

Scheduling Vessels and Container-Yard Operations with Conflicting Objectives

Endre Boros, Lei Lei, Yao Zhao, and Hong Zhong⁺
Rutcor and Rutgers Business School, Rutgers University
⁺SUNY– Oneonta, State University of New York

We consider the problem of coordinating the operations of two supply chain partners: a foreign shipping company and a domestic port. The two partners have conflicting business objectives, and the issue is to determine the *optimal cycle time*, by which the shipping company removes the empty containers from the domestic port, so that the joint profit of the two partners is maximized. The domestic port prefers a shorter cycle time to mitigate its empty container accumulation and land use problems, while the shipping company wishes a longer cycle time to save its expensive vessel capacities. We propose an iterative procedure to search for this optimal cycle time. In each iteration, a candidate cycle time is evaluated by solving a deterministic vessel scheduling problem and a stochastic container-yard capacity optimization problem. We prove the properties of the vessel scheduling problem, derive the optimality condition under which the vessel scheduling problem can be decomposed, and show that the profit function of the domestic port is convex and thus the optimal container-yard capacity can be determined efficiently. Empirical observations on the algorithm computational performance collected from over 300 randomly generated test cases under various problem settings are reported.

Key words: Supply chain collaboration, vessel scheduling, container-yard management.

1. Introduction

It is known that the collaboration among transportation partners has a major impact on the performance of a supply chain (Carter and Ferrin, 1995). Achieving this collaboration in practice, however, is not always an easy task. Due to many practical reasons (e.g., expensive vessel waiting time for loading empty containers, ever increasing trade imbalances, and low cost of producing new containers in Asia, etc), the accumulation of (used and often damaged/unwanted) empty containers, and consequently the land use, has been a major issue faced by many US ports (Boile, et al, 2004, VTC/HDR, 2004). This inefficiency has in turn increased the shipping and importation cost of many domestic shippers who outsourced their productions to low cost foreign countries.

In this study, we develop mathematical models and algorithms to support a collaborative planning and scheduling of container operations for supply chain logistics partners. In particular, we consider a hypothesized model (see Figure 1) involving a *domestic port* (port P_1) that operates a container-yard for empty containers and a *foreign shipping company* that is home-based at a foreign port (port P_0) and owns the empty

containers accumulated at port P_1 . To relieve the empty container accumulation at port P_1 , the shipping company is willing to allocate a portion of the vessels' return trips to bring back its own empty containers. However, by offering this, the vessel capacity for profitable commercial cargos on the return trips is also reduced. The *domestic port* receives random arrivals of empty containers from inland customers (shippers) and wishes to reduce its overall cost of managing the container-yard. The issue of the coordination is the *cycle time* (denoted as T throughout this paper) between two consecutive visits of the vessels to the container-yard to remove the empty containers. From the domestic port's point of view, the more frequently the shipping company visits the container-yard (i.e., the smaller the T value), the lower the operational cost and the less pressure imposed by container accumulation will be. On the other hand, from the shipping company's point of view, the less frequent visit to the container-yard, the more vessel capacity will be available for its profit-making business trips. Our focus of this study is thus on identifying the optimal cycle time, T^* , that *maximizes the joint profit of the two supply chain partners* over a given planning horizon.

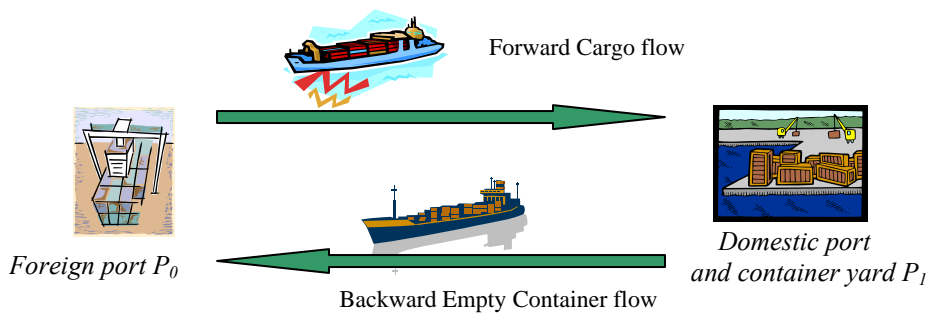


Figure 1. An illustrative graph for the port to port operations.

It should be pointed out that the port and foreign vessel collaboration problem in practice is very complicated. It contains many detailed operational components and government regulation issues that we are not able to cover in our mathematical programming formulation and optimization. As the amount of U.S. business outsourced abroad exceeds the level of \$4 trillions (Logistics Today, 2006), as the ratio of importation to exportation gets closer to 2:1, and as the accumulation of empty containers at ports exceeded 2.5 million tractor-equivalent units occupying thousands of acres of expensive land like NJ/NY area, repositioning empty containers to their origins is no longer a cost/profit issue but an obligation under the trade agreements. Since our focus of this study is to develop new theoretical and algorithmic results for the related vessel scheduling and the vessel-port collaboration problem, our hypothesized shipping model is a simplified one. Therefore, our results are limited to only serving as a decision support tool for the related optimization problems. Nevertheless, this study proves theoretically two important properties of the vessel scheduling problem and addresses a variation of

the vessel-port collaboration problem for which we are not aware of any available optimization result in the literature of scheduling.

Our study was partly motivated by the practical issues faced by a primary Asian shipping company in its collaboration with a major U.S. port on repositioning the empty containers. Most assumptions below (although simplified) are based on the real life container operations of this company, upon which we shall define our vessel scheduling and the container-yard capacity optimization problems over a given planning horizon H .

Assumptions on the foreign shipping company's operations

The shipping company is home-based at a foreign port, denoted as P_0 , and owns a fleet of *identical* vessels, V_0 . Each vessel has a loading capacity, c , measuring the maximum number of standard containers it may carry. The vessels in set V_0 are divided into two groups for the given planning horizon, those transporting empty containers on their return trips are called *backhaul* vessels, and those assigned to transport commercial imports/exports cargos on both forward and return trips are called *business* vessels. The backhaul vessels transport cargo containers on their forward trips (thus make a profit) to port P_1 , while serve as express liners reserved for empty containers on their return trips back to port P_0 .

In the hypothesized shipping problem upon which we conduct our mathematical analysis, we assume that there is always a sufficient amount of customer orders that need to be shipped from P_0 to P_1 . Let K be the set of customer orders to be considered for transportation by the backhaul vessels during a given planning horizon H . All customer orders are released at port P_0 and are heading for port P_1 . Order i , $i \in K$, is released at time r_i and becomes available for shipping after that. Starting from r_i , the order can wait at most for π time units or the customer will withdraw the order and seek for a different shipping company. Associated with each order i , there is also a shipping quantity (in terms of the number of containers used), d_i , a profit α_i , and a receiving time window $[a_i, b_i]$ at the domestic port P_1 . A holding cost, $\$h$, occurs if order i arrives at P_1 before a_i . The delivery becomes infeasible if the order arrives after b_i . In addition, vessels' waiting time at port P_1 results in a penalty, $\$q/(\text{unit time, vessel})$, and cannot exceed a maximum waiting time limit denoted as W^{\max} . Let $2 \cdot \tau$ denote the round trip time between the foreign port P_0 and the domestic port P_1 . Figure 2 illustrates this shipping process associated with an individual backhaul vessel trip.

For any cycle time T agreed by the two supply chain partners, the shipping company has to commit $|V| = \lceil 2\tau / T \rceil$ backhaul vessels for a given planning horizon H ,

where $V \in V_0$ denotes the set of backhaul vessels needed to meet the cycle time T . We assume that each backhaul vessel is assigned to perform $l_0 \geq 1$ round trips over the given planning horizon. This leads to a total of $|V| \cdot l_0$ vessel visits to the container-yard at port P_1 to be scheduled. Let $\Lambda = \{T, 2T, \dots, |V| \cdot l_0 \cdot T\}$ be the set of fixed visiting time points, $|\Lambda| = |V| \cdot l_0$, at exactly each of which a backhaul vessel carrying empty containers departs port P_1 .

In this study, we focus on two optimization problems: *scheduling the forward trips of backhaul vessels* and *coordinating the operations of backhaul vessels and the domestic port to return empty containers to the foreign port*. To make the resulting vessel scheduling problem manageable as a mixed integer program in the analysis, we have omitted many details of the real life shipping process, such as the vessel loading/unloading operations at ports, positioning the containers onboard ship to facilitate an efficient discharge at the destination port, seasonality of the demand, possibility to halt loading empty containers in the case a vessel is behind the schedule, and port labor idle time, etc. Also, to focus on the collaboration of the backhaul vessels and the domestic port, we simplify in this study the operational optimization of business vessels. Let β be the average profit generated by a business vessel over the planning horizon H . The profit of the shipping company for any given cycle time T is thus defined by

$$\xi^F(T) = \beta \cdot (|V_0| - |V|) + G(T),$$

where $G(T)$ stands for the total profit generated by the backhaul vessels in set V , and

$$G(T) = \text{Profit of forward trips} - (\text{order holding cost at } P_1 + \text{vessel waiting cost at } P_1).$$

We assume that the transportation of empty containers does not generate profit for the

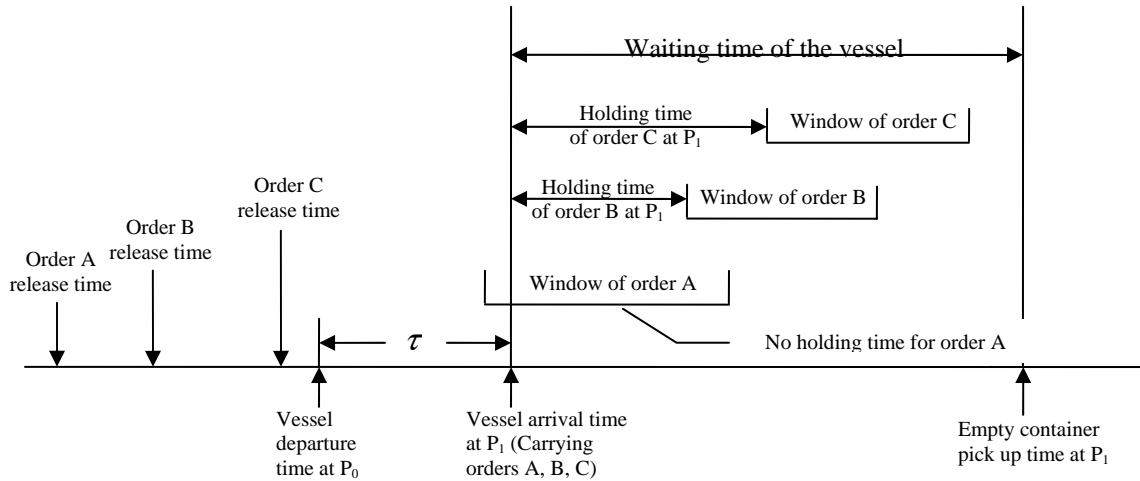


Figure 2. An illustration of a vessel waiting time at port P_1 .

shipping company, which reflects the practice that transporting the empty containers back to the origin is more an obligation issue than a profit-making business. Backhaul vessels generate profit only by transporting cargo containers in its forward trips from P_0 to P_1 . Clearly, the function $\beta \cdot (|V^0| - |V|)$ increases as the number of backhaul vessels, $|V|$, decreases, or equivalently as the cycle time T increases.

Assumptions on domestic port's operations

The container-yard at the domestic port, port P_1 , receives a service fee (revenue) of $\$b^c$ for every *vessel-load* containers it handled (i.e., arrived at port P_1 as cargo containers and later returned by the in-land domestic shippers as empty containers to the container-yard at port P_1), where a unit of vessel-load is measured by c which stands for the maximum loading capacity of a vessel. Port P_1 pays for a total of $\$b^s \cdot Q$ for operating a container-yard of capacity Q (in the unit of c) over the given planning horizon^[a]. Let X stand for the number of empty containers accumulated at the container-yard when a backhaul vessel arrives. Thus, X is a random variable (since the empty containers are returned by the customers at random times). The assigned backhaul vessel can only remove $\text{Min}\{X, c\}$ empty containers at the visit. It is assumed that the amount of empty containers exceeding the container-yard capacity Q is overflowed to the other domestic ports (or say *blocked*). Each unit of overflowed containers costs $\$\phi$. Given the return rate of empty containers, λ , to the container-yard at port P_1 , the domestic port's profit is thus defined as

$$\xi^D(T) = b^c \cdot \lambda \cdot H - b^s \cdot Q - \phi \cdot E(\delta(Q, T)),$$

where all quantities are in units of c , $\delta(Q, T)$ stands for total quantity of containers blocked over the planning horizon H , and $E(\delta(Q, T))$ is the expected value of $\delta(Q, T)$.

Our problem is to determine the optimal cycle time, T^* , the vessel operation schedule that accommodates T^* , and the optimal capacity Q^* of the container-yard at P_1 so that the joint profit, $\xi^{total} = \xi^F(T^*) + \xi^D(T^*)$, is maximized. We shall call this problem the *coordination* problem, or problem **P**, in the rest of this paper. This coordination problem imposes several challenges. First, we are facing a challenging deterministic vessel scheduling problem, which includes some well known NP-hard sub-problems. Second, to determine the optimal container-yard capacity under random arrivals, we need to solve a stochastic optimization problem. Neither of these problems alone is trivial.

^[a]. Since the length of the planning horizon is fixed, parameter b^s is treated as a constant here.

In this study, we propose an iterative approach to solve this coordination problem. In each iteration, a candidate cycle time (T) is evaluated by solving a deterministic vessel scheduling problem and a stochastic container-yard capacity optimization problem. To improve the efficiency of the search, we analyze the properties of the associated vessel scheduling problem which lead to an efficient local optimization algorithm. We then prove that the profit function of the domestic port, $\xi^F(T)$, is convex, and thus an efficient search procedure exists for determining the optimal container-yard capacity. The optimal cycle time T^* is finally determined after a linear search over a given range of candidates.

There has been an abundant literature on the vessel scheduling problem during the past few decades. Among those more recent works, Perakis (1996) proposed a model to find the optimal fleet size and the associated liner routes. By generating a number of candidate routes for different vessels, the problem was solved as a linear programming model. Fagerholt (1999) presented a model to determine the optimal fleet size and their weekly liner routes. The problem was solved by employing a set partitioning approach as a multi-trip vehicle routing problem. Bendall and Stent (2001) proposed a model to determine the optimal fleet configuration and associated fleet deployment plan in a container vessel hub and spoke application. Sambracos et al. (2004) solved the coastal freight shipping problem via two phases: the strategic planning phase by a linear programming model to determine the fleet size; and the operational scheduling phase via solving a vehicle-routing type problem. Fagerholt (2004) presented a decision support system, called TurboRouter, that applies rule-based heuristics to assign cargos to vessels under various practical constraints. Chen et al (2006) proved the solvability of a special case of the bi-directional container vessel scheduling problem and used that property to develop a heuristic scheduling algorithm. Two comprehensive reviews of the results in this area can be found in Ronen (1993) and Christiansen et al. (2004). To our knowledge, however, vessel scheduling with conflicting objectives has not received much attention in the literature.

The container yard capacity problem is related to that of a finite Dam, see, e.g., Moran (1954), Prabhu (1965) and Avi-Itzhak and Ben-Tuvia (1963). It is also related to the problem of queuing systems with limited waiting room, such as the problem of call center blocking. Harel (1990) and Jager and Doorn (1986) prove various convexity properties for the Erlang loss formula in service rate, arrival rate and the number of servers. The Erlang loss formula is based on M/G/x/x queueing model which differs from our model with one server and finite buffer. More recently, Kumaran, et al. (2003) considers fluid queues with continuous-state space, and proves that the fraction of overflow of a queueing system with finite buffer and constant service time is convex in the buffer size and service rate. In contrast, we consider discrete state space where each container takes certain amount of buffer capacity. We provide a novel and simple proof

technique based on sample path analysis to show the convexity result. Other related convexity results in queueing-inventory problems can be found in Shanthikumar and Yao (1991) and Li and Zhang (2000). An important feature that distinguishes the container-yard problem analyzed in this paper from those commonly considered in the literature is the cost associated with the overflow, its trade-off with the cost of container-yard capacity, and the need of having both in one analysis.

The remaining part of this paper is organized as follows. In Section 2, we analyze the properties of the vessel scheduling problem with a given cycle time T . These properties allow us to develop a fast local optimization algorithm for solving the backhaul vessel scheduling problem. We also discuss the condition under which the heuristic indeed achieves the optimal solution. In Section 3, we prove the convexity of the profit function of the container-yard under a given cycle time T , and propose an incremental search procedure to determine the optimal container-yard capacity with respect to T . Based on the analytical results in Sections 2 and 3, we then propose an iterative search procedure to solve the coordination problem \mathbf{P} . In Section 4, we report on the empirical performance of the search algorithms, and finally in Section 5, we conclude the study and discuss future extensions of this work.

2. The vessel scheduling problem with a given cycle time T

For any given cycle time, T , let $\Lambda = \{T, 2T, \dots, |V| \cdot l_0 \cdot T\}$ be the sequence of fixed (empty container) pickup times at the container-yard, where each pickup time, $s_{v,n} \in \Lambda$, is associated with a particular backhaul vessel $v \in V$ and one of its trips, n , $1 \leq n \leq l_0$. Since we ignore the loading/unloading time here, $s_{v,n}$ is also the departure time of vessel v from P_1 during its n th trip over the planning horizon. Let $\mathbf{P}(T)$ denote the respective backhaul vessel scheduling problem, and let

- $Z_{v,n,k}$ Binary variables, $Z_{v,n,k} = 1$ if order k is carried by v , $v \in V$, during its n th trip, $k \in K$, $1 \leq n \leq l_0$;
- e_k The earliness of the arrival time of order k at domestic port P_1 ;
- $W_{v,n}$ The waiting time of vessel v during its n th trip at domestic port P_1 ;
- $t_{v,n}$ The departure time of the n th trip of vessel v from P_0 , $v \in V$, $1 \leq n \leq l_0$.

Problem $\mathbf{P}(T)$:

$$\text{Max. } \xi^F(T) = \beta(|V^0| - |V|) + \sum_v \sum_n \sum_k \alpha_k \cdot Z_{v,n,k} - \sum_v \sum_n q \cdot W_{v,n} - \sum_k h \cdot d_k \cdot e_k \quad (1)$$

s.t.

Constraints on the order-vessel assignment

$$\sum_v \sum_n Z_{v,n,k} \leq 1 \quad \forall k \in K \quad (2)$$

The vessel capacity constraints

$$\sum_{k \in K} Z_{v,n,k} d_k \leq c \quad \forall v \in V, \forall n = 1, 2, \dots, l_0 \quad (3)$$

Constraints on the departure times of consecutive vessel trips

$$t_{v,n} + 2\tau + W_{v,n} \leq t_{v,n+1} \quad \forall v \in V, \forall n = 1, 2, \dots, l_0 \quad (4)$$

Lower bound on the departure time of each vessel trip

$$t_{v,n} \geq r_k \cdot Z_{v,n,k}, \quad \forall v \in V, \forall n = 1, 2, \dots, l_0, \forall k \in K \quad (5)$$

$$t_{v,n} + \tau + e_k \geq a_k \cdot Z_{v,n,k}, \quad \forall v \in V, \forall n = 1, 2, \dots, l_0, \forall k \in K \quad (6)$$

Upper bound on the departure time of each vessel trip

$$t_{v,n} \leq (r_k + \pi) + M \cdot (1 - Z_{v,n,k}), \quad \forall v \in V, \forall n = 1, 2, \dots, l_0, \forall k \in K \quad (7)$$

$$t_{v,n} + \tau \leq b_k + M \cdot (1 - Z_{v,n,k}), \quad \forall v \in V, \forall n = 1, 2, \dots, l_0, \forall k \in K \quad (8)$$

Constraints on the empty-container pickup time

$$t_{v,n} + \tau + W_{v,n} = s_{v,n} \quad \forall v \in V, \forall n = 1, 2, \dots, l_0 \quad (9)$$

Constraints on the vessel waiting time at the domestic port

$$W_{v,n} \leq W^{\max}, \quad \forall v \in V, n = 1, 2, \dots, l_0 \quad (10)$$

Other constraints

$Z_{v,n,k} \in \{0,1\}$, $\forall v \in V, \forall k \in K, 1 \leq n \leq l_0$, and all the other variables are non-negative.

Note that, for any given cycle time T , $|V|$ is a constant and so is the total profit by the business vessels, $\beta \cdot (|V^0| - |V|)$. Therefore, the objective function of $\mathbf{P}(\mathbf{T})$ is equivalent to

$$\text{Max. } G(T) = \sum_v \sum_n \sum_k \alpha_k \cdot Z_{v,n,k} - \sum_v \sum_n q \cdot W_{v,n} - \sum_k h \cdot d_k \cdot e_k \quad (1')$$

Problem $\mathbf{P}(\mathbf{T})$ is a generalized bin-packing problem and involves $|V| \cdot l_0 \cdot |K|$ binary variables. Solving $\mathbf{P}(\mathbf{T})$ directly could be very time consuming specially when the problem size is large. Therefore, we are interested in solving $\mathbf{P}(\mathbf{T})$ heuristically.

To construct such a heuristic vessel scheduling algorithm, let's first consider the following single-vessel and single-trip scheduling problem. Let v denote the given vessel that is about to perform its n th trip, and $s_{v,n}$ be the respective pickup time at P_1 for the empty containers. Since $s_{v,n}$ is fixed, the time interval within which vessel v may depart from P_0 is known and is given as

$$[s_{v,n} - \tau - W^{\max}, s_{v,n} - \tau].$$

Therefore, the set of orders that can be feasibly considered for the transportation by vessel v during its n th (forward) trip to P_1 , where $v \in V, 1 \leq n \leq l_0$, must satisfy the following conditions:

$$\Phi_{v,n} = \left\{ k \in K \left| \begin{array}{l} r_k + \pi \geq s_{v,n} - \tau - W^{\max} \\ r_k \leq s_{v,n} - \tau \\ s_{v,n} - W^{\max} \leq b_k \end{array} \right. \right. \quad \left. \begin{array}{l} (11) \\ (12) \\ (13) \end{array} \right\}$$

Let $Z_k, k \in \Phi_{v,n}$, be binary variables, and $Z_k = 1$ iff order k is chosen to be delivered by vessel v during the trip. Let W be the vessel waiting time at P_1 and t be the departure time of the vessel from P_0 . Then, this single-vessel single-trip problem, denoted as $\mathbf{P}(v,n)$, can be stated as follows.

$$\mathbf{P}(v,n): \quad \text{Max. } G(v,n) = \sum_{\forall k \in \Phi_{v,n}} \alpha_k \cdot Z_k - \sum_{\forall k \in \Phi_{v,n}} h \cdot d_k \cdot e_k - q \cdot W \quad (1a)$$

s.t.

The vessel capacity constraint

$$\sum_{\forall k \in \Phi_{v,n}} d_k \cdot Z_k \leq c \quad (3a)$$

The lower bound on vessel departure time from port P_0

$$t - r_k Z_k \geq 0 \quad \forall k \in \Phi_{v,n} \quad (5a)$$

$$t + \tau + e_k \geq a_k \cdot Z_k \quad \forall k \in \Phi_{v,n} \quad (6a)$$

The upper bound on vessel departure time from port P_0

$$t \leq r_k + \pi + M(1 - Z_k) \quad \forall k \in \Phi_{v,n} \quad (7a)$$

$$t + \tau \leq b_k + M(1 - Z_k) \quad \forall k \in \Phi_{v,n} \quad (8a)$$

Constraint on the empty-container pickup time

$$t + \tau + W = s_{v,n} \quad (9a)$$

Others

$$Z_k \in \{0,1\}, \quad t, W, e_k \geq 0, \quad \forall k \in \Phi_{v,n}$$

As we can see, problem $\mathbf{P}(v,n)$ is a knapsack problem with additional constraints but no more than $|K|$ binary variables. It is much easier to solve than problem $\mathbf{P}(T)$. Now we prove the following result.

Proposition 1. For any given problem $\mathbf{P}(\mathbf{T})$, if condition

$$\pi < T - W^{\max} \quad (14)$$

holds, then $\mathbf{P}(\mathbf{T})$ can be decomposed into $|V| \cdot l_0$ independent problems, $\mathbf{P}(\mathbf{v}, \mathbf{n})$, with $v=1, 2, \dots, |V|$, $n=1, 2, \dots, l_0$.

Proof: Since all the vessel departure times from domestic port P_1 , $s_{v,n}$, $\forall v \in V$, $n=1, 2, \dots, l_0$, are fixed, the vessel departure times from port P_0 are independent of each other. Therefore, the claim is proved if we can show

$$\Phi_{v,n} \cap \Phi_{v^+,n^+} = \emptyset,$$

where $\Phi_{v,n}$ and Φ_{v^+,n^+} are the order sets to be evaluated for the shipment by two consecutive vessel trips, (v, n) and (v^+, n^+) , to the domestic port, respectively, and $s_{v,n} + T = s_{v^+,n^+}$, $1 \leq n, n^+ \leq l_0$, $v, v^+ \in V$. Consider two arbitrary orders, $k \in \Phi_{v,n}$, $k^+ \in \Phi_{v^+,n^+}$. We must have

$$r_k + \pi \leq s_{v,n} - \tau + \pi = s_{v^+,n^+} - T - \tau + \pi \quad \text{by (12)}$$

$$< s_{v^+,n^+} - T - \tau + T - W^{\max} \quad \text{by (14)}$$

$$= s_{v^+,n^+} - \tau - W^{\max} \leq r_{k^+} + \pi \quad \text{by (11)}$$

which implies $r_k < r_{k^+}$, for any $k \in \Phi_{v,n}$, and $k^+ \in \Phi_{v^+,n^+}$. Therefore, $\Phi_{v,n} \cap \Phi_{v^+,n^+} = \emptyset$ follows. \square

Proposition 2. If $\sum_{\forall k \in \Phi_{v,n}} d_k \leq c$, then problem $\mathbf{P}(\mathbf{v}, \mathbf{n})$ can be solved in $O(|\Phi_{v,n}|)$ time.

Proof: Note that the variable W can be expressed as $W = s_{v,n} - \tau - t$, and hence it can be eliminated from the formulation simply by assuming for variable t that

$$0 \leq t \leq s_{v,n} - \tau.$$

Consequently, the objective function $G(v, n)$ depends on t positively. We next observe that in the optimum we must have

$$e_k = \max\{0, a_k \cdot Z_k - \tau - t\}.$$

Thus orders $k \in \Phi_{v,n}$ for which $Z_k = 1$ in the optimal contribution $\alpha_k - h \cdot d_k \cdot e_k$ to the objective function, and since this contribution must

be non-negative (otherwise, we would increase the objective function by switching Z_k to 0), the following inequality holds:

$$t \geq a_k - \tau - \frac{\alpha_k}{h \cdot d_k}$$

Adding this to the other constraints of problem $\mathbf{P}(\mathbf{v}, \mathbf{n})$, we get that $Z_k = 1$ in the optimum for exactly those orders $k \in \Phi_{\mathbf{v}, \mathbf{n}}$ for which

$$\max\{r_k, a_k - \tau - \frac{\alpha_k}{h \cdot d_k}\} \leq t \leq \min\{r_k + \pi, b_k - \tau\} \quad (15)$$

holds. Moreover, since $G(\mathbf{v}, \mathbf{n})$ depends positively on t , we must have

$$t \in \Omega = \{ \min.\{r_k + \pi, b_k - \tau\} \mid k \in \Phi_{\mathbf{v}, \mathbf{n}} \}$$

at the optimum. Thus, problem $\mathbf{P}(\mathbf{v}, \mathbf{n})$ can be solved by trying all values $t \in \Omega$, and for each setting $Z_k = 1$ for exactly those orders $k \in \Phi_{\mathbf{v}, \mathbf{n}}$ for which (15) holds. \square

Let's remark that even if $\sum_{\forall k \in \Phi_{\mathbf{v}, \mathbf{n}}} d_k > c$, problem $\mathbf{P}(\mathbf{v}, \mathbf{n})$ can be reduced to $|\Omega| \leq |\Phi_{\mathbf{v}, \mathbf{n}}|$ binary knapsack problems, similarly to the proof above. Consequently, for any fixed $\varepsilon > 0$, we can find in polynomial time a solution to $\mathbf{P}(\mathbf{v}, \mathbf{n})$ which has no more than ε relative error (see Ibarra and Kim 1975).

Based on the above results, we propose the following heuristic, called **V-Scheduler(T)**, for solving $\mathbf{P}(\mathbf{T})$ with respect to each given cycle time T .

V-Scheduler(T)

Step 1: Construct sets $\Phi_{\mathbf{v}, \mathbf{n}}$ by (11)-(13) for all the vessels $\mathbf{v} \in V$ and all their trips;

Step 2. Solve the single-vessel single-trip problem $\mathbf{P}(\mathbf{v}, \mathbf{n})$ based on set $\Phi_{\mathbf{v}, \mathbf{n}}$ defined in

step 1, and return with $G(\mathbf{v}, \mathbf{n})$, for all $\mathbf{v} \in V$, $1 \leq \mathbf{n} \leq l_0$;

Step 3. Let $G(\mathbf{T}) = \sum_{\forall \mathbf{v}, \forall \mathbf{n}} G(\mathbf{v}, \mathbf{n})$.

As we can see from the analysis above, if $\pi < T - W^{\max}$, then **V-Scheduler(T)** finds the optimal solution to $\mathbf{P}(\mathbf{T})$. Otherwise, the optimality is not guaranteed. Nevertheless, we show in Section 4 that this heuristic has a promising empirical performance for all the test instances randomly generated in our study under various parameter settings. Also note that, in practice, condition $\pi < T - W^{\max}$ is not uncommon

since expensive vessel waiting cost penalizes larger values of W^{\max} , and shipping companies usually promise shorter order waiting times π to offer a competitive service level for its business customers.

3. The container yard capacity optimization problem

For any given empty container pick-up cycle time T , the problem faced by the domestic port, P_1 , is to determine its capacity, Q , so as to maximize the operational profit over the planning horizon. We define all the quantities in the unit of c . Since the arrival rate of empty containers, λ , to the container-yard is fixed, the expected revenue, $b^c \cdot \lambda \cdot H$, is a constant, where parameter b^c stands for the service fee per c containers passing through port P_1 . Therefore, maximizing the operational profit is equivalent to minimizing the operational costs. To formally define this problem, let

- $D(T)$ The total count of empty containers received by P_1 in the cycle of cycle time T . We assume that $D(T)$ follows a Poisson process with rate $\lambda \cdot T$.
- Q The capacity of the empty container yard (in the unit of c);
- I The quantity of empty containers (in the unit of c) in the container-yard at the beginning of a cycle;
- B The quantity of overflows (in the unit of c) in one cycle;
- $g(Q)$ The long-run average cost per unit of time, and

The objective function of the container-yard capacity optimization problem, for any given cycle time T , can now be defined formally as

$$g(Q) = b^s \cdot Q / H + \phi \cdot E(B) / T \quad (16)$$

In our analysis, we shall focus on the long-run average cost per unit of time, $g(Q)$, for the following reasons: first, the long-run average cost is independent of the random initial state. Second, if the initial state does not follow steady-state distribution, the underlying Markov chain converges quickly to the steady-state. Hence, given a relatively long planning horizon, the long-run average cost per time period is an adequate measure of the cost. The dynamics of the system are given by:

$$I_{n+1}(Q) = (\min\{I_n(Q) + D_n(T), Q\} - c)^+ \quad (17)$$

$$B_n(Q) = (I_n(Q) + D_n(T) - Q)^+ \quad (18)$$

where $a^+ = \max\{a, 0\}$. We refer to Prabhu (1965) for detailed discussion of the transition probability of I_n and the steady-state distribution of I and $E(B)$.

We now prove the convexity of the cost function $g(Q)$, as defined by (16). To start, let's first consider two consecutive levels of the capacity, Q and $Q+1$, and the corresponding $I(Q)$ and $B(Q)$. Proposition 3 below indicates that small changes in Q results in small changes in $I(Q)$ and $B(Q)$.

Proposition 3. *If $0 \leq I_0(Q) = I_0(Q+1) \leq Q$, then for all $n \geq 0$,*

$$0 \leq I_n(Q+1) - I_n(Q) \leq 1 \quad (19)$$

$$0 \leq B_n(Q) - B_n(Q+1) \leq 1 \quad (20)$$

Proof: The proof is by induction. At $n = 0$, inequality (19) holds by initial condition. For $n \geq 0$, we shall show that if $0 \leq I_n(Q+1) - I_n(Q) \leq 1$, then $0 \leq I_{n+1}(Q+1) - I_{n+1}(Q) \leq 1$ and $0 \leq B_n(Q) - B_n(Q+1) \leq 1$. To do so, note that by Eq. (17),

$$\begin{aligned} I_{n+1}(Q) &= (\min\{I_n(Q) + D_n(T), Q\} - c)^+ \\ I_{n+1}(Q+1) &= (\min\{I_n(Q+1) + D_n(T), Q+1\} - c)^+. \end{aligned}$$

Let's consider the following two cases, where in Case 1, $I_n(Q) = I_n(Q+1)$; in Case 2, $I_n(Q) + 1 = I_n(Q+1)$. It can be shown that in both cases, $I_{n+1}(Q+1) \geq I_{n+1}(Q)$ and $I_{n+1}(Q+1) - I_{n+1}(Q) \leq 1$. To prove inequality (20), note that by Eq. (18),

$$\begin{aligned} B_n(Q) &= (I_n(Q) + D_n(T) - Q)^+ \\ B_n(Q+1) &= (I_n(Q+1) + D_n(T) - Q - 1)^+. \end{aligned}$$

Let's consider the same two cases. In Case 1, $B_n(Q) \geq B_n(Q+1)$ and $B_n(Q) - B_n(Q+1) \leq 1$. In Case 2, $B_n(Q) = B_n(Q+1)$. The proof is now completed. \square

Remark. *Since the proof is based on sample path analysis, no probability law needs to be postulated on the demand process.*

Proposition 4. *$g(Q)$ is convex in Q for $Q \geq 1$.*

Proof: We only need to show that the steady-state $E(B)$ is convex in Q . To this end, consider $B(Q-1), B(Q)$ and $B(Q+1)$ for any $Q > 1$. Since the Markov chains are ergodic (see Prabhu 1965), we assume $I_0(Q-1) = I_0(Q) = I_0(Q+1)$ without loss of generality. For any $J > 0$, we show

$$\sum_{n=0}^J B_n(Q-1) - \sum_{n=0}^J B_n(Q) \geq \sum_{n=0}^J B_n(Q) - \sum_{n=0}^J B_n(Q+1) \quad (21)$$

By Proposition 3 and the initial condition, we have at most 4 cases in any cycle $n \geq 0$,

- Case 1: $I_n(Q+1) = I_n(Q) = I_n(Q-1)$
- Case 2: $I_n(Q+1) = I_n(Q) = I_n(Q-1) + 1$
- Case 3: $I_n(Q+1) = I_n(Q) + 1, I_n(Q) = I_n(Q-1) + 1$
- Case 4: $I_n(Q+1) = I_n(Q) + 1 = I_n(Q-1) + 1.$

By Eq. (18), we must have

$$\begin{aligned} B_n(Q-1) - B_n(Q) &\geq B_n(Q) - B_n(Q+1) \text{ in Case 1,} \\ B_n(Q-1) - B_n(Q) &= 0 \leq B_n(Q) - B_n(Q+1) \text{ in Case 2,} \\ B_n(Q-1) - B_n(Q) &= B_n(Q) - B_n(Q+1) \text{ in Case 3,} \\ B_n(Q-1) - B_n(Q) &\geq B_n(Q) - B_n(Q+1) \text{ in Case 4.} \end{aligned}$$

It is easy to see that Case 1 in one cycle can only generate Case 1 or 2 or 3 in the next cycle, Case 2 in one cycle can only generate Case 1 or 2 or 3 in the next cycle, Case 3 in a cycle only generates Case 1 or 3 or 4 the next cycle, and finally, Case 4 in a cycle only generates Case 1 or 4 in the next cycle. Therefore, by the initial condition, each Case 2 must be generated by Case 1 in an earlier cycle $n' < n$, where

$$B_{n'}(Q-1) - B_{n'}(Q) = 1 > B_{n'}(Q) - B_{n'}(Q+1) = 0.$$

Furthermore, Proposition 3 implies that in Case 2, $B_n(Q) - B_n(Q+1) \leq 1$, where equality holds only when the next cycle is Case 3. Thus, each Case 2 with

$$B_n(Q-1) - B_n(Q) = 0 < B_n(Q) - B_n(Q+1) = 1$$

corresponds one-for one to an earlier cycle n' with

$$B_n(Q-1) - B_n(Q) = 1 > B_n(Q) - B_n(Q+1) = 0,$$

which implies that inequality (20) holds for any $J > 0$. \square

Based on Proposition 4, we propose the following algorithm for computing the optimal container-yard capacity Q^* and operation cost $g(Q^*)$ under a given cycle time T .

Min_Cost_Capacity (T)

Step 0: Set $Q = 1$.

Step 1: Compute $g(Q)$ and $g(Q+1)$ by (16).

Step 2. If $g(Q) \leq g(Q+1)$, then $Q^ = Q$ and $g(Q^*) = g(Q)$, stop.*

Otherwise, $Q \leftarrow Q+1$, go to Step 1.

In practice, b^c may be an increasing function in Q because the port needs to buffer greater variation in inventory. If $b^c(Q)$ is concave, then Proposition 4 and algorithm ***Min_Cost_Capacity (T)*** still hold if we redefine $g(Q) = b^s \cdot Q / H + \phi \cdot E(B) / T - b^c(Q) \lambda$. If $b^c(Q)$ is not concave, then a full scale search is required to identify the optimal Q^* for each cycle time T .

Our analysis in Sections 2 and 3 lead to the following iterative search procedure for the optimal cycle time T^* . Let $[T^{\min}, T^{\max}]$ be the given range of possible cycle times, where T^{\min} requires all the vessels in set V^0 to serve as backhaul vessels, and T^{\max} requires the use of only a single backhaul vessel to meet the cycle time T .

Algorithm: Cycle_time

Step 1. $T \leftarrow T^{\min}$, $\xi^0 = \text{any feasible lower bound on the total profit}$;

Step 2. (Iteration)

*Call algorithm **V-Scheduler(T)** and return with $G(T)$;*

*Call algorithm **Min_Cost_Capacity(T)** and return with $g(Q^*)$;*

Compute $\xi^{Total} = \beta \cdot (|V^0| - |V|) + G(T) + b^c \cdot \lambda \cdot H - g(Q^) \cdot H$;*

If $\xi^{Total} > \xi^0$, then $\xi^0 \leftarrow \xi^{Total}$, $T(\xi^0) \leftarrow T$, $T \leftarrow T + \Delta T$, and repeat Step 2;

Step 3. Let $\xi^ = \xi^0$, $T^* \leftarrow T(\xi^0)$, and stop.*

4. Empirical observations

In this section, we report our empirical observations on the computational performance of the proposed heuristic vessel scheduling algorithm $V_Scheduler(T)$ under a given cycle time T . The heuristic solutions are compared to the best solutions obtained by using the CPLEX solver (CPLEX version 9.1) to solve problem $P(T)$ directly within a preset CPU time limit (i.e., one-hour). The associated mixed integer programming problems encountered in the heuristic search process were all solved by the same CPLEX package on a Dell Desktop (PowerEdge 400SC, Pentium4 2.8GHz, 1G RAM). We also present a numerical example illustrating the proposed iterative search algorithm, *Cycle-Time*, that aims to maximize the joint profit of the shipping company and the container-yard at the domestic port.

The parameters and their range of values used in this empirical study are summarized in Table 1 below. Each combination of these parameter values defines a problem setting, for which ten (10) test instances were randomly generated by sampling the order sizes (d_k) from a respective uniform distribution.

Table 1: Parameters used for evaluating the performance of V-Scheduler(T)

<i>Parameter</i>	<i>Base value</i>	<i>Range of the parameter values</i>
<i>Vessel capacity c</i>	4,000 containers	
<i>Number of business orders K</i>	30	
<i>Length of planning horizon H</i>	100 days	
<i>Fleet size of the shipping company V^0</i>	8	
<i>Cycle time T</i>	15 days	
<i>Maximum order waiting time π</i>	12 days	5, 8, 10, 13, 16 days
<i>Maximum vessel waiting time W^{max}</i>	4 days	2, 5, 8, 10, 12 days
<i>Unit profit per cargo container α_k/d_k</i>	\$200	200, 400, 600, 800
<i>Order size (number of containers) d_k</i>	Uniform (80, 1500)	Uniform (80, d^{max}), where $d^{max} = 300, 800, 1200, 1500$.
<i>Order release time (at port P_0) r_i</i>	Uniform (0, H)	
<i>Order receiving time windows $[a_i, b_i]$</i>	Uniform (0, H)	
<i>Traveling time between two ports τ</i>	12 days	
<i>Vessel waiting cost q</i>	\$15,000/day	
<i>Cargo holding cost (at port P_i) h</i>	\$10/(day, container)	

The empirical observations on the computational performance of the proposed heuristic $V_Scheduler(T)$ are summarized in Tables 2, 2a, 2b, 3, 4a, 4b, 5a, 5b, and 5c. In these tables, ‘‘CPLEX Optimizer’’ refers to the approach that solves $P(T)$ directly by the CPLEX solver, ‘‘MIP-based Heuristic’’ stands for the proposed vessel scheduling heuristic algorithm $V_Scheduler(T)$, and ‘‘Performance (error) Gap’’ stands for the relative error between the objective values of the heuristic solution and the best CPLEX

solution obtained within one-hour of CPU time (which are thus optimal if and only if the CPLEX solver terminates the search before the CPU time limit is exceeded). As we can see, due to the combinatorial nature of the vessel scheduling problem, the CPLEX solver failed to find the optimal solution within the preset CPU time limit for most test instances. Whenever this is the case, the best solution found by CPLEX within the CPU time limit was used as a surrogate of the optimal solution in the comparison.

Table 2 shows the computational performance of the proposed vessel scheduling heuristic against the *unit profit* of customer orders α_k / d_k with all the other parameters fixed at their base values. In this table, “Average” and “Maximum” represent the average and the maximum error measures over the 10 randomly generated test instances with respect to each α_k / d_k value. As we can see, the average error gaps between the heuristic solutions and the best solutions obtained by the approach to solve $P(T)$ directly within the CPU time limit were well within 2% for all the test instances, while the heuristic required much less time to find the near optimal solution. For some instances, we received a negative error gap. This means that the proposed vessel scheduling algorithm $V_Scheduler(T)$ found a better solution than the best solution obtained by the CPLEX solver within the one-hour time limit. Meanwhile, the proposed heuristic required a significant less amount of computational effort. Note that while the CPLEX solver was also used for solving a generalized knapsack problem during the search process by the proposed heuristic, the solution effort was much less than that required for solving $P(T)$ directly.

Table 2: Performance of the heuristic against relative unit profit of cargo orders

α_k / d_k	CPU Time (Seconds)				Performance GAP		
	CPLEX Optimizer		MIP-based Heuristic		Average	Maximum	Std. Dev.
	Average	Maximum	Average	Maximum			
200	3600*	3600*	6.11	21.70	-0.02%*	0.41%*	0.46%
400	3600*	3600*	3.89	17.38	-0.21%*	0.68%*	0.70%
600	3600*	3600*	6.92	35.61	0.02%*	0.18%*	0.06%
800	3600*	3600*	5.53	24.67	0.16%*	1.94%*	1.05%

*The respective gap was based on the best solution by the CPLEX solver within one-hour CPU time limit.

We have also compared the heuristic solution with the optimal solution obtained by letting CPLEX solver run without time limit until the optimal solution is found. Table 2a reports these observations, where we set the parameter value, α_k / d_k , at its lower, and the upper, bound of those in Table 2 (i.e., $\alpha_k / d_k = 200$, and $\alpha_k / d_k = 800$, respectively), while keeping all the other parameters unchanged. For each given set of parameters, we randomly generated 5 test cases and report the optimal solution and the solution time need by CPLEX solver and by our proposed heuristic. As we can see, the proposed

heuristic demonstrated again a very competitive performance when compared to the optimal solutions in this case.

In Table 2b, we compared our proposed heuristic solution with the CPLEX solution that meets the 5% MIPGAP when the search terminates. As we can see, the proposed heuristic demonstrated a strong performance in this case as well while required much less amount of time comparing to what needed by the CPLEX solver to reach the 5% MIPGAP.

Table 2a: Heuristic vs. Optimal solutions

α_k / d_k	CPLEX Optimal		MIP-based Heuristic		Gap in solutions
	Solution	CPU time (s)	Solution	CPU time (s)	
200	26,037,840	1249	25,902,940	0.67	0.52%
200	26,649,350	2716	26,649,350	74.47	0.00%
200	26,253,020	830	26,253,020	0.70	0.00%
200	26,376,590	1316	26,267,940	1.56	0.41%
200	27,175,470	9923	27,085,380	0.45	0.33%
800	37,227,200	7729	37,215,850	0.50	0.03%
800	34,323,070	271	34,323,070	1.50	0.00%
800	34,897,670	709	34,386,280	0.55	1.47%
800	35,630,730	6607	34,805,340	17.91	2.32%
800	35,377,340	1172	35,285,880	0.52	0.26%

Table 2b: Heuristic vs. Theoretical Approximation (MIPGAP=5%) Solutions

α_k / d_k	CPLEX Solutions that satisfy the 5% MIPGAP		MIP-based Heuristic		Gap in solutions
	Solution	CPU time (s)	Solution	CPU time (s)	
200	26,942,160	1948	26,916,760	1.92	0.09%
200	26,121,790	410	26,121,790	0.66	0.00%
200	27,285,200	10538	27,285,200	0.53	0.00%
200	26,919,980	2721	26,810,020	2.39	0.41%
200	27,182,300	459	27,182,300	0.41	0.00%
800	33,347,000	57	33,233,290	23.30	0.34%
800	35,377,200	857	35,377,200	2.80	0.00%
800	36,764,760	1318	36,764,760	0.61	0.00%
800	34,720,770	363	34,720,770	122.61	0.00%
800	36,388,160	8268	36,388,160	0.44	0.00%

Table 3 reports on the computational performance of the proposed vessel scheduling heuristic against the order size d_k , where $d_k \in \text{Uniform}(80, d^{max})$ and $d^{max} = 300, 800, 1200, \text{ and } 1500$, respectively. All the other problem parameters were fixed at their based values specified in Table 1. As we can see, the heuristic vessel scheduling

algorithm, $V_Scheduler(T)$, demonstrated a strong performance in this second set of test instances as well, while consistently required a much less amount of search effort.

Table 3: Performance of the heuristic against cargo loads

d_k	CPU Time (Seconds)				Performance GAP		
	CPLEX Optimizer		MIP-based Heuristic		Average	Maximum	Std. Dev.
	Average	Maximum	Average	Maximum			
$U(80,300)$	3600*	3600*	43.26	104.81	0.02%	0.09%	0.03%
$U(80,800)$	3600*	3600*	50.73	228.47	0.10%	0.36%	0.13%
$U(80,1200)$	3600*	3600*	64.14	378.72	0.00%	0.27%	0.12%
$U(80,1500)$	3600*	3600*	47.40	284.30	-0.09%	0.05%	0.28%

*The respective gap was based on the best solution by the CPEX solver within one-hour CPU time limit.

Tables 4a and 4b show the performance of the proposed vessel scheduling algorithm against the relative maximum order waiting time at port P_0 , π/T , where $T=15$ and π was set to be 5, 8, 10, 13 and 16, respectively. All the other parameters remained at their base values listed in Table 1. The proposed heuristic $V_Scheduler(T)$ demonstrated again a strong computational performance, as compared to the approach of solving $P(T)$ directly.

Table 4a: Performance of the heuristic against π/T ratio ($|K|=30$)

π/T	CPU Time (Seconds)				Performance GAP		
	CPLEX Optimizer		MIP-based Heuristic		Average	Maximum	Std. Dev.
	Average	Maximum	Average	Maximum			
5/15	3600*	3600*	67.70	299.88	-0.08%	0.00%	0.19%
8/15	3600*	3600*	7.31	40.47	-0.02%	0.00%	0.06%
10/15	3600*	3600*	26.66	241.11	-0.01%	0.00%	0.03%
13/15	3600*	3600*	2.85	9.61	0.02%	0.62%	0.36%
16/15	3600*	3600*	5.65	18.89	0.31%	1.11%	0.39%

*The respective gap was based on the best solution by the CPEX solver within one-hour CPU time limit.

Table 4b: Performance of the heuristic against π/T ratio ($|K|=20$)

π/T	CPU Time (Seconds)				Performance GAP		
	CPLEX Optimizer		MIP-based Heuristic		Average	Maximum	Std. Dev.
	Average	Maximum	Average	Maximum			
5/15	1563.60	3600*	3.01	6.50	0.00%	0.00%	0.00%
8/15	1944.15	3600*	1.38	7.00	0.00%	0.00%	0.00%
10/15	1284.62	3600*	2.40	12.50	0.00%	0.00%	0.00%
13/15	1440.38	3600*	9.88	70.48	0.09%	0.28%	0.12%
16/15	1833.49	3600*	14.08	44.53	0.22%	0.96%	0.33%

*The respective gap was based on the best solution by the CPEX solver within one-hour CPU time limit.

It is interesting to note that the proposed heuristic $V_Scheduler(T)$ terminated at the optimal solution in this experiment under parameter settings $\pi/T=5/15$, $8/15$ and $10/15$. The reason is that when condition $\pi < T - W^{\max}$ holds, where we had $T=15$ and $W^{\max}=4$ for this experiment, problem $P(T)$ becomes decomposable according to Proposition 1 and therefore the proposed heuristic guarantees the optimality. This is also evidenced by the maximum errors that are equal to zero for the respective entries in Tables 4a and 4b. As we can also see, the heuristic was able to find the optimal vessel schedule, in the worst case, in no more than five minutes of CPU time, versus an excessive CPU time (more than one hour) needed by the approach that solves $P(T)$ directly.

Table 5a: Performance of the heuristic against W^{\max} / T ratio ($|K|=30$)

W^{\max} / T	CPU Time (Seconds)				Performance GAP		
	CPLEX Optimizer		MIP-based Heuristic		Average	Maximum	Std. Dev.
	Average	Maximum	Average	Maximum			
2/15	3600*	3600*	22.48	195.27	-0.11%	0.00%	0.15%
5/15	3600*	3600*	5.83	38.66	0.00%	0.36%	0.16%
8/15	3600*	3600*	3.89	10.23	-0.18%	0.16%	0.31%
10/15	3600*	3600*	22.01	195.78	-0.02%	0.65%	0.32%
12/15	3600*	3600*	3.97	6.95	-0.19%	0.30%	0.54%

*The respective gap was based on the best solution by the CPEX solver within one-hour CPU time limit.

Table 5b: Performance of the heuristic against W^{\max} / T ratio ($|K|=20$)

W^{\max} / T	CPU Time (Seconds)				Performance GAP		
	CPLEX Optimizer		MIP-based Heuristic		Average	Maximum	Std. Dev.
	Average	Maximum	Average	Maximum			
2/15	2025.35	3600*	6.27	32.50	0.00%	0.00%	0.00%
5/15	2283.95	3600*	10.11	95.34	0.07%	0.19%	0.08%
8/15	1090.65	3600*	0.78	1.38	0.08%	0.38%	0.13%
10/15	913.07	3600*	5.40	24.58	0.21%	0.79%	0.32%
12/15	1139.24	3600*	1.23	3.66	0.11%	0.31%	0.13%

*The respective gap was based on the best solution by the CPEX solver within one-hour CPU time limit.

Tables 5a and 5b summarize the empirical performance of the proposed vessel scheduling heuristic against *relative maximum vessel waiting time* W^{\max} / T , where $T=15$ while W^{\max} varied from 2 days to 12 days. All the other problem parameters remained the same at their base values (see Table 1). Again, when $W^{\max} = 2$ days, the heuristic found the optimal solution (Proposition 1) since the condition $\pi < T - W^{\max}$ was satisfied. Table 5c reports on our observations when comparing the heuristic solution with the optimal solutions by letting CPLEX solver run without time limit until the optimal solution is found. As we can see, the proposed heuristic achieved near optimal solutions

with a much less computational effort.

Table 5c: Heuristic vs. Optimal Solutions

W^{\max} / T	CPLEX Optimal		MIP-based Heuristic		Gap in solutions
	Solution	CPU time (s)	Solution	CPU time (s)	
10/15	26,373,830	663	26,373,830	0.53	0.00%
10/15	26,513,570	512	26,513,570	0.70	0.00%
10/15	26,445,680	238	26,445,680	2.36	0.00%
10/15	26,554,720	2916	26,469,800	0.52	0.32%
10/15	26,123,230	401	25,986,960	0.77	0.52%
12/15	26,585,730	893	26,585,730	0.69	0.00%
12/15	26,616,250	1001	26,615,410	0.73	0.00%
12/15	26,672,010	10531	26,638,210	0.69	0.13%
12/15	26,603,540	639	26,588,910	0.69	0.05%
12/15	26,546,240	158	26,515,520	2.31	0.12%

Table 5c summarizes the observations from an experiment where, for each given set of parameters, we randomly generated 5 test cases and report the optimal solution (by letting CPLEX run until the optimal solution is found) and the solution time need by the CPLEX solver. In the same table, we also report on the performance of the proposed vessel scheduling heuristic. As we can see, the heuristic algorithm demonstrated again a very competitive performance when compared to the optimal solutions in this case.

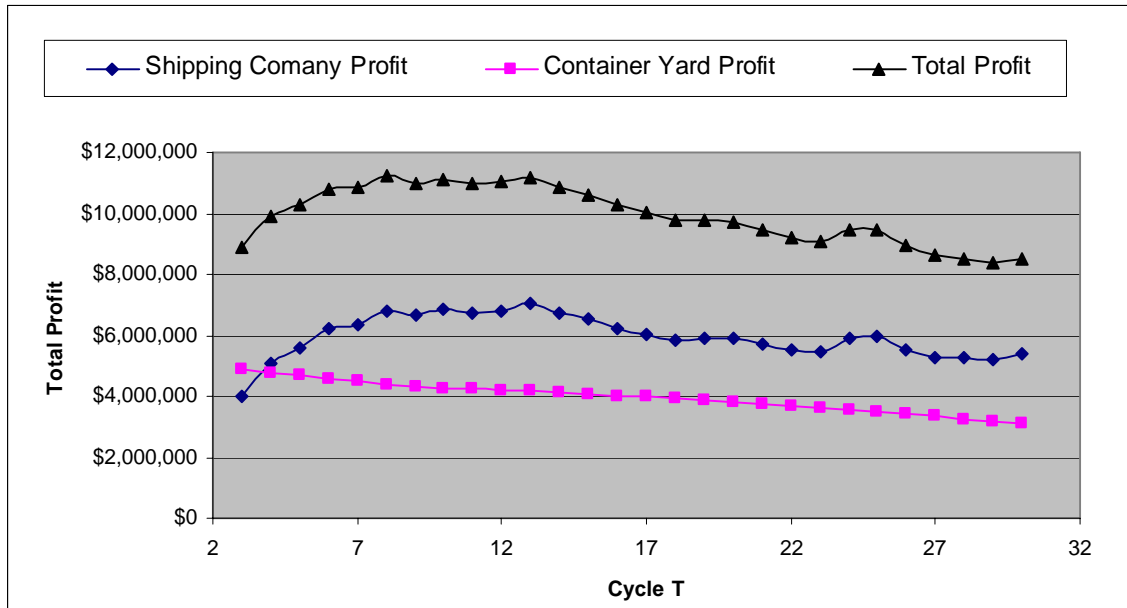


Figure 3: Shipping company profit vs. container yard profit.

Figure 3 shows what we observed from a numerical example where the proposed iterative search procedure, *Cycle_Time*, was applied to find the cycle time T^* to maximize the joint operational profit of the shipping company and the domestic

container-yard. For the given range of possible cycle times, $[T^{\min}, T^{\max}]$, this iterative search procedure calls the algorithms $V_Scheduler(T)$ and $Min_Cost_Capacity(T)$ to evaluate each candidate cycle time T and the resulting joint profit of the supply chain partners. For each cycle time T , the heuristic $V_Scheduler(T)$ constructs the maximum-profit vessel schedules to meet the given cycle time, while the $Min_Cost_Capacity$ algorithm determines optimal container-yard capacity to minimize the sum of capacity and overflow costs. In this experiment, we set $[T^{\min}, T^{\max}] = [3, 30]$ so that T^{\min} requires all the vessels in the fleet to be backhaul vessels (i.e., $V^0 = V$ with no business vessels in this case) and that T^{\max} allows a maximum number of business vessels and requires the use of only a single backhaul vessel during the planning horizon. We assumed that the expected profit per business vessel to be $\beta = \$4,000,000$ over the planning horizon, while all the other problem parameters remained at their base values (see Table 1). The parameters defining the container-yard operations at the domestic port P_I were as follows: the empty container arrival rate was set at $\lambda = 0.03c/day$, the container-yard leasing cost was assumed to be $b^s / H = \$8,000$ per vessel load per day, the overflow penalty cost per vessel load was $\phi = \$2,400,000$, and finally, the revenue per vessel load was $b^c = \$2,000,000$.

From Figure 3, we can observe that as the cycle time T increased, the container yard's profit decreased consistently, due to an increasing quantity of blocked containers and the need to have a larger container-yard capacity to avoid excessive overflows. On the other hand, the shipping company's profit increased as the cycle time T increased. However, the shipping company's operational profit is not a monotonic increasing function of cycle time T due to the discrete nature of the vessel scheduling problem. More detailed results obtained from this numerical example is listed in Table 6 where the optimal cycle T^* is 8 days, which results in a joint profit for the two supply chain partners at $\xi^* = \$11,217,380$.

5. Conclusion and future studies

We studied the problem of coordinating the operations between a foreign shipping company and a domestic container-yard for empty containers. The two partners had conflicting business objectives and the issue was to determine the *optimal cycle time* by which the shipping company removes the empty containers accumulated at the domestic port so that the joint profit of the two partners is maximized. The shipping company prefers a larger cycle time as it would allow more vessels to be available for transporting profitable cargo containers, while the container-yard prefers a shorter cycle time as it reduces the cost of overflows as well as the need for container-yard capacity. An iterative

search procedure searching for this optimal cycle time was proposed. Each iteration evaluates a candidate cycle time by solving a deterministic vessel scheduling problem and a stochastic container-yard capacity optimization problem. A mixed-integer programming based heuristic vessel scheduling algorithm was proposed, and the condition under which the heuristic finds the optimal solution was identified. An incremental search procedure based on the convexity of the profit function of the domestic port was derived to determine the optimal container-yard capacity. Empirical observations on the algorithm performance were reported, and a numerical example that illustrates the use of the proposed iterative procedure was included.

This study can be extended in several ways. First, the port-to-port shipping process hypothesized in this study is a relatively simple one as compared to those encountered in practice. Many important factors, such as loading/unloading operations, port availability, possibility for a vessel to leave at an earlier time, seasonality in the demand, etc., have been omitted here in order to make the optimization problem manageable in the mathematical analysis. A detailed simulation study that focuses on the same vessel-port collaboration issue but takes these important factors into account will be an interesting extension of the work. Second, we have assumed that the empty containers can be loaded to a backhaul vessel instantaneously at the pre-specified time points. This is, however, no longer the case if the container loading facilities (e.g., cranes, berths) have to be scheduled to accommodate the need. Coordinating the vessel operations and the loading/unloading facility operations would require a more sophisticated scheduling algorithm. In addition, we have considered only port-to-port with homogeneous vessels in this study. However, in reality vessels owned by a shipping company could vary significantly in their speeds and loading capacities, and a vessel may visit many ports along its route. When the heterogeneity of vessels and the vessel routing have to be considered, the vessel dispatching issue arises and the resulting scheduling problem could become much more complex but also more interesting.

Acknowledgement

The authors are grateful to the helpful comments and suggestions made by the referee, which have lead to a nice improvement of the paper.

References:

Avi-Itzhak, B. and S. Ben-Tuvia, 1963. The problem of optimizing a collective reservoir system. *Operations Research*, 11 pp.122-136.

- Bendall, H.B. and Stent, A.F., 2001. A scheduling model for a high speed containership service: a hub and spoke short-sea application. *International Journal of Maritime Economics*, 3 pp.262-77.
- Boile, M., Theofanis, S. and Mittal, N., 2004. Empty Intermodal Containers - A Global Issue, *Proceedings of the 2004 Transportation Research Forum Annual Forum*.
- Brooks, M., 2004. Guest Editorial: Is liner shipping unique? *Mar. Pol. Mgmt.* 31 (4), pp.257-68
- Carter, J. R. and Ferrin, B. G., 1995. The impact of transportation costs on supply chain management. *J Bus Logist*, 16 (1), pp.189-212.
- Chen, Z. L., Lei, L., and Zhong, H., 2006. Container vessel scheduling with bi-directional flows. *Operation Research Letters* (Accepted).
- Cho, S-C. and Perakis, A.N., 1996. Optimal liner fleet routing strategies. *Maritime Policy and Management*, 23 (3) pp.249-59.
- Christiansen, M., Fagerholt, K. and Ronen, D., 2004. Ship routing and scheduling: status and perspective. *Transportation Science*, 38 (1), pp.1-18
- Claessens, E.M., 1987. Operation procedure in maritime fleet management. *Maritime Policy and Management*, 14 (1) pp.27-48.
- Fagerholt, K., 1999. Optimal fleet design in a ship routing problem. *International Transactions in Operational Research*, 6 (5) pp.453-64.
- Fagerholt K., 2004. Designing optimal routes in a liner shipping problem. *Maritime Policy and Management*, 31 (4), pp.259-68
- Fagerholt, K., 2004. A computer-based decision support system for vessel fleet scheduling-experience and future research. *Decision Support Systems*, 37 pp.35-47.
- Harel, A., 1990. Convexity properties of the Erlang loss formula. *Operations Research*, 38 pp. 499-505.
- Ibarra, O. H. and Kim, C. E., 1975. Fast Approximation Algorithms for the Knapsack and Sum of Subset Problems. *Journal of ACM*. 22, pp. 463-468.
- Industry Survey Series: Mainstream Container Services. 2003. US Department of Transportation, Maritime Administration, June 2004.
- Jagers, A.A. and E.A. Doorn, 1986. On the continued Erlang loss function. *Operations Research letters*, 5 pp.43-46.
- Jaramillo, D.I. and Perakis, A.N., 1991. Fleet deployment optimization for liner shipping. Part 2: Implementation and results. *Maritime Policy and Management*, 18 (4) pp.235-62.
- Kohli, R. and H. Park, 1994. Coordinating buyer-seller transactions across multiple products. *Mgmt Sci.*, 40 pp. 45-50.
- Kumaran, K., M. Mandjes and A. Stolyar, 2003. Convexity properties of loss and overflow functions. *Operations Research Letters*, 31 pp. 95-100.
- Lane, T., T.D. Heaver and D. Uyeno, 1987. Planning and scheduling for efficiency in liner shipping. *Maritime Policy and Management*, 14(2) pp.109-25.
- Lee, H. L. and M. J. Rosenblatt, 1986. A generalized quantity discount model to increase supplier profit. *Mgmt Sci.*, 32 pp. 1177-1185.

Lei, L., Wang, Q. And Fan, C., Optimal Business Policies for a Supplier-Transporter-Buyer Channel with a Price-Sensitive Demand. *Journal of Operational Research Society* (in print, 2006).

Li, L. and H. Zhang, 2000. The multistage service facility start-up and capacity model. *Operations Research*, 48 pp. 490-497.

Logistics Today, 2006. Outsourcing is only as good as the relationship between the buyer and the provider [online]. <http://www.logisticstoday.com/displayStory.asp> [cited Feb., 2006].

Moran, P.A.P., 1954. A probability theory of Dams and storage systems. *Aust. J. Appl. Sci.* 5 pp.116-124.

Perakis, A.N. and Jaramillo, D.I., 1991. Fleet deployment optimization for liner shipping. Part 1: Background, problem formulation and solution approaches. *Maritime Policy and Management*, 18 (3) pp.183-200.

Prabhu, N.U., 1965. *Stochastic processes*. Macmillan, New York.

Rana, K. and Vickson, R.G., 1988. A model and solution algorithm for optimal routing of a time-chartered containership. *Transportation Science*, 22 (2) pp.83-95.

Rana, K. and Vickson, R.G., 1991. Routing container ships using Lagrangean relaxation and decomposition. *Transportation Science*, 25 (3) pp.201-14.

Ronen, D., 1983. Cargo ships routing and scheduling: survey of models and problems. *European Journal of Operational Research*, 12, pp.119-26.

Ronen, D., 1993. Shipping scheduling: the last decade. *European Journal of Operational Research*, 71, pp.325-33.

Sambracos, E. et al., 2004. Dispatching of small containers via coastal freight liners: The case of the Aegean Sea. *European Journal of Operational Research*, 152 pp.365-81.

Shanthikumar, J.D. and D.D. Yao, 1991. Strong stochastic convexity: closure properties and applications. *Journal of Applied Probability*, 28 pp. 131-145.

Shintani, K. et al., 2005. The container shipping network design problem with empty container repositioning. *Transportation Research Part E*, doi:10.1016/j.tre.2005.05.003.

VTC/HDR Team, 2004. Land use planning support for projected growth at port Newark and Elizabeth. *Report of Alan M. Voorhees Transportation Center*, Rutgers University.

Table 6: Shipping company and container yard profit against pick-up cycle T

Cycle T	Shipping Company						Container Yard					Total Profit
	m	A1	A2	A3	A4	Total	Q*	B1	B2	B3	Total	
3	8	\$0	\$4,719,400	\$0	\$745,600	\$3,973,800	1	\$6,000,000	\$800,000	\$313,881	\$4,886,120	\$8,859,920
4	6	\$1,000,000	\$4,719,400	\$0	\$618,220	\$5,101,180	1	\$6,000,000	\$800,000	\$413,802	\$4,786,200	\$9,887,380
5	5	\$1,500,000	\$4,719,400	\$15,000	\$599,800	\$5,604,600	1	\$6,000,000	\$800,000	\$513,875	\$4,686,130	\$10,290,730
6	4	\$2,000,000	\$4,719,400	\$0	\$527,880	\$6,191,520	1	\$6,000,000	\$800,000	\$610,590	\$4,589,410	\$10,780,930
7	4	\$2,000,000	\$4,719,400	\$15,000	\$367,070	\$6,337,330	1	\$6,000,000	\$800,000	\$705,350	\$4,494,650	\$10,831,980
8	3	\$2,500,000	\$4,719,400	\$0	\$403,840	\$6,815,560	1	\$6,000,000	\$800,000	\$798,177	\$4,401,820	\$11,217,380
9	3	\$2,500,000	\$4,695,600	\$90,000	\$440,210	\$6,665,390	1	\$6,000,000	\$800,000	\$889,089	\$4,310,910	\$10,976,300
10	3	\$2,500,000	\$4,719,400	\$60,000	\$310,400	\$6,849,000	2	\$6,000,000	\$1,600,000	\$135,188	\$4,264,810	\$11,113,810
11	3	\$2,500,000	\$4,415,200	\$15,000	\$154,950	\$6,745,250	2	\$6,000,000	\$1,600,000	\$165,938	\$4,234,060	\$10,979,310
12	2	\$3,000,000	\$4,064,800	\$0	\$241,380	\$6,823,420	2	\$6,000,000	\$1,600,000	\$200,044	\$4,199,960	\$11,023,380
13	2	\$3,000,000	\$4,403,200	\$30,000	\$354,870	\$7,018,330	2	\$6,000,000	\$1,600,000	\$237,488	\$4,162,510	\$11,180,840
14	2	\$3,000,000	\$3,930,600	\$60,000	\$122,200	\$6,748,400	2	\$6,000,000	\$1,600,000	\$278,227	\$4,121,770	\$10,870,170
15	2	\$3,000,000	\$3,709,600	\$30,000	\$132,740	\$6,546,860	2	\$6,000,000	\$1,600,000	\$322,197	\$4,077,800	\$10,624,660
16	2	\$3,000,000	\$3,518,600	\$180,000	\$91,660	\$6,246,940	2	\$6,000,000	\$1,600,000	\$369,961	\$4,030,040	\$10,276,980
17	2	\$3,000,000	\$3,497,800	\$30,000	\$409,510	\$6,058,290	2	\$6,000,000	\$1,600,000	\$420,319	\$3,979,680	\$10,037,970
18	2	\$3,000,000	\$3,197,800	\$30,000	\$325,850	\$5,841,950	2	\$6,000,000	\$1,600,000	\$473,636	\$3,926,360	\$9,768,310
19	2	\$3,000,000	\$3,345,400	\$90,000	\$333,970	\$5,921,430	2	\$6,000,000	\$1,600,000	\$529,773	\$3,870,230	\$9,791,660
20	2	\$3,000,000	\$3,181,600	\$75,000	\$196,760	\$5,909,840	2	\$6,000,000	\$1,600,000	\$588,581	\$3,811,420	\$9,721,260
21	2	\$3,000,000	\$2,855,000	\$30,000	\$128,200	\$5,696,800	2	\$6,000,000	\$1,600,000	\$649,893	\$3,750,110	\$9,446,910
22	2	\$3,000,000	\$2,722,600	\$60,000	\$117,690	\$5,544,910	2	\$6,000,000	\$1,600,000	\$713,534	\$3,686,470	\$9,231,380
23	2	\$3,000,000	\$2,640,600	\$30,000	\$127,840	\$5,482,760	2	\$6,000,000	\$1,600,000	\$779,322	\$3,620,680	\$9,103,440
24	1	\$3,500,000	\$2,631,600	\$0	\$215,250	\$5,916,350	2	\$6,000,000	\$1,600,000	\$847,573	\$3,552,430	\$9,468,780
25	1	\$3,500,000	\$2,660,800	\$30,000	\$150,110	\$5,980,690	2	\$6,000,000	\$1,600,000	\$917,200	\$3,482,800	\$9,463,490
26	1	\$3,500,000	\$2,200,200	\$30,000	\$147,520	\$5,522,680	2	\$6,000,000	\$1,600,000	\$988,413	\$3,411,590	\$8,934,270
27	1	\$3,500,000	\$1,922,400	\$60,000	\$79,860	\$5,282,540	2	\$6,000,000	\$1,600,000	\$1,061,020	\$3,338,980	\$8,621,520
28	1	\$3,500,000	\$1,873,600	\$60,000	\$60,350	\$5,253,250	2	\$6,000,000	\$1,600,000	\$1,134,820	\$3,265,180	\$8,518,430
29	1	\$3,500,000	\$1,803,600	\$105,000	\$17,810	\$5,180,790	2	\$6,000,000	\$1,600,000	\$1,209,620	\$3,190,380	\$8,371,170
30	1	\$3,500,000	\$2,026,000	\$15,000	\$112,340	\$5,398,660	2	\$6,000,000	\$1,600,000	\$1,285,250	\$3,114,750	\$8,513,410

m= no. of backhaul vessel; A1= profit of business vessel; A2= profit from spot cargoes; A3= cargo waiting cost; A4=vessel waiting cost;
Q*= optimal container yard capacity (in terms of c); B1= container yard management fee; B2 = yard leasing cost; B3 = yard overflow cost;
Shipping company profit= A1+A2-A3-A4; Container yard profit = B1-B2-B3; Total profit = shipping company profit + container yard profit.