	–	0	1
12	–	4	2	1867450	–	3	2	0	1
13	–	2	4	1831440	–	1	4	0	0

a fleet with different combinations of ship categories were analyzed. Scenario 4 considers the same number of ships for the three categories. This scenario ratifies the fact that the use of 30 DWT ships affects the objective function, increasing the number of route inversions. Scenarios 5, 6, and 7 consider the same number of ships but distributed in only two categories. Most of the best results are obtained in scenarios where 35 DWT ships are used. Scenarios 8–13 are requests from the company managers, possibly after receiving quotations from chartering companies. Scenario 13 dominates these six additional scenarios both in costs and in delays; however, it is closely followed, in costs, by scenarios 10 and 12, but with some delays. Although scenarios 8 and 9 use different compositions of ships 30 DWT and 40 DWT, they present almost the exact transportation costs without delay. Scenario 11 presents a high transportation costs. In summary, scenarios 2 and 6 present the best costs, with acceptable delays, followed by scenarios 1 and 13, with slightly higher costs but without delays in any route. These results were discussed with the company managers, offering them some important highlights during the negotiation process with charter companies.

During the chartering process, the possibility of increasing the number of products to be collected in the ports appeared due to negotiations between Brazil, Russia, and Belarus governments. Faced with this possibility and considering the difficulty of supply due to the Ukraine war, the company considered increases in the initial demands. The company managers exploited several values after directly contacting the suppliers in Europe. For economy sake, Table 15 shows the results, for scenarios 1, 2, 6, and 13 (the best ones by the previous analysis), of 10% and 20% increases in the total amount of all products to be collected in the pickup ports. These increases were equally applied to all products and used to illustrate the complete sensitivity analysis.

Scenarios 1 and 13, with larger ships, present the best results for both cargoes increasing. Scenarios 2 and 6 are dominated on the performance measures, requiring considerably higher

Tabela 15 – Sensitivity analysis considering increases in the amount of each product to be transported

		Optimization Approach						
Scenario	Cargo increase (%)	Solution	30k	35k	40k	AV	RI	MD
1	10%	1852980	–	–	5	0	0	0
	20%	2114640	–	–	6	0	0	1
2	10%	1935540	–	5	–	0	1	1
	20%	2381860	-	6	–	0	1	2
6	10%	2454860	3	3	-	0	2	3
	20%	3166190	3	4	–	1	3	1
13	10%	1861400	–	1	4	0	0	0
	20%	2097320	–	2	4	0	0	1

transportation costs, route inversions, and port delays than scenarios 1 and 13 for both demand increase situations. These results were expected since more cargo to be transported demands additional or larger ships. Both scenarios do not consider 40k DWT ships. Scenario 1 presents slightly better results when the cargo increases by 10%, while the opposite is observed when the increase is 20%. If we compare with the solutions in Table 14, there is an increase of 2% and 1.6% in the solutions of scenarios 1 and 13, respectively, for a 10% of increase in the cargo. In both scenarios, a single ship changes a pickup port in its route. For a 20% of increase in demand, the transportation costs have a more significant increase, of around 16.45% and 14.51% for scenarios 1 and 13, respectively. Almost all ship routes are changed for this cargo condition, reflecting higher transportation costs.

In summary, scenarios 2 and 6 were the best options for initial cargo demand. However, scenarios 1 and 13 were more robust solutions, with better handling possible cargo increases, due to the higher number of larger ships. The company's managers used the results of the whole planning process to sign the COA. Due to confidentiality issues, we cannot disclose the final CRS plan used for this supply.

The optimization approach was generally well-received by the company managers. The capacity of the optimization approach to offer good quality solutions in a reasonable running time was the main advantage pointed out by the managers. The managers also praised the possibility of performing sensitivity analysis. Furthermore, incorporating segregated storage and ship stability was relevant for an efficient negotiation with the chartering companies. However, there needs to be more clarity between the existence of an optimization approach and its application to real-world problems. A user-friendly decision support system is being implemented, ensuring flexibility and adaptability.

4.8 CONCLUSION

This paper proposes a Lagrangian relaxation approach for solving the deep-sea segregated storage multi-ship routing and scheduling problem of bulk carriers, simultaneously considering multi-commodity, heterogeneous fleet, time windows, draft limits, split load, and flexible cargoes. The problem is formulated as a MILP model, which objective function minimizes the total logistics costs, while the constraints represent the operational requirements associated with real-world problems. The solution to the problem specifies a fully operational plan, consisting of (i) each ship route as a feasible sequence of ports; (ii) the arrival and departure times in/from ports of each ship route; (iii) the quantities of each product to be loaded in a ship compartment in each pickup port; and (iv) the quantities of each product to be unloaded from a ship compartment in each delivery port.

Computational tests, using real-world instances, indicate that the Lagrangian heuristics effectively and efficiently solve the problem. The segregated storage constraints significantly impact the final solution value, increasing by around 140% and 30%, on average, the solutions obtained by the company and by the Lagrangian method, respectively. Furthermore, this constraint led CPLEX and the matheuristic introduced in Santos et al. (2020) to fail in finding a viable solution for most instances tested, given a CPU timeout of 6 hours. The Lagrangian method obtained the best solutions for all tested instances at a reasonable time, decreasing by around 40%, on average, the cost values informed by the company. The optimization approach was also evaluated in a real-world planning process, playing a relevant role in chartering.

There are several possible future extensions to this work. The proposed formulation can be improved by incorporating stochastic parameters, such as travel times, loading/unloading times, and demand. We also plan to evaluate the application of the developed optimization approach for chemical tanker routing and scheduling problems. Tankers have a much larger number of compartments and transport products with stricter security requirements than bulk carriers. Furthermore, a multi-objective version of the problem is being envisaged, in which environmental and service level aspects would be incorporated into the model. Remarkably, this reformulation of the problem is very challenging in developing a solution method capable of finding the Pareto frontier within a reasonable time.

5 CONCLUSIONS

"Cargo routing and scheduling problem in deep-sea transportation: Case study from a fertilizer company" presents an optimization approach to solving the real-world cargo routing and planning problems faced by chemical companies in supplying their blending plants in Brazil. The problem can be characterized as a multi-product pickup and delivery problem with heterogeneous fleets, dedicated compartments, TWs, and draft limits. A feasible route must satisfy several constraints, some of which are specific to the real-world problem at hand. Our optimization approach was developed on the basis of case studies, but as outlined in Section 2.6.3, it also applies to similar maritime problems faced by companies using trampers to transport large volumes of bulk cargo. It is relatively easy to implement.

We have extended his previous MILP formulation developed for tramp freight routes and planning (ARNESEN et al., 2017; MALAGUTI et al., 2018) to introduce heterogeneous fleets and multiple products. CPLEX was used to solve small and medium instances and provided relatively good solutions in a reasonable amount of time. For real large-scale instances, we have developed a mathematical approach that integrates a relaxed definition strategy and some built-in cutting planning heuristics into CPLEX. This matheuristic was evaluated using real-world examples and proved to be highly suitable for solving maritime problems, surpassing CPLEX results in both efficiency and effectiveness for nearly all cases tested. The matheuristic, in particular, demonstrated its ability to achieve much faster and better solutions than the human planner and CPLEX, especially in large and very large instances, and to deal with a wide variety of planning contexts. Similar results were obtained when applying the optimization approach to a more general problem without considering the details of the company context.

"Multi-objective optimization of the maritime cargo routing and scheduling problem" presents a multi-objective approach to maritime CRSP. This is a variant of the heterogeneous fleet collection and delivery problem with multiple products, time windows and draft limits. This problem is formulated as a MILP model, whose constraints represent different real-world requirements for marine transportation. The solution to this problem is (i) the order of ports that each vessel calls; (ii) the quantity of each product to be loaded and unloaded from the vessel at the port; Overall planning should simultaneously minimize overall transportation costs, scheduling, and partial shipment delays.

A weighted max-min-fuzzy solving approach was developed to solve the multi-objective formulation. Given the complexity of the MILP models generated during the analysis process, two heuristics were developed based on several cutting plan heuristics built into modern MILP solvers. The first uses the developed polishing and repairing algorithms to improve and speed up the CPLEX solver's solution search. The second algorithm integrates the first matheuristic-based algorithm with a modified relaxation and correction strategies. The former heuristic performed slightly better in preserving the Pareto front in an experimentation based on a real instance of the Brazilian company. New constraints have been added to the more general formulation to consider details of the real case. Insights gained from this real-world application demonstrate the flexibility and effectiveness of optimization approaches to address a variety of real-world problems.

"The segregated storage multi-ship routing and scheduling problem" proposes a Lagrangian relaxation approach for solving the deep-sea segregated storage multi-ship routing and scheduling problem of bulk carriers, simultaneously considering multi-commodity, heterogeneous fleet, time windows, draft limits, split load, and flexible cargoes. The problem is formulated as a MILP model where the objective function minimizes total logistics costs, and the constraints represent operational requirements related to real-world problems. The solution to this problem is to (i) specify a fully functional plan that composes each vessel's route as a sequence of possible ports. (ii) the time of arrival and departure from each port of each route; (iii) the quantity of each product to be loaded into the hold of the vessel at each port of collection; (iv) the quantity of each product unloaded from the hold at each port of delivery.

Computer tests using real-world instances show that the Lagrangian heuristic can solve problems effectively and efficiently. Individual storage limits have a large impact on the value of the final solution, averaging approximately 140% and 30% increases for the enterprise and Lagrangian solutions, respectively. Furthermore, this constraint led CPLEX and the matheuristic introduced in Santos et al. (2020) to fail in finding a viable solution for most instances tested, given a CPU timeout of 6 hours. The Lagrangian method achieved an optimal solution within a reasonable time for all tested instances, reducing the company's reported costs by an average of about 40%. The optimized approach was also evaluated in the actual planning process and played a relevant role during the charter.

REFERENCES

AGRA, A.; ANDERSSON, H.; CHRISTIANSEN, M.; WOLSEY, L. A maritime inventory routing problem: Discrete time formulations and valid inequalities. **Networks**, Wiley Online Library, v. 62, n. 4, p. 297–314, 2013.

AGRA, A.; CHRISTIANSEN, M.; DELGADO, A.; SIMONETTI, L. Hybrid heuristics for a short sea inventory routing problem. **European Journal of Operational Research**, Elsevier, v. 236, n. 3, p. 924–935, 2014.

AL-KHAYYAL, F.; HWANG, S.-J. Inventory constrained maritime routing and scheduling for multi-commodity liquid bulk, part I: Applications and model. **European Journal of Operational Research**, Elsevier, v. 176, n. 1, p. 106–130, 2007.

AMID, A.; GHODSYPOUR, S.; O'BRIEN, C. A weighted max–min model for fuzzy multi-objective supplier selection in a supply chain. **International Journal of Production Economics**, Elsevier, v. 131, n. 1, p. 139–145, 2011.

ANDERSSON, H.; CHRISTIANSEN, M.; FAGERHOLT, K. The maritime pickup and delivery problem with time windows and split loads. **INFOR: Information Systems and Operational Research**, Taylor & Francis, v. 49, n. 2, p. 79–91, 2011.

ANDERSSON, H.; HOFF, A.; CHRISTIANSEN, M.; HASLE, G.; LØKKETANGEN, A. Industrial aspects and literature survey: Combined inventory management and routing. **Computers & Operations Research**, Elsevier, v. 37, n. 9, p. 1515–1536, 2010.

APPELGREN, L. H. A column generation algorithm for a ship scheduling problem. **Transportation Science**, INFORMS, v. 3, n. 1, p. 53–68, 1969.

ARNESEN, M. J.; GJESTVANG, M.; WANG, X.; FAGERHOLT, K.; THUN, K.; RAKKE, J. G. A traveling salesman problem with pickups and deliveries, time windows and draft limits: Case study from chemical shipping. **Computers & Operations Research**, Elsevier, v. 77, p. 20–31, 2017.

BARBUCHA, D.; FILIPOWICZ, W. Segregated storage problems in maritime transportation. **IFAC Proceedings Volumes**, Elsevier, v. 30, n. 8, p. 557–561, 1997.

BATTARRA, M.; PESSOA, A. A.; SUBRAMANIAN, A.; UCHOA, E. Exact algorithms for the traveling salesman problem with draft limits. **European Journal of Operational Research**, Elsevier, v. 235, n. 1, p. 115–128, 2014.

BRAEKERS, K.; RAMAEKERS, K.; NIEUWENHUYSE, I. V. The vehicle routing problem: State of the art classification and review. **Computers & Industrial Engineering**, Elsevier, v. 99, p. 300–313, 2016.

BRAVO, M.; ROJAS, L. P.; PARADA, V. An evolutionary algorithm for the multi-objective pick-up and delivery pollution-routing problem. **International Transactions in Operational Research**, Wiley Online Library, v. 26, n. 1, p. 302–317, 2019.

BRØNMO, G.; CHRISTIANSEN, M.; FAGERHOLT, K.; NYGREEN, B. A multi-start local search heuristic for ship scheduling – a computational study. **Computers & Operations Research**, Elsevier, v. 34, n. 3, p. 900–917, 2007.

BRØNMO, G.; CHRISTIANSEN, M.; NYGREEN, B. Ship routing and scheduling with flexible cargo sizes. **Journal of the Operational Research Society**, Taylor & Francis, v. 58, n. 9, p. 1167–1177, 2007.

BRØNMO, G.; NYGREEN, B.; LYSGAARD, J. Column generation approaches to ship scheduling with flexible cargo sizes. **European Journal of Operational Research**, Elsevier, v. 200, n. 1, p. 139–150, 2010.

CAPRARA, A.; FISCHETTI, M.; TOTH, P. A heuristic method for the set covering problem. **Operations Research**, INFORMS, v. 47, n. 5, p. 730–743, 1999.

CHAN, F. T.; SHEKHAR, P.; TIWARI, M. Dynamic scheduling of oil tankers with splitting of cargo at pickup and delivery locations: a multi-objective ant colony-based approach. **International Journal of Production Research**, Taylor & Francis, v. 52, n. 24, p. 7436–7453, 2014.

CHANG, K.-H. **Design theory and methods using CAD/CAE: The computer aided engineering design series**. Waltham, MA 02451, USA: Academic Press, 2014.

CHRISTIANSEN, M.; FAGERHOLT, K.; FLATBERG, T.; HAUGEN, Ø.; KLOSTER, O.; LUND, E. H. Maritime inventory routing with multiple products: A case study from the cement industry. **European Journal of Operational Research**, Elsevier, v. 208, n. 1, p. 86–94, 2011.

CHRISTIANSEN, M.; FAGERHOLT, K.; NYGREEN, B.; RONEN, D. Ship routing and scheduling in the new millennium. **European Journal of Operational Research**, Elsevier, v. 228, n. 3, p. 467–483, 2013.

CHRISTIANSEN, M.; FAGERHOLT, K.; RONEN, D. Ship routing and scheduling: Status and perspectives. **Transportation Science**, INFORMS, v. 38, n. 1, p. 1–18, 2004.

COCCOLA, M. E.; DONDO, R.; MENDEZ, C. A. A MILP-based column generation strategy for managing large-scale maritime distribution problems. **Computers & Chemical Engineering**, Elsevier, v. 72, p. 350–362, 2015.

DAS, I.; DENNIS, J. E. A closer look at drawbacks of minimizing weighted sums of objectives for pareto set generation in multicriteria optimization problems. **Structural Optimization**, Springer, v. 14, n. 1, p. 63–69, 1997.

DE, A.; CHOUDHARY, A.; TIWARI, M. K. Multiobjective approach for sustainable ship routing and scheduling with draft restrictions. **IEEE Transactions on Engineering Management**, IEEE, v. 66, n. 1, p. 35–51, 2017.

DUMAS, Y.; DESROSIERS, J.; SOUMIS, F. The pickup and delivery problem with time windows. **European Journal of Operational Research**, Elsevier, v. 54, n. 1, p. 7–22, 1991.

EHRGOTT, M. Multiobjective optimization. **AI Magazine**, v. 29, n. 4, p. 47–47, 2008.

EVANS, J. R.; TSUBAKITANI, S. Solving the segregated storage problem with Benders' partitioning. **Journal of the Operational Research Society**, Springer, v. 44, n. 2, p. 175–184, 1993.

FAGERHOLT, K. Ship scheduling with soft time windows: An optimisation based approach. **European Journal of Operational Research**, Elsevier, v. 131, n. 3, p. 559–571, 2001.

FAGERHOLT, K.; CHRISTIANSEN, M. A combined ship scheduling and allocation problem. **Journal of the Operational Research Society**, Taylor & Francis, v. 51, n. 7, p. 834–842, 2000.

FAGERHOLT, K.; CHRISTIANSEN, M. A travelling salesman problem with allocation, time window and precedence constraints - an application to ship scheduling. **International Transactions in Operational Research**, Wiley Online Library, v. 7, n. 3, p. 231–244, 2000.

FAGERHOLT, K.; RONEN, D. Bulk ship routing and scheduling: solving practical problems may provide better results. **Maritime Policy & Management**, Taylor & Francis, v. 40, n. 1, p. 48–64, 2013.

FEILLET, D.; DEJAX, P.; GENDREAU, M.; GUEGUEN, C. An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. **Networks: An International Journal**, Wiley Online Library, v. 44, n. 3, p. 216–229, 2004.

FISCHETTI, M.; LODI, A. Repairing mip infeasibility through local branching. **Computers & Operations Research**, Elsevier, v. 35, n. 5, p. 1436–1445, 2008.

FISHER, M. L. The Lagrangian relaxation method for solving integer programming problems. **Management Science**, INFORMS, v. 27, n. 1, p. 1–18, 1981.

FRISKE, M. W.; BURIOL, L. S. Applying a relax-and-fix approach to a fixed charge network flow model of a maritime inventory routing problem. In: SPRINGER. **International Conference on Computational Logistics**. [S.l.], 2018. p. 3–16.

HELD, M.; KARP, R. M. The traveling-salesman problem and minimum spanning trees: Part II. **Mathematical Programming**, Springer, v. 1, n. 1, p. 6–25, 1971.

HVATTUM, L. M.; FAGERHOLT, K.; ARMENTANO, V. A. Tank allocation problems in maritime bulk shipping. **Computers & Operations Research**, Elsevier, v. 36, n. 11, p. 3051–3060, 2009.

JETLUND, A. S.; KARIMI, I. Improving the logistics of multi-compartment chemical tankers. **Computers & Chemical Engineering**, v. 28, n. 8, p. 1267 – 1283, 2004. ISSN 0098-1354.

JOZEFOWIEZ, N.; SEMET, F.; TALBI, E.-G. Multi-objective vehicle routing problems. **European Journal of Operational Research**, Elsevier, v. 189, n. 2, p. 293–309, 2008.

KELLY, J. D.; MANN, J. L. Flowsheet decomposition heuristic for scheduling: a relax-and-fix method. **Computers & Chemical Engineering**, Elsevier, v. 28, n. 11, p. 2193–2200, 2004.

KOBAYASHI, K.; KUBO, M. Optimization of oil tanker schedules by decomposition, column generation, and time-space network techniques. **Japan Journal of Industrial and Applied Mathematics**, Springer, v. 27, n. 1, p. 161–173, 2010.

KORSVIK, J. E.; FAGERHOLT, K. A tabu search heuristic for ship routing and scheduling with flexible cargo quantities. **Journal of Heuristics**, Springer, v. 16, n. 2, p. 117–137, 2010.

KORSVIK, J. E.; FAGERHOLT, K.; LAPORTE, G. A tabu search heuristic for ship routing and scheduling. **Journal of the Operational Research Society**, Taylor & Francis, v. 61, n. 4, p. 594–603, 2010.

KORSVIK, J. E.; FAGERHOLT, K.; LAPORTE, G. A large neighbourhood search heuristic for ship routing and scheduling with split loads. **Computers & Operations Research**, Elsevier, v. 38, n. 2, p. 474–483, 2011.

LADAGE, A.; BAATAR, D.; KRISHNAMOORTHY, M.; MAHAJAN, A. A revised formulation, library and heuristic for a chemical tanker scheduling problem. **Computers & Operations Research**, Elsevier, v. 133, p. 105345, 2021.

LIN, C.-C. A weighted max-min model for fuzzy goal programming. **Fuzzy Sets and Systems**, Elsevier, v. 142, n. 3, p. 407–420, 2004.

LIN, D.-Y.; TSAI, Y.-Y. The ship routing and freight assignment problem for daily frequency operation of maritime liner shipping. **Transportation Research Part E: Logistics and Transportation Review**, Elsevier, v. 67, p. 52–70, 2014.

LÜBBECKE, M. E.; DESROSIERS, J. Selected topics in column generation. **Operations Research**, INFORMS, v. 53, n. 6, p. 1007–1023, 2005.

MALAGUTI, E.; MARTELLO, S.; SANTINI, A. The traveling salesman problem with pickups, deliveries, and draft limits. **Omega**, Elsevier, v. 74, p. 50–58, 2018.

MALLIAPPI, F.; BENNELL, J. A.; POTTS, C. N. A variable neighborhood search heuristic for tramp ship scheduling. In: SPRINGER. **International Conference on Computational Logistics**. [S.l.], 2011. p. 273–285.

MANSOURI, S. A.; LEE, H.; ALUKO, O. Multi-objective decision support to enhance environmental sustainability in maritime shipping: a review and future directions. **Transportation Research Part E: Logistics and Transportation Review**, Elsevier, v. 78, p. 3–18, 2015.

MARLER, R. T.; ARORA, J. S. The weighted sum method for multi-objective optimization: new insights. **Structural and Multidisciplinary Optimization**, Springer, v. 41, n. 6, p. 853–862, 2010.

MOHAMMADI, M.; GHOMI, S. F.; KARIMI, B.; TORABI, S. A. Rolling-horizon and fix-and-relax heuristics for the multi-product multi-level capacitated lotsizing problem with sequence-dependent setups. **Journal of Intelligent Manufacturing**, Springer, v. 21, n. 4, p. 501–510, 2010.

NEO, K.-H.; OH, H.-C.; KARIMI, I. Routing and cargo allocation planning of a parcel tanker. **Computer Aided Chemical Engineering**, Elsevier, v. 21, p. 1985–1990, 2006.

OMBUKI, B.; ROSS, B. J.; HANSHAR, F. Multi-objective genetic algorithms for vehicle routing problem with time windows. **Applied Intelligence**, Springer, v. 24, n. 1, p. 17–30, 2006.

OSTERMEIER, M.; HENKE, T.; HÜBNER, A.; WÄSCHER, G. Multi-compartment vehicle routing problems: State-of-the-art, modeling framework and future directions. **European Journal of Operational Research**, Elsevier, v. 292, n. 3, p. 799–817, 2021.

PAPAGEORGIOU, D. J.; NEMHAUSER, G. L.; SOKOL, J.; CHEON, M.-S.; KEHA, A. B. MIRPLib–a library of maritime inventory routing problem instances: Survey, core model, and benchmark results. **European Journal of Operational Research**, Elsevier, v. 235, n. 2, p. 350–366, 2014.

PAQUETE, L.; STÜTZLE, T. A two-phase local search for the biobjective traveling salesman problem. In: FONSECA, C. M.; FLEMING, P. J.; ZITZLER, E.; THIELE, L.; DEB, K. (Ed.). **Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science vol. 2632**. [S.l.]: Springer, 2003. p. 479–493.

PARRAGH, S. N.; DOERNER, K. F.; HARTL, R. F. A survey on pickup and delivery models part II: Transportation between pickup and delivery locations. **Journal für Betriebswirtschaft**, v. 58, n. 2, p. 81–117, 2008.

POCHET, Y.; WOLSEY, L. A. **Production planning by mixed integer programming**. [S.l.]: Springer Science & Business Media, 2006.

RAKKE, J. G.; CHRISTIANSEN, M.; FAGERHOLT, K.; LAPORTE, G. The traveling salesman problem with draft limits. **Computers & Operations Research**, Elsevier, v. 39, n. 9, p. 2161–2167, 2012.

RODRIGUES, V. P.; MORABITO, R.; YAMASHITA, D.; SILVA, B. J. da; RIBAS, P. C. Ship routing with pickup and delivery for a maritime oil transportation system: MIP model and heuristics. **Systems**, Multidisciplinary Digital Publishing Institute, v. 4, n. 31, p. 1–21, 2016.

RONEN, D. Cargo ships routing and scheduling: Survey of models and problems. **European Journal of Operational Research**, Elsevier, v. 12, n. 2, p. 119–126, 1983.

RONEN, D. Ship scheduling: The last decade. **European Journal of Operational Research**, Elsevier, v. 71, n. 3, p. 325–333, 1993.

ROPKE, S.; CORDEAU, J.-F. Branch and cut and price for the pickup and delivery problem with time windows. **Transportation Science**, INFORMS, v. 43, n. 3, p. 267–286, 2009.

ROTHBERG, E. An evolutionary algorithm for polishing mixed integer programming solutions. **INFORMS Journal on Computing**, INFORMS, v. 19, n. 4, p. 534–541, 2007.

SANTOS, P. T. G. dos; BORENSTEIN, D. Multi-objective optimization of the maritime cargo routing and scheduling problem. **International Transactions in Operational Research**, Wiley Online Library, 2022.

SANTOS, P. T. G. dos; KRETSCHMANN, E.; BORENSTEIN, D.; GUEDES, P. C. Cargo routing and scheduling problem in deep-sea transportation: Case study from a fertilizer company. **Computers & Operations Research**, v. 119, p. 104934, 2020.

SAVELSBERGH, M. W.; SOL, M. The general pickup and delivery problem. **Transportation Science**, INFORMS, v. 29, n. 1, p. 17–29, 1995.

SHEN, Q.; CHU, F.; CHEN, H. A Lagrangian relaxation approach for a multi-mode inventory routing problem with transshipment in crude oil transportation. **Computers & Chemical Engineering**, Elsevier, v. 35, n. 10, p. 2113–2123, 2011.

SONG, D.-P.; LI, D.; DRAKE, P. Multi-objective optimization for planning liner shipping service with uncertain port times. **Transportation Research Part E: Logistics and Transportation Review**, Elsevier, v. 84, p. 1–22, 2015.

SONG, J.-H.; FURMAN, K. C. A maritime inventory routing problem: Practical approach. **Computers & Operations Research**, Elsevier, v. 40, n. 3, p. 657–665, 2013.

STÅLHANE, M.; ANDERSSON, H.; CHRISTIANSEN, M.; CORDEAU, J.-F.; DESAULNIERS, G. A branch-price-and-cut method for a ship routing and scheduling problem with split loads. **Computers & Operations Research**, Elsevier, v. 39, n. 12, p. 3361–3375, 2012.

STANZANI, A. d. L.; PUREZA, V.; MORABITO, R.; SILVA, B. J. V. d.; YAMASHITA, D.; RIBAS, P. C. Optimizing multiship routing and scheduling with constraints on inventory levels in a Brazilian oil company. **International Transactions in Operational Research**, Wiley Online Library, v. 25, n. 4, p. 1163–1198, 2018.

TROTTIER, L.-P.; CORDEAU, J.-F. Solving the vessel routing and scheduling problem at a canadian maritime transportation company. **INFOR: Information Systems and Operational Research**, Taylor & Francis, p. 1–26, 2019.

UGGEN, K. T.; FODSTAD, M.; NØRSTEBØ, V. S. Using and extending fix-and-relax to solve maritime inventory routing problems. **TOP**, Springer, v. 21, n. 2, p. 355–377, 2013.

VELDHUIZEN, D. A. V.; LAMONT, G. B. **Multiobjective evolutionary algorithm research: A history and analysis**. Wright-Patterson AFB, Ohio, 1998.

VILHELMSEN, C.; LUSBY, R. M.; LARSEN, J. Tramp ship routing and scheduling with voyage separation requirements. **OR Spectrum**, Springer, v. 39, n. 4, p. 913–943, 2017.

WANG, J.; ZHOU, Y.; WANG, Y.; ZHANG, J.; CHEN, C. P.; ZHENG, Z. Multiobjective vehicle routing problems with simultaneous delivery and pickup and time windows: formulation, instances, and algorithms. **IEEE Transactions on Cybernetics**, IEEE, v. 46, n. 3, p. 582–594, 2016.

WANG, X.; ARNESEN, M. J.; FAGERHOLT, K.; GJESTVANG, M.; THUN, K. A two-phase heuristic for an in-port ship routing problem with tank allocation. **Computers & Operations Research**, Elsevier, v. 91, p. 37–47, 2018.

YAMASHITA, D.; SILVA, B. J. V. da; MORABITO, R.; RIBAS, P. C. A multi-start heuristic for the ship routing and scheduling of an oil company. **Computers & Industrial Engineering**, Elsevier, v. 136, p. 464–476, 2019.

ZHANG, H.; ZHANG, Q.; MA, L.; ZHANG, Z.; LIU, Y. A hybrid ant colony optimization algorithm for a multi-objective vehicle routing problem with flexible time windows. **Information Sciences**, Elsevier, v. 490, p. 166–190, 2019.

ZHANG, W.; YANG, D.; ZHANG, G.; GEN, M. Hybrid multiobjective evolutionary algorithm with fast sampling strategy-based global search and route sequence difference-based local search for VRPTW. **Expert Systems with Applications**, Elsevier, v. 145, p. 113151, 2020.

ZIMMERMANN, H.-J. Fuzzy programming and linear programming with several objective functions. **Fuzzy Sets and Systems**, Elsevier, v. 1, n. 1, p. 45–55, 1978.

ZITZLER, E.; DEB, K.; THIELE, L. Comparison of multiobjective evolutionary algorithms: Empirical results. **Evolutionary Computation**, MIT Press, v. 8, n. 2, p. 173–195, 2000.

ZITZLER, E.; THIELE, L. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. **IEEE Transactions on Evolutionary Computation**, IEEE, v. 3, n. 4, p. 257–271, 1999.

ZOGRAFOS, K. G.; ANDROUTSOPOULOS, K. N. A heuristic algorithm for solving hazardous materials distribution problems. **European Journal of Operational Research**, Elsevier, v. 152, n. 2, p. 507–519, 2004.

ZOU, S.; LI, J.; LI, X. A hybrid particle swarm optimization algorithm for multi-objective pickup and delivery problem with time windows. **Journal of Computers**, v. 8, n. 10, p. 2583–2589, 2013.