Resource-Constrained Scheduling for Maritime Traffic Management

#3072

Abstract

In this paper, we address the problem of mitigating congestion and preventing hotspots in maritime traffic on a busy water area, with the Singapore Straits and Port waters as a case study. The problem is cast as a variant of continuous-time event-based resource-constrained project scheduling problem (RCPSP), where vessel navigation is modeled as sequences of activities, each of which requires certain resources including traversable sea space. Congestion is controlled by putting a limit on the capacity of resources. Our main contribution is to formalize this problem as a RCPSP variant by incorporating several maritime-related constraints. For scalability and quality bounds, we also develop a rolling-horizon Benders decomposition approach. To rigorously test our approach, we build a maritime traffic simulator using e-navigation charts of Singapore waters. We show our approach's effectiveness in reducing the congestion using real GPS trajectories of all the ships in Singapore straits over a 55-day period.

1 Introduction

The Straits of Malacca and Singapore is a key trade route and one of the busiest shipping areas in the world, providing the shortest route between the Indian Ocean and the South China Sea. This makes it a popular route for oil tankers traveling between the Persian Gulf and East Asia, as well as many other vessels. It covers a length of approximately 1,000 km and is very shallow in some areas, with the narrowest point just about 2.8 km wide. In Singapore Strait alone, there are more than 50,000 vessel movements in a year with an average of 200-300 passages per day. The port of Singapore, as a result, has been consistently ranked in the top three of the world's busiest ports in terms of total shipping tonnage, with all of the anchorages heavily utilized, where more than 95% of the vessels stay for less than 10 days. Movements in and around the strait, port waters and anchorages, therefore, require extreme caution and careful planning [Witherby Publishing Group, 2015]. Our goal in this work is to suggest appropriate adjustments to vessel schedules such that the density of traffic in Singapore straits is reduced, thereby increasing the safety of navigation, while keeping the delays caused to ships at a minimum.

The water area used as the case study for this work is shown in Figure 1 based on the e-navigation charts of the strait. It is divided into two subareas. We call the one above the dashed line Singapore Port Waters and the one below Singapore Strait. One major feature on the strait is the *traffic separation scheme* (TSS). It is a set of mandatory one-directional routes designed to reduce collision risk among vessels transitioning through the strait. Features inside the port waters include the *fairway*, where vessels may travel in any direction, and the *berths* and *anchorages*. At various points along the boundary of the port waters are pilot boarding grounds where pilots embark or disembark from vessels that are entering or leaving the port waters. Pilots are rigorously-trained personnels under the port authority who guide shipmasters in maneuvering the port waters. They are compulsory for all large vessels such as tankers. The water area is also divided into smaller zones. Most are obtained by dividing the TSS, fairway, and anchorages into smaller parts. These zones form the basis of our scheduling approach. A small sample of the zones are shown in Figure 1 for illustration.

Vessels in the area can be grouped into three broad categories depending on their activities: those that are transiting, those that are entering to anchor/berth before leaving, and those that operates only within the area such as ferries with fixed routes and schedules, tug/pilot boats and patrol boats. Figure 1 shows example activities for vessels from the first two categories. Activities of vessel 1, which is transiting the strait, are denoted by dotted arrowed lines, while vessel 2, which is anchoring/berthing, by solid arrowed lines. As shown, vessel 1 is traveling on the westbound lane of the TSS, and its activities correspond to traversing lane's zones. E.g., its third activity (1,3) is traversing zone 3W3 where the required resource to perform this activity is some physical space in the zone. Vessel 2, on the other hand, has a more varied sequence of activities, from traversing the TSS, to picking up pilot to anchoring and berthing. Furthermore, the anchoring activity (2,11) may be further divided into a series of sub-activities like taking stores, making crew changes and undergoing maintenance. These activities also require other resources beside sea space. The crossing area, marked by the triangle sign is where the activities of the two vessels cross

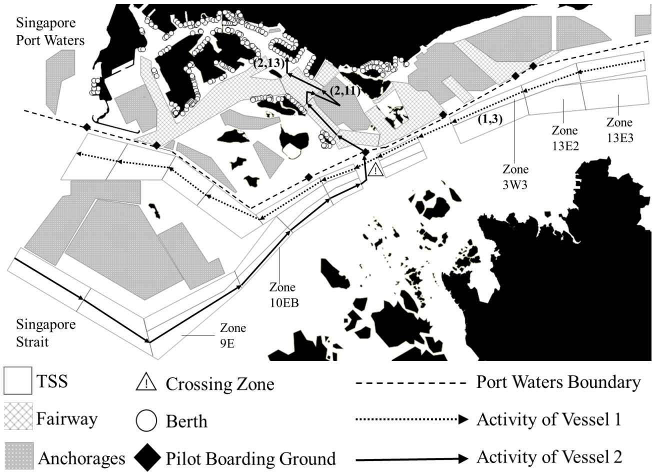


Figure 1: Examples of vessel activities within the Singapore Strait and Port Waters. Solid black regions are landmass

each other. This is one of the incident-prone areas since it is the cross point between most vessels that are entering/exiting the port waters and those that are transiting.

Based on our discussions with port authorities in Singapore, a significant number of near-miss and vessel collision incidents are related to the congested conditions in the Singapore straits. There are traffic information provided through vessel traffic information system (VTIS) to warn vessels of potential incidents. These are, however, short term in nature, with lead time of 5-10 minutes. While these may help shipmasters navigate through tough encounters involving multiple vessels, they are incapable of regulating the flows of traffic that create those tough encounters in the first place. To do so, the whole lifecycle of the vessels must be considered, from the moment they enter the strait to the moment they leave, and plans that regulate traffic hours in advance are required. In this paper, we propose a resourced-constrained scheduling approach to arrive at such plans. Such advance planning is possible for large ships (tankers and cargo vessels) as pre-arrival passage plan for such ships is available 24 hours in advance [Maritime and Port Authority of Singapore, 2016].

In this work, we focus primarily on the navigation of vessels. For a vessel, the traversing of a zone in the strait/port waters is considered an activity. Each vessel is modeled as performing a sequence of such navigation activities with no precedence relation with other vessels. In this sense, it is a restricted form of the standard RCPSP [Kolisch and Hartmann, 1999], with dependency graphs consisting of a set of linear dependencies. Nevertheless, the problem is no less difficult than the standard RCPSP for the following reasons. There is no lag/slack time between any two consecutive activities as vessels cannot stop mid way during navigation on TSS or fairway. This might be seen as a special case of RCPSP/max with time lags set to zero. The vessels, however, are allowed to speed up or slow down on the TSS/fairway up to certain limits depending on the zones. This translates to activities having minimum and maximum durations, where the actual durations are part of the *decision variables*—duration of an activity directly translates into an average speed as the length of the zone is known. This is the crucial difference from an RCPSP where activity durations are either fixed or follow some given distribution, whereas in our case, activity durations are output of the scheduler. And since we require higher precision on the durations of activities, especially when traversing the TSS, we opt for event-based continuous-time formulation [Koné *et al.*, 2011], instead of the more common discrete-time formulation. We also develop Benders decomposition technique to solve the underlying math program which cannot be solved using standard optimization solvers. Our results on a real world data set consisting of GPS traces of ships in the Singapore strait and port waters over a 55-day period show that our approach effectively minimizes peak hour traffic density thereby significantly reducing the congestion.

2 Related Works

Studies on vessel collisions from the maritime domain have been done mostly on the micro level, where risks of collisions are computed based on physical and navigational characteristics of vessels [Pietrzykowski and Uriasz, 2009; van Dorp and Merrick, 2011; Qu *et al.*, 2011]. These studies are also usually descriptive in nature, providing risk measures but not prescriptive actions. One reason being the complexity of navigating through an encounter, since the best actions usually depend on a lot of human factors and judgments. In this work, on the other hand, we are looking at preventive coordination at a macro level, where the prescriptions given to vessels are not navigational in nature, but closer to a schedule or a passage plan. In the future, however, we would like to incorporate more elements from the domain knowledge into our model, like using existing risk measures to determine the resource consumptions of vessels.

RCPSP is an extensively studied problem, and a large amount of works exist in both Operations Research and Constraint Programming literature. One way to categorize the literature is to divide them based on the formulation used. Discrete-time formulations are more prevalent in CP/AI literature, where techniques such as lazy clause generation and time-tabling constraint propagation have been employed successfully to resolve hard benchmark problems [Vilím, 2011; Schutt *et al.*, 2013a; 2013b]. Continuous-time formulations can be further divided into "enumerative" formulations, with exponential number of variables/constraints (one example is [Alvarez-Valdés and Tamarit, 1993]), and "compact" formulations [Mingozzi *et al.*, 1998], with polynomial number of variables/constraints. Techniques like column generation, which solve the problem by adding constraints iteratively, work better given an enumerative formulation. And decomposition techniques like Benders work better with compact formulations. See [Koné *et al.*, 2011] for a survey of different math programming formulations of RCPSP. Such techniques cannot be directly used in our problem as activity duration is itself a decision variable unlike standard RCPSPs.

And finally, Benders decomposition has been successfully applied in various problems with scheduling elements, like airline scheduling [Cordeau *et al.*, 2001; Merciera *et* *al.*, 2005; Papadakos, 2009], prescriptive evacuation problem [Romanski and Hentenryck, 2016] and network design [Marín and Jaramillo, 2009].

3 Benders Decomposition

In this section, we present Benders decomposition formulation of the problem as described in Section 1. We start with a short introduction of Benders decomposition in Section 3.1 and the notations used in Section 3.2. The master problem, which deals with assignments of activities to events is given in Section 3.3, and the subproblem, which deals with the times of events, in Section 3.4. And finally, Section 3.5 describes the Benders cuts that are added to the master problem.

3.1 Preliminaries

Benders decomposition [Benders, 1962] is a technique for decomposing mixed integer linear programs of the form:

$$\begin{array}{lll} \text{[P]} & \min & c_1x + c_2y \\ & \text{s.t.} & Ax + By \geq b \\ & & x \geq 0, y \in \mathbb{Y} \end{array}$$

where x are continuous real-valued variables and $\mathbb{Y}$ is a discrete (integral) domain, into the master problem [MP]:

$$\begin{array}{llll} \text{[MP]} & \min & Q & \\ & \text{s.t.} & c_2y + \pi(b - By) \leq Q & \forall \pi \in \Pi_p \quad (1) \\ & & \pi(b - By) \leq 0 & \forall \pi \in \Pi_r \quad (2) \\ & & y \in \mathbb{Y} & \end{array}$$

and the subproblem [PSP] (with its dual [DSP]):

$$\begin{array}{lll} \text{[PSP]} & \min & c_1x \\ & \text{s.t.} & Ax \geq b - By \\ & & x \geq 0 \end{array} \qquad \begin{array}{lll} \text{[DSP]} & \max & (b - By)\pi \\ & \text{s.t.} & \pi A \leq c_1 \\ & & \pi \geq 0 \end{array}$$

where Π_p and Π_r are the set of extreme points and rays, respectively, of the dual subproblem [DSP]. The constraints in (1) are called optimality cuts and the ones in (2) are called feasibility cuts. It has be shown that [P] and [MP] are equivalent. The size of the sets Π_p and Π_r can be extremely large and the idea behind Benders decomposition is to start with initial sets $\Pi_p^0 \subseteq \Pi_p$ and $\Pi_r^0 \subseteq \Pi_r$ (usually empty), and to iteratively add to these sets and resolving [MP], with the hope that optimality can be reached with a much smaller sets $\Pi_p^* \subseteq \Pi_p$ and $\Pi_r^* \subseteq \Pi_r$. Solving the restricted master problem provides a lower bound to the optimal value, while solving the subproblem provides an upper bound. Convergence is guaranteed by the fact that the set of extreme points and rays are finite.

3.2 Notations

Let $\mathcal{R}$ be the set of resources. Examples of a resource are an area of traversable sea space, an anchorage area, a berth, available pilots, etc. Each resource $r \in \mathcal{R}$ has a capacity $K_r \in \mathbb{N}$. Let $\mathcal{N}$ be the set of n vessels. To each vessel $i \in \mathcal{N}$, is associated a list of m activities to be carried out sequentially without time lag between two consecutive activities. For clarity of presentation, we assume without loss of generality, that all vessels have the same number of activities m. In addition, each vessel $i \in \mathcal{N}$ has a release time s_i which is the earliest start time of its first activity. We denote by (i, j) the j-th activity of vessel i, and $\mathcal{A}$ the set of all activities ($|\mathcal{A}| = nm$). Each activity $(i, j) \in \mathcal{A}$ requires a set of resources, and we denote by a_r^{ij}, the amount of resource r required by activity (i, j). Furthermore, each activity (i, j) has a minimum and a maximum processing time, denoted by $T_{\min}^{ij}$ and $T_{\max}^{ij}$ respectively.

Let $\mathcal{E} = \{1, \dots, nm\}$ be the set of events. The first set of decision variables is $z = \{z_e^{ij} | (i, j) \in \mathcal{A}, e \in \mathcal{E}\}$ where $z_e^{ij} = 1$ iff activity (i, j) is being carried out at t_e, and $z_e^{ij} = 0$ otherwise. When expressing a constraint, the dummy variable z_0^{ij} or z_{nm+1}^{ij} might be used, in which case, its value is always 0 for any activity (i, j). The second set of decision variables is $t = \{t_e | e \in \mathcal{E}\}$, where t_e denotes the time of event e. The times of events are ordered sequentially, that is, $t_1 \leq \cdots \leq t_{nm}$. To simplify notations, we define the following derivative variables $w = \{w_{ef}^{ij} | (i, j) \in \mathcal{A}, e \in \mathcal{E}, f \in \mathcal{E}, e < f\}$, where $w_{ef}^{ij} = (z_e^{ij} - z_{e-1}^{ij}) + (z_{f-1}^{ij} - z_f^{ij})$. In other words,

$$w_{ef}^{ij} = \begin{cases} 2 & \text{if } (i,j) \text{ starts at } e \text{ and ends at } f, \\ 1, 0, -1 & \text{otherwise.} \end{cases}$$

The objective of the problem is to minimize the makespan, i.e., the total time required to complete all activities.

3.3 Master Problem

The master problem [MP] can be formulated as the following integer linear program with a single continuous variable as objective:

$$\text{[MP]} \qquad \min Q \tag{3}$$

subject to:

$$\forall (i,j) \in \mathcal{A}: \sum_{e=1}^{nm} z_e^{ij} \geq 1 \tag{4}$$

$$\forall (i,j) \in \mathcal{A}, e \in \mathcal{E} \setminus \{1\}: \sum_{e'=1}^{e-1} z_{e'}^{ij} \leq (e-1)(1 - (z_e^{ij} - z_{e-1}^{ij})) \tag{5}$$

$$\forall (i,j) \in \mathcal{A}, e \in \mathcal{E} \setminus \{1\}: \sum_{e'=e}^{nm} z_{e'}^{ij} \leq (nm - e + 1)(1 + (z_e^{ij} - z_{e-1}^{ij})) \tag{6}$$

$$\forall (i,j) \in \mathcal{A} | j \neq m, e \in \mathcal{E}: \sum_{e'=1}^{e} z_{e'}^{ij+1} \leq (1 - z_e^{ij}) e \tag{7}$$

$$\forall (i,j) \in \mathcal{A} | j \neq m, e \in \mathcal{E} \setminus \{1\}: z_e^{ij+1} \geq z_{e-1}^{ij} - z_e^{ij} \tag{8}$$

$$\forall e \in \mathcal{E}, r \in \mathcal{R}: \sum_{i=1}^{n} \sum_{j=1}^{m} a_r^{ij} z_e^{ij} \leq K_r \tag{9}$$

$$\forall \pi \in \Pi_p: \phi(\pi, z) \leq Q \tag{10}$$

$$\forall \pi \in \Pi_r: \phi(\pi, z) \leq 0 \tag{11}$$

$$\forall (i,j) \in \mathcal{A}, e \in \mathcal{E}: z_e^{ij} \in \{0,1\}, Q \in \mathbb{R} \tag{12}$$

where $\phi(\pi, z)$ is the expression of the Benders cuts which will be elaborated in the Section 3.5.

3.4 Subproblem

Given an assignment to the binary variables z, the primal subproblem [PSP] is given by the following linear program:

$$\text{[PSP]} \qquad q(z) = \min M \tag{13}$$

subject to:

$$\forall i \in \mathcal{N}, e \in \mathcal{E}: t_e \geq (z_e^{i1} - z_{e-1}^{i1}) s_i \tag{14}$$

$$\forall i \in \mathcal{N}, e \in \mathcal{E}: M \geq t_e + (z_e^{im} - z_{e-1}^{im}) T_{\min}^{im} \tag{15}$$

$$\forall e \in \mathcal{E} \backslash \{nm\}: t_{e+1} \geq t_e \tag{16}$$

$$\forall (i,j) \in \mathcal{A}, e, f \in \mathcal{E} | f > e: t_f \geq t_e + (w_{ef}^{ij} - 1) T_{\min}^{ij} \tag{17}$$

$$\forall (i,j) \in \mathcal{A} | j \neq m, e, f \in \mathcal{E} | f > e:$$

$$t_f \leq t_e + (w_{ef}^{ij} - 1) T_{\max}^{ij} + (2 - w_{ef}^{ij}) U_{ef} \tag{18}$$

$$\forall i \in \mathcal{N}, e \in \mathcal{E}: t_e \in \mathbb{R}_{\geq 0}, D_i \in \mathbb{R} \tag{19}$$

where U_{ef} is a suitable upper bound. Big number formulation in MILP may loosen its LP relaxation and incur degenerate behavior if not handled correctly. In our case, a loose upper bound causes the cuts generated by the subproblem to be weak. The following value of U_{ef} gives a tight upper bound:

$$U_{ef} = (f - e) \left[\max_{(i,j) \in \mathcal{A}} \left\{ T_{\max}^{ij}, \max_{i'} s_{i'} - \min_{i'} s_{i'} \right\} \right]. \tag{20}$$

Proposition 1. *For any events $e, f \in \mathcal{E} | f > e$, U_{ef} as defined in Equation* (20) *is a tight upper bound for the time difference between the two events.*

3.5 Pareto-Optimal Benders Cuts

Let α, β, γ and δ be the dual variables corresponding to constraints (14), (15), (17) and (18) respectively, in other words, $\pi = (\alpha, \beta, \gamma, \delta)$. We can ignore constraint (16) which has constant term zero and no binary variables, and thus does not contribute any term to the Benders cut. The expression of the Benders cuts $\phi(\pi, z)$ is then given by

$$\phi(\pi, z) =$$

$$\sum_{i=1}^{n} \sum_{e=1}^{nm} \alpha_e^i s_i (z_e^{i1} - z_{e-1}^{i1}) + \sum_{i=1}^{n} \sum_{e=1}^{nm} \beta_e^i T_{\min}^{im} (z_e^{im} - z_{e-1}^{im}) \tag{21}$$

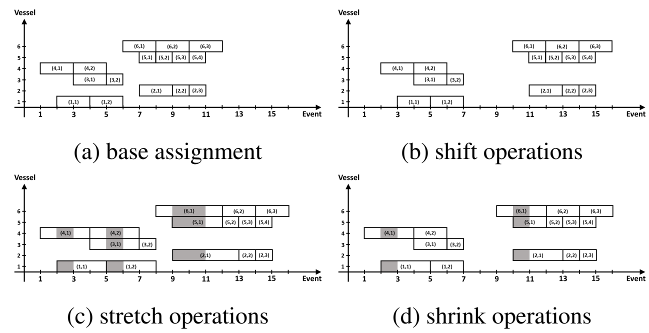
$$+ \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{e=1}^{nm-1} \sum_{f=e+1}^{nm} \gamma_{ef}^{ij} T_{\min}^{ij} w_{ef}^{ij} \tag{22}$$

$$- \sum_{i=1}^{n} \sum_{j=1}^{m-1} \sum_{e=1}^{nm-1} \sum_{f=e+1}^{nm} \delta_{ef}^{ij} (T_{\max}^{ij} - U_{ef}) w_{ef}^{ij} \tag{23}$$

$$- \sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{e=1}^{nm-1} \sum_{f=e+1}^{nm} \gamma_{ef}^{ij} T_{\min}^{ij} \tag{24}$$

$$+ \sum_{i=1}^{n} \sum_{j=1}^{m-1} \sum_{e=1}^{nm-1} \sum_{f=e+1}^{nm} \delta_{ef}^{ij} (T_{\max}^{ij} - 2U_{ef}). \tag{25}$$

When π is given, the terms (21)–(23) are a linear expression in z while the terms (24) and (25) are constants. Furthermore, term (25) is a nonpositive constant since, by definition, $U_{ef} \geq T_{\max}^{ij}$ and $\delta_{ef}^{ij} \geq 0$. This is where having a loose U_{ef} values may weaken the cut significantly because of the large negative constant it may have. We employ the common technique of [Magnanti and Wong, 1981] to generate Pareto-optimal cuts to be added to the master problem.



(a) base assignment (b) shift operations

(c) stretch operations (d) shrink operations

Figure 2: Example of equivalent assignments

4 Symmetry-Breaking Constraints

Similar to many combinatorial optimization problems, symmetry occurs in our problem as formulated in Section 3. Specifically, we refer to the assignments of activities to events in the master problem [MP], where two assignments are symmetric if both return the same objective value (makespan) when passed to the subproblem [PSP]. Symmetries are problematic because they increase the search space and algorithms like branch-and-bound/cut are particularly susceptible because a lot of efforts will be spent on visiting symmetric states in the search tree. Methods for dealing with symmetries exist in both Operations Research and Constraint Programming literature. For recent surveys on the subject, see [Gent *et al.*, 2006; Margot, 2010; Walsh, 2012]. Checking for and removing all symmetric solutions are a hard problem in itself and at times, might not be the best thing to do. In this section, we present some static inequalities that are added to [MP] to remove a large number of symmetric solutions. We start by defining a set of operations on assignments, under which the optimal makespan is preserved. We then present the inequalities that break the symmetry defined by these operations. And finally, we present a related constraint to strengthen the LP relaxation of [MP]. Experiments on the effectiveness of these constraints are given in Section 5.

Given an assignment z, we define the following operations on z: *shift*, *stretch* and *shrink*. The shift operation takes a non-overlapping subassignment and shift it left or right without creating new overlaps. The stretch operation takes the assignment to an event and duplicates it to the right (left) and shifts subsequent assignments to the right (left). The shrink operation, which is the inverse of stretch, removes a duplicate assignment from two consecutive events and "shrink" the rest of the assignments. Figure 2 gives some examples of these operations. Assignment 2(a) is used as the base. It has two non-overlapping subassignments, one involving vessels 1, 3 and 4, and another, 2, 5, and 6. Assignment 2(b) is obtained by shifting these subassignments around. Assignment 2(c) is obtained by taking 2(b) and duplicating some events. Event 11 is duplicated twice to the left, event 5 once to the right and event 2 once to the left. And finally, assignment 2(d) is obtained by removing duplicate events in 2(c). Note that these operations define equivalent classes, where two assignments are in the same class if one can be reached from the other with a series of these operations. We have the following property.

Proposition 2. *Let z and z' be two assignments of activities to events. If z and z' belong to the same equivalent class defined by the operations: shift, stretch and shrink, then the subproblem* [PSP] *return the same objective value for both.*

Since assignments in the same class are symmetric to each other, we only need to consider one assignment from each class during branch-and-bound. To do this, we add the following two global constraints to [MP]:

$$\forall e \in \mathcal{E} \backslash \{nm\}: \sum_{i=1}^{n}\sum_{j=1}^{m} z_{e+1}^{ij} \leq n \cdot \sum_{i=1}^{n}\sum_{j=1}^{m} z_{e}^{ij}, \quad (26)$$

$$\forall e \in \{0,\ldots,nm\}: \sum_{i=1}^{n}\sum_{j=1}^{m} z_{e}^{ij} \oplus z_{e+1}^{ij} \geq \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m} z_{e}^{ij}, \quad (27)$$

where $\oplus$ is the exclusive-or operation. Constraint (26) ensures a left-shifted assignment, where an event has zero assignment only if subsequent events have zero assignment as well. The same effect can be obtained by ordering the variables during search, which is a common technique in symmetry breaking. Constraint (27) ensures that no consecutive events have the same assignment except the zero assignment. Together, they make only the "compact" assignment from each class feasible. The compact assignment is one that is left-shifted and can't be shrunk further. In the examples of Figure 2, only assignment 2(a) is feasible given these constraints.

The left term of Constraint (27) has an interesting interpretation. For any event e, its value denotes the number of activities that start or end at $e+1$. Given this, we can add the following constraint to [MP] to strengthen its LP relaxation:

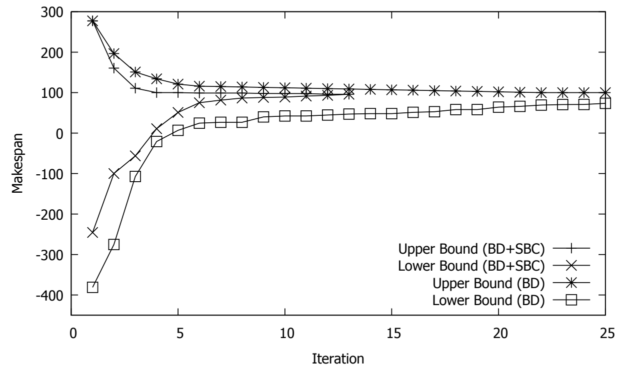
$$\sum_{e=0}^{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} z_{e}^{ij} \oplus z_{e+1}^{ij} = 2nm, \quad (28)$$

which ensures that the total number of activity-start and end in all events equals to twice the number of activities. It is a valid equality cut because it doesn't change the feasible region of [MP], but it reduces that of its LP relaxation.

Proposition 3. *Equality* (28) *is a valid and useful cut for the master problem* [MP].

5 Experimental Results

We test our approach on synthetic and real instances. For synthetic benchmarks, we measure the effectiveness of symmetry breaking constraints (SBC) proposed in Section 4 on small to medium sized instances. For each instance, we generate a random connected undirected graph with a fixed number of vertices (#V) and edges (#E), and varying number of vessels (#*Vessels*). For each vessel, we generate a random path of a fixed length through the graph. The capacity of each edge is uniformly generated between 1-3, and the time to traverse an edge is sampled uniformly from [5sec, 10sec]. The start time of each vessel is uniformly generated between 0-5 seconds. We measure the optimality of the Benders decomposition with and without the symmetry breaking constraints. The results are shown in Figures 3(a,b). For each row in figure 3(b), 100 random instances are generated, and optimality



(a) Lower vs upper bounds at different iterations

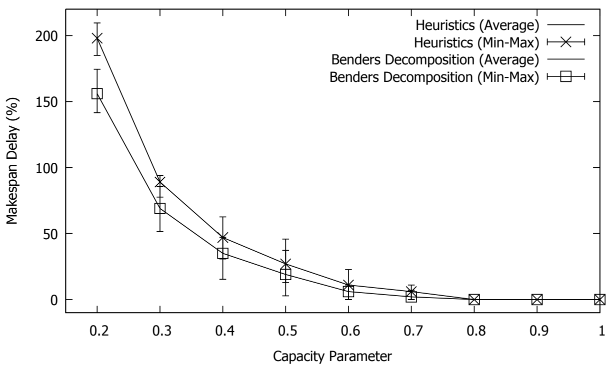
(#V,#E)	#Vessels	#Activities	BD+SBC	BD
(10,12)	10	5	100%	74%
(10,12)	**15**	**5**	**100%**	**73%**
(10,12)	20	5	100%	68%
(10,12)	25	5	100%	59%
(10,12)	30	5	100%	46%

(b) Optimality at iteration 100 with varying vessels

Figure 3: Synthetics benchmarks: Optimality of BD+SBC vs BD

gaps are measured at iteration 100. The average optimality over the 100 instances are given in the last two columns. As shown, with symmetry breaking constraints (BD+SBC), Benders decomposition is able to achieve convergence for all instances within 100 iterations, while the standard version (BD) exhibits a degenerate behavior. Figure 3(a) shows the gaps between upper and lower bounds of BD+SBC and BD at different iterations for one sample instance with 15 vessels.

For real world dataset, we use 55-day historical AIS data of vessels moving in Singapore waters from March 2nd to April 25th, 2014, consisting of over 4 millions record. An AIS record contains, among other things, a timestamp, vessel unique ID, lat-long position, speed over ground, direction and navigation status (e.g., at anchor, not under command). An AIS transponder on-board sends a record every 2-10 seconds while the vessel is underway, and every 3 minutes at anchor. All power-driven vessels within Singapore waters are required by law to have an approved transponder on board. In this work, we consider only tankers and cargos, which are the largest vessel types causing hotspots. From the records of each day, we perform the following to create scheduling instances. We first divide the day into 24 1-hour periods, and group the vessels according to these periods. A vessel belongs to a period if its first record on that day falls within that period. Then, from the vessel set in one period, we create an instance, where the activities of a vessel are inferred from its records. The recorded length of an activity becomes the minimum duration in the instance, and the maximum is set to twice the minimum (i.e., a vessel can go twice as slow). The earliest release time of a vessel is the timestamp of its first record on that day. Resource capacity and requirement are computed as follows. We identify the size (diameter) of the smallest tankers/cargos u, and that becomes 1 unit of resource. A vessel with the size of approximately $2u$, for ex-



(a) Plot for the busiest day

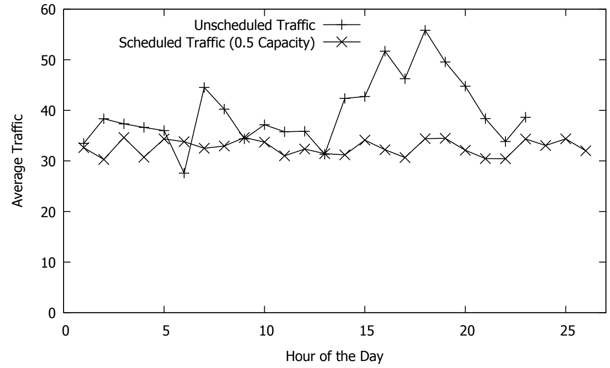
Date (2014)	BD+SBC (% Delay) 0.5C	0.4C	0.3C	Heuristics (% Delay) 0.5C	0.4C	0.3C
03/02	19	33	69	19	42	86
03/03	21	30	63	28	39	80
03/04	18	39	72	27	48	89
03/05	16	38	71	24	47	88
03/06	14	33	71	27	42	88
03/07	14	30	67	24	39	84
03/08	16	37	66	24	46	83
03/09	19	35	64	21	44	81
03/10	19	39	70	19	48	87
03/11	14	37	67	22	46	84
03/12	16	38	67	25	47	84
03/13	**19**	**35**	**69**	**24**	**44**	**86**
03/14	19	39	70	22	48	87
03/15	17	32	63	24	41	80
03/16	14	32	63	23	41	80

(b) Results for the first 15 days

Figure 4: Capacity vs delay tradeoff

ample, will consume 2 units of space resources. A zone's capacity is set to the maximum recorded units, at any timepoint, over all the records. A rolling-horizon approach is then used to reschedule the vessels in a day, where we start by solving the first-period instance. The second-period instance is then solved by fixing the resulting schedule from the first instance and so on.

The results are shown in Figures 4 and 5. In Figure 4(b), we show the improvements of Benders decomposition over the initial heuristic solutions for different capacity levels. The initial solutions are obtained using a greedy approach that schedule the vessels one at a time, using minimum incurred delay as criteria. The capacity level determines the reduction in the zones' capacities. E.g., at 0.5C, capacities of all the zones are reduced by half with minimum 1 unit. Reduction in capacity automatically reduces the congestion in zones. Figure 4(b) shows the average percentage of delay over 24 periods for different capacity levels for the first 15 days (other 40 days show similar trends and are omitted). It shows our BD+SBC approach is able to improve on the heuristic solution significantly across all the settings, with pronounced improvements over low capacity levels (0.3C). Figure 4(a) shows the plot for the busiest day (March 13th). An important observation is that delays are minimal until 0.6C capacity, which implies that schedule adjustment has the potential



(a) Plot for the busiest day

Date (2014)	Scheduled Traffic Min	Max	SD	Unscheduled Traffic Min	Max	SD
03/13 - Thu	**30.09**	**34.84**	**1.57**	**27.6**	**55.83**	**6.53**
03/03 - Mon	28.52	34.45	1.61	25.85	55.03	5.8
03/21 - Fri	27.8	34.29	1.78	26.24	53.95	6.15
04/16 - Wed	28.38	32.75	1.25	25.81	54.27	6.7
04/17 - Thu	27.61	30.06	1.54	23.77	52.25	5.82

(b) Results for the busiest 5 days

Figure 5: Comparison of traffic intensities.

to significantly reduce congestion with minimal resulting delays.

In Figures 5(a, b), we show how rescheduling affects the traffic intensity in a day. Here the traffic intensity is measured as the number of tankers/cargos within the planning area at any point of time. Figure 5(b) shows the min, max and standard deviation (SD) of traffic intensity for the 5 busiest days. The scheduled traffic is with half capacity (0.5C). It shows that by rescheduling, congestion can be significantly reduced. Significantly lower standard deviation for the scheduled traffic shows that peak traffic intensity is flattened out, resulting in congestion reduction. Figure 5(a) shows the plot of traffic intensities for the busiest day, comparing unscheduled with scheduled traffic. This figure further shows that the peak traffic intensity is flattened out over the entire day, thus reducing congestion.

6 Summary

In this paper, we addressed the problem of mitigating congestion in a busy water area. Our main contribution is the formulation of the vessel scheduling problem as a variant of the RCPSP. Our maritime domain constraints require that activity durations, which translate into average vessel speeds, also become decision variables unlike standard RCPSPs. For scalability, we developed a Benders decomposition approach to solve the resulting math program. Empirical results on synthetic instances, and a real world dataset consisting of vessel records over a 55-day period in Singapore waters, one of the busiest port in the world, confirm that intelligent scheduling adjustments can result in significant peak traffic reduction.

References

[Alvarez-Valdés and Tamarit, 1993] Ramón Alvarez-Valdés and José Manuel Tamarit. The project scheduling polyhedron: Dimension, facets and lifting theorems. *European Journal of Operational Research*, 67(12):204–220, 1993.

[Benders, 1962] Jacques F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, 1962.

[Cordeau *et al.*, 2001] Jean-François Cordeau, Goran Stojkovic, François Soumis, and Jacques Desrosiers. Benders decomposition for simultaneous aircraft routing and crew scheduling. *Transportation Science*, 35(4):375–388, 2001.

[Gent *et al.*, 2006] Ian P. Gent, Karen E. Petrie, and Jean-François Puget. Symmetry in constraint programming. In Francesca Rossi, Peter van Beek, and Toby Walsh, editors, *Handbook of Constraint Programming*, pages 329–376. Elsevier, 2006.

[Kolisch and Hartmann, 1999] Rainer Kolisch and Sönke Hartmann. Heuristic algorithms for the resource-constrained project scheduling problem: Classification and computational analysis. In *Project Scheduling*, pages 147–178. Springer US, 1999.

[Koné *et al.*, 2011] Oumar Koné, Christian Artigues, Pierre Lopez, and Marcel Mongeau. Event-based MILP models for resource-constrained project scheduling problems. *Computers & Operations Research*, 38(1):3–13, 2011.

[Magnanti and Wong, 1981] Thomas L. Magnanti and Richard T. Wong. Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria. *Operations Research*, 29(3):464–484, 1981.

[Margot, 2010] François Margot. Symmetry in integer linear programming. In *50 Years of Integer Programming 1959-2008*, pages 647–686. Springer Berlin Heidelberg, 2010.

[Marín and Jaramillo, 2009] Ángel Marín and Patricia Jaramillo. Urban rapid transit network design: Accelerated benders decomposition. *Annals of Operations Research*, 169(1):35–53, 2009.

[Maritime and Port Authority of Singapore, 2016] Maritime and Port Authority of Singapore. *Port marine circular No. 10*, 2016.

[Merciera *et al.*, 2005] Anne Merciera, Jean-François Cordeau, and François Soumis. A computational study of Benders decomposition for the integrated aircraft routing and crew scheduling problem. *Computers & Operations Research*, 32(6):1451–1476, 2005.

[Mingozzi *et al.*, 1998] Aristide Mingozzi, Vittorio Maniezzo, Salvatore Ricciardelli, and Lucio Bianco. An exact algorithm for the resource-constrained project scheduling problem based on a new mathematical formulation. *Management Science*, 44(5):714–729, 1998.

[Papadakos, 2009] Nikolaos Papadakos. Integrated airline scheduling. *Computers & Operations Research*, 36(1):176–195, 2009.

[Pietrzykowski and Uriasz, 2009] Zbigniew Pietrzykowski and Janusz Uriasz. The ship domain: A criterion of navigational safety assessment in an open sea area. *Journal of Navigation*, 62(1):93–108, 2009.

[Qu *et al.*, 2011] Xiaobo Qu, Qiang Meng, and Li Suyi. Ship collision risk assessment for the Singapore Strait. *Accident Analysis & Prevention*, 43(6):2030–2036, 2011.

[Romanski and Hentenryck, 2016] Julia Romanski and Pascal Van Hentenryck. Benders decomposition for large-scale prescriptive evacuations. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, pages 3894–3900, 2016.

[Schutt *et al.*, 2013a] Andreas Schutt, Thibaut Feydy, and Peter J. Stuckey. Explaining time-table-edge-finding propagation for the cumulative resource constraint. In *Proceedings of the 10th CPAIOR international conference*, pages 234–250, 2013.

[Schutt *et al.*, 2013b] Andreas Schutt, Thibaut Feydy, Peter J. Stuckey, and Mark G. Wallace. Solving RCPSP/max by lazy clause generation. *Journal of Scheduling*, 16(3):273–289, 2013.

[van Dorp and Merrick, 2011] J. René van Dorp and Jason R. W. Merrick. On a risk management analysis of oil spill risk using maritime transportation system simulation. *Annals of Operations Research*, 187(1):249–277, 2011.

[Vilím, 2011] Petr Vilím. Timetable edge finding filtering algorithm for discrete cumulative resources. In *Proceedings of the 8th CPAIOR international conference*, pages 230–245, 2011.

[Walsh, 2012] Toby Walsh. Symmetry breaking constraints: Recent results. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*, pages 2192–2198, 2012.

[Witherby Publishing Group, 2015] Witherby Publishing Group. *Passage planning guide: Malacca and Singapore straits*. Witherby Seamanship International, 2015 edition, 2015.