Old Dominion University

ODU Digital Commons

Exact Algorithms for Practical Instances of the Railcar Loading Problem at Marine Container Terminals

ManWo Ng
Old Dominion University, mng@odu.edu

Dung-Ying Lin

WILEY | Hindawi

Research Article

Exact Algorithms for Practical Instances of the Railcar Loading Problem at Marine Container Terminals

ManWo Ng [1] and Dung-Ying Lin [2]

^{1}Department of Information Technology and Decision Sciences, Old Dominion University, Norfolk, VA, USA
^{2}Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu, Taiwan

Correspondence should be addressed to Dung-Ying Lin; dylin@ie.nthu.edu.tw

With the growth in global trade and its environmental footprint, sustainable modes of freight movement are increasingly important in today's globalized world. This study focuses on on-dock rail, where the rail terminal is located within the marine container terminal. On-dock rail has in recent years become an essential mode of transportation to move containers out of congested marine container terminals. This study contributes to the literature by presenting tailored exact solution algorithms for a recently proposed optimization model to optimize the loading of double-stack trains. In particular, a 3-stage solution framework is presented for the case when rail cars have a single well. A complementary solution algorithm for the case when a rail car has multiple wells is also provided. Our results show that computational times can be drastically reduced, from 2+ hours to a matter of seconds, rendering the model suitable for practical use.

1. Introduction

According to the UN Conference on Trade and Development (UNCTAD), over 80 percent of global merchandise trade by volume and more than 70 percent by value are transported using the maritime mode of transportation. To manage this (growing) trade volume, there is a critical need to effectively manage the maritime transportation system, including the rapidly growing containerized trade. This has in recent years led to numerous advances in improved ocean container shipping management strategies (e.g., see [1–3]).

Rail, especially on-dock rail—where the rail terminal is located within the marine container terminal—is increasingly being recognized as a sustainable and necessary mode of transportation to move containers in and out of congested marine container terminals. This trend fits within the broader call in the maritime industry to reduce ocean shipping's carbon footprint [4]. In response to this call, maritime researchers have examined a range of alternatives, from the use of alternative fuels, to the setting of emission standards, speed optimization, and the use of shore power (e.g., see [5–7]).

To maximize rail's capacity, it is critical to build trains that are fully utilized when leaving the marine container terminal. Our focus in this study was on double-stack trains. For single-stack trains—that are subject to very different loading constraints—the reader is referred to the literature (e.g., see [8]). A limited number of studies have appeared in the literature that examined the problem of loading double-stack trains, with each study valid under its own set of assumptions appropriate for the geography and operating environment. To the best of our knowledge, Pacanovsky et al. [9] was the first research appearing in the literature discussing the problem of loading double-stack trains. In their work, an inland rail yard was considered where containers arrive continuously and a location on a double-stack train needs to be assigned immediately upon arrival. A heuristic was used to assign the containers. Lai et al. [10] presented an optimization model to optimize the aerodynamics of intermodal freight trains carrying both containers and trailers. It was noted that despite commercial decision support software being available, deciding where to place containers on (double-stack) trains was still "a largely manual process" in practice. Lang et al. [11] examined the

loading of double-stack trains *in China*. While one of their goals was to maximize rail car maximization, their key focus appears to be on keeping the vertical center of gravity low. It is interesting to note that their loading constraints are very different from the model considered in this study. This is most likely due to the different equipment designs in China. For instance, they reported the possibility of loading two $20'$ containers on top of two other $20'$ containers, which is not common practice in the United States. Upadhyay et al. [12] presented a model for double-stack train loading tailored to the practice in India. As Lang et al. [11], they reported the possibility of loading two $20'$ containers on top of two other $20'$ containers. In addition to this key difference, Upadhyay et al. [12] also considered other constraints and practices in India that are uncommon in the United States. Most recently, Ng and Talley [13] presented an integer optimization model to load double-stack trains at marine container terminals. Unlike other models discussed above, Ng and Talley's model is tailored to the practice at U.S. container ports. The above-related works are summarized in Table 1, in terms of two key variables: the country for which the model was developed and whether the model has been developed for marine container terminals.

The current research is inspired by Ng and Talley [13]. The main focus in Ng and Talley [13] was on presenting a model that captured real-world operating practice. (Before proceeding, it is recommended that the reader reviews the model by Ng and Talley [13], which is summarized in Appendix A. For a more detailed discussion, the reader is referred to the original work.) The resulting model was solved for small problem instances with an off-the-shelf commercial solver. When the number of containers and rail cars is more realistic, i.e., larger, it becomes computationally prohibitive when using off-the-shelf solvers. (As we shall demonstrate in Section 5, it can take more than 2 days to find an optimal loading plan for practical instances of the problem, when using off-the-shelf solvers.) Marine container terminals need answers quickly for the model to be useful in practice. In this study, we exploit the problem structure to develop tailored solution methods for the rail loading problem. As shall be demonstrated, computational times will be drastically reduced, making the model suitable for practical use.

The remainder of this study is organized as follows. In Section 2, a tailored, three-stage solution method is provided to solve the optimization model. Section 3 considers an extension of the model in which rail cars can have multiple wells. A new solution framework is provided for this case. Section 4 illustrates the proposed methods with detailed computational results. Section 5 concludes the study.

2. Three-Stage Solution Approach for Single-Well Rail Cars

To solve model (P) in Appendix A, a tailored 3-stage solution approach is proposed in this section. The reader is advised to study the model formulation and its notation in Appendix A first before proceeding. The proposed method decomposes the problem into three stages. The first two stages determine

TABLE 1: Summary of key studies of double-stack rail optimization.

Study	Country	Maritime
Pacanovsky et al. [9]	United States	No
Lai et al. [10]	United States	No
Lang et al. [11]	China	No
Upadhyay et al. [12]	India	No
Ng and Talley [13]	United States	Yes

an effective lower bound, while the final stage calculates a sharp upper bound for the objective function in model (P).

2.1. The First Stage. Note that the utilization of a rail car can be either 100%, 50%, or 0%. In the first stage, we try to maximize the number of 100% utilization cars. To this end, the following procedure is followed:

(1) The objective function is changed to Max $Z^1 = \sum_j \sum_k y_{jk1}$ in model (P). Furthermore, an additional constraint $1/2 \left(\sum_{i \in I_h} x_{ik} + \sum_{m \in M_h} v_{mk} \right) + \sum_{j \in J_h} y_{jk2} = \sum_{j \in J_h} y_{jk1}$, $\forall k \in K$, is added to ensure that there must be an upper-level container if the lower level is loaded with containers. This constraint forces the rail cars to be loaded to 100% utilization.

(2) If the number of 100% utilization cars (say Q) is equal to the number of available cars, i.e., $Q = |K|$, then the optimal solution is 100%.

(3) On the other hand, if $Q < |K|$, then there is at least one rail car that cannot be loaded to 100% utilization. In this case, the containers that have not been assigned a rail car will be transferred to the next stage.

In other words, the first stage solves the following program **P1**:

$$Z^1 = \text{Max} \sum_j \sum_k y_{jk_1}. \tag{1}$$

s.t.

Constraints (A.2)–(A.19).

$$\frac{1}{2} \left(\sum_{i \in I_h} x_{ik} + \sum_{m \in M_h} v_{mk} \right) + \sum_{j \in J_h} y_{jk_2} = \sum_{j \in J_h} y_{jk_1}, \quad \forall k \in K. \tag{2}$$

As this formulation only finds 100% utilization cars, the solution space is small. As will be shown in the numerical experiment in Section 4, this formulation (or more accurately, its linearized version) can be solved efficiently using commercial optimization packages (i.e., CPLEX).

2.2. The Second Stage. When $Q < |K|$, the solution method enters the second stage. In this case, there must be at least one rail car that cannot be loaded to 100% utilization. In this second stage, we then try to maximize the number of 50% utilization cars.

In this stage, the containers that have been assigned to the rail cars and the rail cars loaded with 100% utilization in the first stage are no longer considered. The interest in the

second stage is only in the remaining containers and rail cars that have not been assigned/loaded.

It is to be noted that in the second stage no rail cars will achieve 100% utilization. That is, we can discard constraints (A.8) and (A.12). Also, we eliminate constraint (2) from the first stage because the remaining containers will be loaded on the cars with a maximum utilization of 50% only and we cannot force the car's utilization to 100%.

In other words, the second stage solves the following program **P2**:

$$Z^2 = \text{Max} \ \frac{1}{4|K|} \left(\sum_k \sum_i x_{ik} + \sum_k \sum_m v_{mk} + 2 \sum_l \sum_k \sum_j y_{jkl} \right),$$

s.t.

$$(3)$$

Constraints (A.2)–(A.7), constraints (A.9)–(A.11), constraint (A.13), and constraints (A.14)–(A.19).

As the first stage, this second-stage problem can be efficiently solved using CPLEX. Proposition 1 now is as follows.

Proposition 1. $Z^1 + Z^2$ is a lower bound to model (P).

Proof. Since the first stage identifies 100% utilization cars and the second stage the 50% utilization rail cars, the first two stages together provide a feasible solution to the original problem. **Q.E.D.** □

2.3. The Third Stage. Because of the sequential nature of the first two stages, the loading plan thus far is generally suboptimal. To improve on the utilization, a third stage is introduced. In the third stage, we first derive an upper bound on the utilization. After that, we solve the original formulation considering all containers and rail cars with an additional constraint that bounds the value of objective function. Before presenting the details, we introduce the following notation.

   (i) $\underline{Z}$: the sum of the objective values from the first two stages of the solution method, i.e., $\underline{Z} = Z^1 + Z^2$.
   (ii) a_h: the numbers of $20'$ laden containers destined for hub h that are not loaded in the first two stages.
   (iii) b_h: the numbers of $20'$ empty containers destined for hub h that are not loaded in the first two stages.
   (iv) c_h: the numbers of $40'$ containers destined for hub h that are not loaded in the first two stages.
   (v) n: the number of remaining rail cars that are not loaded with any containers after the first two stages.
   (vi) a: $a = \sum_{h \in H} \lfloor a_h / 2 \rfloor$, which represents the maximum possible number of 50% utilization cars for the remaining $20'$ laden containers a_h.

   (vii) b: $b = \sum_{h \in H} \lfloor b_h / 2 \rfloor$, which represents the maximum possible number of 50% utilization cars for the remaining $20'$ empty containers b_h.
   (viii) c: $c = \sum_{h \in H} c_h$, which represents the maximum possible number of 50% utilization cars for the remaining $40'$ laden containers.

Recall that $|K|$ denotes the number of rail cars that can be loaded. The following result can then be derived.

Proposition 2. $\overline{Z} = \underline{Z} + (m \times 0.5/|K|)$ is an upper bound for model (P).

Proof. Using the above notation, note that at most $m = \min\{a + b + c, n\}$ additional rail cars can be loaded with containers after the first two stages of the solution method, each with 50% utilization. Therefore, it follows that $\overline{Z} = \underline{Z} + (m \times 0.5/|K|)$ is an upper bound. **Q.E.D.**

Based on the upper and lower bounds in propositions 1 and 2, we can introduce the following constraint to the model (P):

$$\underline{Z} \le \frac{1}{4|K|} \left(\sum_k \sum_i x_{ik} + \sum_k \sum_m v_{mk} + 2 \sum_l \sum_k \sum_j y_{jkl} \right) \le \overline{Z}.$$

$$(4)$$

Then, we solve the following formulation **P3** in this third stage with all the original containers and rail cars as input:

$$Z = \text{Max} \ \frac{1}{4|K|} \left(\sum_k \sum_i x_{ik} + \sum_k \sum_m v_{mk} + 2 \sum_l \sum_k \sum_j y_{jkl} \right),$$

s.t.

$$(5)$$

Constraints (A.2)–(A.19).

$$\underline{Z} \le \frac{1}{4|K|} \left(\sum_k \sum_i x_{ik} + \sum_k \sum_m v_{mk} + 2 \sum_l \sum_k \sum_j y_{jkl} \right) \le \overline{Z}.$$

$$(6)$$

As will be demonstrated in our numerical experiments, constraint (6) is critical in finding optimal solutions with limited computational effort. Figure 1 summarizes our 3-stage solution framework. □

3. Case of N-Well Rail Cars

Model (P) assumes that rail cars have a single well. While this is a possible scenario, what is more common in practice is to have rail cars that consist of multiple wells N, with the most common case being $N = 5$ (TTX, 2020) (https://www.ttx.com/about/equipment/). Figure 2 shows a 5-well rail car. In this case, the 3-stage framework no longer applies as it was tailored to the single-well case. In this section, a new solution technique is developed for the case of N-well rail cars, where $N > 1$.

To ensure that all well cars that are part of the rail car are dispatched toward the same hub, an additional constraint is
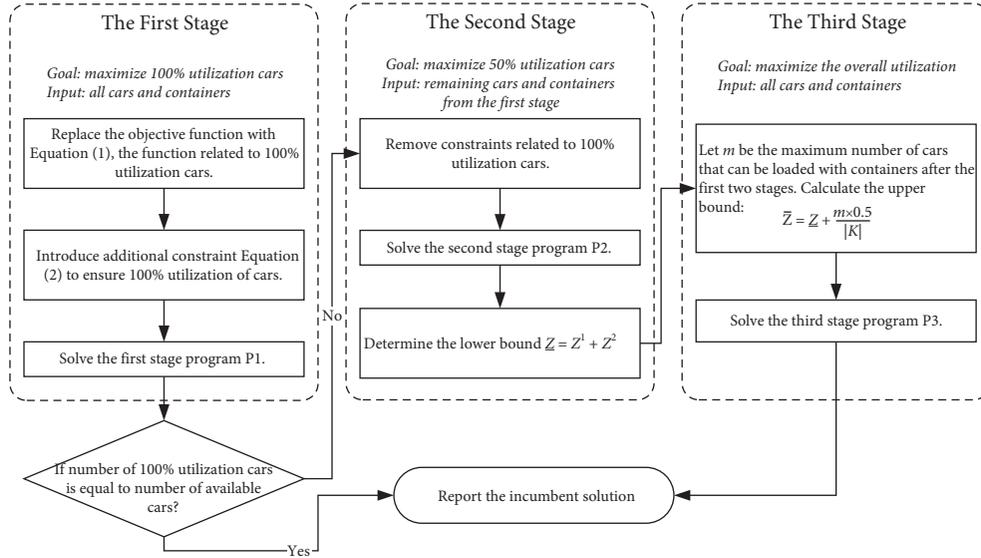
FIGURE 1: Three-stage solution framework.

FIGURE 2: A 5-well rail car.

introduced. In particular, for an N-well rail car ($N > 1$), this additional constraint is as follows:

$$\sum_{i=Nm+1}^{Nm+N} u_i^h = N u_{Nm+1}^h h \in H,$$
(7)
$$m = \left\{0, 1, \ldots, \left\lfloor \frac{K}{N} \right\rfloor - 1\right\}.$$

For example, suppose $N = 5$, then constraint becomes $\sum_{i=5m+1}^{5m+5} u_i^h = 5 u_{5m+1}^h$. Note that m indexes the number of N-well rail cars. For example, if there are three N-well rail cars, then the index m takes on the values 0, 1, and 2.

Unfortunately, including constraint (7) does not allow the resulting optimization problem to be solved efficiently. To alleviate this, we derive an upper bound (UB) on the utilization as follows. In Section 4, we will demonstrate that the UB derived is very sharp and can significantly reduce the computational efforts required to solve the formulation. Recall from Section 2 that a_h, b_h, and c_h represent the number of 20' laden, 20' empty, and 40' containers to hub h, respectively. Further, let F_h and H_h be the minimum numbers of fully loaded (i.e., 100% utilization) and half-loaded (i.e., 50% utilization) rail cars for hub h, respectively. Then, the following three steps can be used to calculate the UB on the utilization when constraint (7) is present.

Step 1. (estimate the minimum number of cars required).

In the first step, we ignore the weight limit and attempt to load all containers with the minimum number of cars. In particular, we create groups for each two randomly selected

20' laden containers and groups for each two randomly selected 20' empty containers. As the 20' containers can only be placed in the lower level, we load these groups to the cars first. Therefore, the minimum number of cars required to load the 20' containers can be estimated as $\lfloor a_i/2 \rfloor + \lfloor b_i/2 \rfloor$, as the weight limit is not considered in this step. Then, we try to load the 40' containers on top of the groups of 20' containers without considering the weight limit. Two scenarios are possible.

(i) $\lfloor a_i/2 \rfloor + \lfloor b_i/2 \rfloor \geq c_h$

Because the number of 20' containers required to load the cars with only 20' laden and empty containers ($\lfloor a_i/2 \rfloor + \lfloor b_i/2 \rfloor$) is greater than the number of 40' containers (c_h), in this case, the number of cars that can be fully loaded F_h is equal to the number of 40' containers c_h. The remainder of the cars can only be half-loaded, and the number of this type of cars H_h is $\lfloor a_h/2 \rfloor + \lfloor b_h/2 \rfloor - c_h$. Therefore, $F_h = c_h$ and $H_h = (a_h/2) + (b_h/2) - c_h$ in this scenario, respectively.

(ii) $\lfloor a_h/2 \rfloor + \lfloor b_h/2 \rfloor < c_h$

In this case, the number of 20' containers is not enough to create the cars with 100% utilization using the combination of 20' and 40' containers. There can be multiple cars with only one 40' container in the lower level of cars. We can estimate $F_h = (\lfloor a_h/2 \rfloor + \lfloor b_h/2 \rfloor) + \lfloor c_h - (\lfloor a_h/2 \rfloor + \lfloor b_h/2 \rfloor)/2 \rfloor$, where the first term $\lfloor a_h/2 \rfloor + \lfloor b_h/2 \rfloor$ is the number of fully loaded cars with a combination of 20' and 40' containers. In the second term, $c_h - (\lfloor a_h/2 \rfloor + \lfloor b_h/2 \rfloor)$ is the remaining 40' containers that cannot be combined with 20' containers to form 100% utilization cars. Therefore, $\lfloor c_h - (\lfloor a_h/2 \rfloor + \lfloor b_h/2 \rfloor)/2 \rfloor$ represents the number of 100% utilization cars with only 40' containers, since only when two 40' containers are loaded to a car can that car have a utilization of 100%. Further, the minimum number of half-loaded

cars is $H_h = (c_h - (\lfloor a_h/2 \rfloor + \lfloor b_h/2 \rfloor)) \mod 2$. If $H_h = 1$, there can be a 50% utilization car with one $40'$. $H_h = 0$, otherwise.

Step 2. (bounding the number of N-well cars).

In this step, we bound the number of N-well cars and subsequently determine how to distribute the single-well cars to these N-well cars.

(i) Let B be the number of N-well cars ($N > 1$) and S be the number of remaining unloaded single-well cars. Then, $B = \lfloor K/N \rfloor$, $S = K - N \times B$ (i.e., if $N = 5$, $B = \lfloor K/5 \rfloor$, $S = K - 5 \times B$), where K represents the number of total single-well cars. If a single-well car is with 100% utilization, we choose to load it first to the N-well car. Let M_h be the number of fully loaded N-well cars for hub h and $U_{h,i}$ be the utilization rate according to different loading approach i for partially loaded N-well cars for hub h. The following procedure estimates the number of N-well cars. If $F_h \geq N$ and $B > 0$, we want to know how many fully loaded cars can be sent for each hub h.

In this case, for each hub, we want to know how many fully loaded cars can be sent. Let $M_h = \min\{(F_h/N), (K/N)\}$, and we can update $F_h \leftarrow F_h - N \times M_h$ and $B \leftarrow B - M_h$.

(ii) If $F_h < N$

Since F_h is the remaining full car and H_h is the remaining half-loaded car to hub h, we can update U_{hi} as follows:

For $h = 1$ to $|H|$
 {
   For $i = 1$ to $F_h + (H_h/N)$
    {
     $U_{hi} =$
   $(1 \times F_h + 0.5 \times \min\{N - F_h, H_h\})/(N)$
     $F_h \leftarrow F_h - F_h$
     $H_h \leftarrow H_h - \min\{N - F_h, H_h\}$
    }
 }

In the equation $U_{h,i} = (1 \times F_h + 0.5 \times \min\{N - F_h, H_h\})/(N)$, the term $1 \times F_h$ is the utilization contributed by the 100% utilization cars in the N-well car. The term $0.5 \times \min\{N - F_h, H_h\}$ is the utilization contributed by the rest of the cars in the N-well car.

(iii) Let $N_n \forall n = 1, 2, \ldots, R$ be the utilization for N-well cars that are not fully loaded. We sort $U_{h,i}$ in descending order and assign cars into the big cars accordingly. Then, $F_h = 0, H_h = H_h - \min\{N - F_h, H_h\}$.

Step 3. (calculate the UB).

The last step calculates the UB. If the number of fully loaded cars is greater than or equal to the number of single cars that need to be assigned ($\sum_h F_h \geq S$),

$UB = \overline{Z^N} = (S \times 1 + N \times \sum_h M_h + N \times \sum_n N_n)/(K)$. The terms $S \times 1$, $N \times \sum_h M_h$, and $N \times \sum_n N_n$ are the utilization contribution of single cars, the utilization contribution of fully loaded N-well cars, and the utilization contribution of partially loaded N-well cars, respectively.

Otherwise, if the number of fully loaded cars is less than the number of single cars that need to be assigned ($\sum_h F_h < S$), $UB = \overline{Z^N} = 1 \times \sum_h F_h + 0.5 \times \min\{S - \sum_h F_h, \sum_h H_h\} + N \times (\sum_h M_h + N \times \sum_n N_n)/(K)$. In this case, $1 \times \sum_h F_h$, $0.5 \times \min\{S - \sum_h F_h, \sum_h H_h\}$, $N \times \sum_h M_h$, and $N \times \sum_n N_n$ are the utilization contributions of 100% utilization single car, 50% utilization single car, fully loaded N-well car, and partially loaded N-well car, respectively.

To summarize, when N-well rail cars are present (with $N > 1$), the following optimization problem is solved:

$$Z = \text{Max } \frac{1}{4|K|} \left(\sum_k \sum_i x_{ik} + \sum_k \sum_m v_{mk} + 2 \sum_l \sum_k \sum_j y_{jkl} \right). \tag{8}$$

s.t.
Constraints (A.2)–(A.19) and constraint (7) and

$$\frac{1}{4|K|} \left(\sum_k \sum_i x_{ik} + \sum_k \sum_m v_{mk} + 2 \sum_l \sum_k \sum_j y_{jkl} \right) \leq \overline{Z^N}. \tag{9}$$

4. Numerical Demonstration

4.1. Single-Well Rail Cars. To validate the proposed solution approach, we generated weights for the containers and weight limits of the rail cars according to real-world empirical distributions. (For a more detailed discussion of the weight data, see [13].) The cargo weight and car weight limit are between 10,000 and 100,000 and 140,000–180,000, respectively. The tolerance factor of a rail car is set to 1 in all cases (cf. Appendix A). The proposed 3-stage solution approach was implemented in the C# programming language. The numerical experiments were conducted on a Windows-based machine with an Intel 3.40 GHz CPU and 16 GB of memory.

Table 2 lists representative results for the 3-stage solution approach for 8 different instances of the problem. The CPLEX column shows the objective values found by CPLEX and the CPU time used when CPLEX is used to solve the problem instance. (All optimization problems are linearized first using the same techniques as in Ng and Talley [13] before CPLEX is applied.) For example, in Instance 1 the optimal utilization is 60%, and it took over 2562 seconds (42.7 minutes) to find this solution. This would make this "brute force" approach unacceptably slow for practical purpose.

When using the proposed 3-stage solution approach, the computation time can be drastically reduced. (Each of the stages is solved using CPLEX, applied to the respective linearized problems.) For example, for Instance 1, the column "Stage 1" shows the solution found after the first stage of the solution approach: a utilization of 50% (i.e., an

Table 2: Empirical results.

	CPLEX	Stage 1	Stage 2	Stage 3
Instance 1		$\#20' = 30$ $\#40' = 10$ $\#Car = 20$		
Objective value	0.600	0.500	0.600	0.600
CPU time (sec)	2562.758	0.039	0.182	0.182
CPU time saved	—	100.00%	99.99%	99.99%
Optimality gap	—	16.67%	0.00%	0.00%
Instance 2		$\#20' = 30$ $\#40' = 10$ $\#Car = 25$		
Objective value	—	0.400	0.480	0.480
CPU time (sec)	172,800*	0.086	0.240	0.240
CPU time saved	—	100.00%	100.00%	100.00%
Optimality gap	—	16.67%	0.00%	0.00%
Instance 3		$\#20' = 30$ $\#40' = 10$ $\#Car = 30$		
Objective value	0.617	0.500	0.600	0.617
CPU time (sec)	111,617.903	0.151	0.774	1.015
CPU time saved	—	100.00%	100.00%	100.00%
Optimality gap	—	18.92%	2.70%	0.00%
Instance 4		$\#20' = 60$ $\#40' = 20$ $\#Car = 30$		
Objective value	—	0.667	0.800	0.817
CPU time (sec)	172,800*	0.192	0.564	1.583
CPU time saved	—	100.00%	100.00%	100.00%
Optimality gap	—	18.37%	2.04%	0.00%
Instance 5		$\#20' = 60$ $\#40' = 20$ $\#Car = 35$		
Objective value	—	0.571	0.700	0.700
CPU time (sec)	172,800*	0.104	7.518	7.518
CPU time saved	—	100.00%	100.00%	100.00%
Optimality gap	—	18.37%	0.00%	0.00%
Instance 6		$\#20' = 10$ $\#40' = 30$ $\#Car = 15$		
Objective value	0.900	0.800	0.900	0.900
CPU time (sec)	128,876.324	5.230	5.421	5.421
CPU time saved	—	100.00%	100.00%	100.00%
Optimality gap	—	11.11%	0.00%	0.00%
Instance 7		$\#20' = 10$ $\#40' = 30$ $\#Car = 20$		
Objective value	—	0.600	0.800	0.800
CPU time (sec)	172,800*	18.953	19.359	19.359
CPU time saved	—	99.99%	99.99%	99.99%
Optimality gap	—	25.00%	0.00%	0.00%
Instance 8		$\#20' = 15$ $\#40' = 45$ $\#Car = 20$		
Objective value	—	0.920	0.960	0.960
CPU time (sec)	172,800*	60.021	1252.628	1252.628
CPU time saved	—	99.97%	99.28%	99.28%
Optimality gap	—	4.17%	0.00%	0.00%

*The CPU time is limited to about 2 days.

optimality gap of 16.67% compared with the optimal utilization of 60%). This required just 0.039 seconds, i.e., a reduction of essentially 100%. To further improve this solution, the column "Stage 2" shows the solution after the second stage. As can be seen, the utilization has improved to 60% (recall that this is the optimal solution). The second stage took 0.182 seconds to find, which is a 99.99% reduction compared with the CPLEX column. While in this specific instance, the optimal solution was found after 2 stages, sometimes stage 3 is required (see column "Stage 3"), as was the case in Instance 3 in Table 2. In Instance 3, the utilization improved from 0.5 to 0.6 to 0.617, which is the optimal solution.

It is interesting to note that for 5 of 8 problem instances (instances 2, 4, 5, 7, and 8), CPLEX failed to find the optimal solution after 2 hours (172,800 seconds), which is the maximum computational time that was allotted to each instance. (In these instances, no objective values have been reported in Table 2.) On the other hand, our 3-stage solution approach was able to solve each of the instances to optimality within seconds.

4.2. N-Well Rail Cars. Since $N = 5$ is the most common in practice, we will present empirical results for this case in this section. The same problem data as for the single-well experiment in the previous section have been used. The results are summarized in Table 3. Note that we have repeated the optimal utilization from the single-well model for comparison purpose.

There are a number of interesting findings. First, compared to the single-well case, the 5-well rail car model can

TABLE 3: Empirical results for $N = 5$.

| Instance no. | Single-well car model | | 5-well car model | | 5-well car model with UB | |
	Objective value	Time (sec)	Objective value	Time (sec)	Objective value	UB
1	0.600	10.901	0.600	1.320	0.600	0.600
2	0.480	318.329	0.480	1.264	0.480	0.480
3	0.617	3600.237	0.617	1.435	0.617	0.617
4	0.817	33.407	0.817	32.604	0.817	0.817
5	0.700	3600.183	0.700	6.425	0.700	0.700
6	0.900	14.217	0.900	12.685	0.900	1.000
7	0.800	521.059	0.750	126.243	0.750	0.800
8	0.960	13.369	0.940	1.624	0.940	0.940

result in lower utilization since it introduces the additional constraint (7). For example, in instances 7 and 8 the utilizations are 0.75 and 0.94, respectively, which corresponds to a decrease of 6.25% and 2.08% compared with the single-well car model. Second, the upper bounds derived in this research are very sharp. Note that besides instances 6 and 7, the upper bounds derived are the exact optimal solutions. Finally, introducing the upper bound into the model indeed significantly reduces the computational efforts when solving the 5-well rail car model. For example, it can reduce the computational time from more than 1 hour to just a little over 1 second (cf. instance 3).

5. Concluding Remarks

On-dock rail has become an essential and sustainable mode of transportation to move containers out of congested marine container terminals. To help marine container terminals maximize rail's capacity in practice, it is critical to have both models and solution methods that are practice-ready.

This study contributed to the latter. Using a recently introduced rail capacity maximization model, we demonstrated that—when using a commercial off-the-shelf solver—the computational times for realistic problem instances are prohibitively long for the model to be practically useful. To address this issue, new and tailored exact solution methods were developed in this research. Particularly, a 3-stage solution framework has been proposed for the case when rail cars have a single well. To complement this case, a solution method for the case when a rail car has multiple wells has also been provided. In the computational study, our results show that computational times can be dramatically reduced from 2+ hours to a matter of seconds, rendering the model suitable for practical use.

There are at least three possible future research directions. First, while the model captures the most salient features observed in practice, it is possible to further extend the model to incorporate less common container types, most notably tank and hazardous cargo containers. Second, stochasticity during the planning and operation (i.e., random disruption or stochastic demand arrival) can also be incorporated. However, this might increase the computation times, which brings us to the last future research direction. Third, while our solution approach is highly efficient, it might be interesting to explore other solution methods, especially when considering different model extensions. For example, evolutionary algorithms have generally shown to be effective in addressing large-scale real-world problems; e.g., see Zhao and Zhang [14, 15].

Appendix

A. Ng and Talley's Model Formulation

For ease of reference, we will summarize the optimization model from Ng and Talley [13] here: once discharged from a vessel, a rail container is stored at the terminal until it gets loaded on a double-stack train. Each container has a designated rail hub, i.e., its destination. Each train carries containers to prespecified rail hubs. It is the marine terminal's rail manager's job to assign rail hubs to specific rail cars. The model below assumes that all rail cars consist of a single-well car. This can easily be relaxed with some more cumbersome notation [13].

The goal of the model is to maximize rail's capacity by loading as many import containers (each with a certain destination, called the rail hub) on the available rail cars (each rail car can be assigned a certain rail hub).

A.1. Sets

$H = \{1, 2, \ldots, d\}$ Set of rail hubs.

I_h Set of *laden* 20′ containers destined for hub $h \in H$.

M_h Set of *empty* 20′ containers destined for hub $h \in H$.

J_h Set of 40′ containers (laden or empty) destined for hub $h \in H$.

$K = \{1, 2 \ldots, r\}$ Set of rail cars to be loaded.

$L = \{1, 2\}$ Position in rail car. $L = 1$ refers to top position. $L = 2$ refers to bottom.

A.2. Parameters

w_j^{40} Gross weight of the (laden or empty) 40′ container $j \in J_h$, $h \in H$.

w_i^{20} Gross weight of the laden 20′ container $i \in I_h$, $h \in H$.

w_m^{mt20} Gross weight of the empty 20′ container $m \in M_h$, $h \in H$.

c_k Weight capacity of rail car $k \in K$.

γ_k Weight tolerance factor for rail car $k \in K$.

A.3. Decision Variables

u_k^h Equals 1 if rail car $k \in K$ is assigned hub $h \in H$, 0 otherwise.

x_{ik} Equals 1 if laden 20′ container $i \in I_h$ is loaded on (the bottom of) rail car $k \in K$, 0 otherwise.

v_{mk} Equals 1 if empty 20′ container $m \in M_h$ is loaded on (the bottom of) rail car $k \in K$, 0 otherwise.

y_{jkl} Equals 1 if 40′ container $j \in J_h$ is loaded in position $l \in L$ on rail car $k \in K$, 0 otherwise.

z_k Equals 1 if there is at least one laden 20′ container loaded on rail car $k \in K$, 0 otherwise.

τ_k Equals 1 if there is at least one empty 20′ container loaded on rail car $k \in K$, 0 otherwise.

A.4. Model (P)

$$\text{Max } Z = \frac{1}{4|K|}\left(\sum_k \sum_i x_{ik} + \sum_k \sum_m v_{mk} + 2\sum_l \sum_k \sum_j y_{jkl} \right).$$

(A.1)

s.t.

$$\sum_{i \in I_h} x_{ik} \le 2 z_k u_k^h, \quad \forall k \in K, \ h \in H,$$

(A.2)

$$\sum_{k \in K} x_{ik} \le 1, \quad \forall i \in I_h, \ h \in H,$$

(A.3)

$$\sum_{m \in M_h} v_{mk} \le 2 \tau_k u_k^h, \quad \forall k \in K, \ h \in H,$$

(A.4)

$$\sum_{m \in M_h} v_{mk} \le 1, \quad \forall m \in M_h, \ h \in H,$$

(A.5)

$$z_k + \tau_k \le 1, \quad \forall k \in K,$$

(A.6)

$$\sum_{l \in L} \sum_{k \in K} y_{jkl} \le 1, \quad \forall j \in J_h, \forall h \in H,$$

(A.7)

$$\sum_{\forall j \in J_h} y_{jk1} \le u_k^h, \quad \forall k \in K, \ h \in H,$$

(A.8)

$$\sum_{\forall j \in J_h} y_{jk2} \le (1 - z_k) u_k^h, \quad \forall k \in K, \ h \in H,$$

(A.9)

$$\sum_{\forall j \in J_h} y_{jk2} \le (1 - \tau_k) u_k^h, \quad \forall k \in K, \ h \in H,$$

(A.10)

$$\sum_{l \in L} \sum_{j \in J_h} w_j^{40} y_{jkl} + \sum_{i \in I_h} w_i^{20} x_{ik} + \sum_{m \in M_h} w_m^{mt20} v_{mk} \le c_k, \quad \forall k \in K, \ h \in H,$$

(A.11)

$$\sum_{j \in J_h} w_j^{40} y_{jk1} \le \gamma_k \left(\sum_{j \in J_h} w_j^{40} y_{jk2} + \sum_{i \in I_h} w_i^{20} x_{ik} + \sum_{m \in M_h} w_m^{mt20} v_{mk} \right), \quad \forall k \in K, \ h \in H,$$

(A.12)

$$\sum_h u_k^h = 1, \quad \forall k \in K,$$

(A.13)

$$u_k^h \in \{0, 1\}, \quad \forall k \in K, \ h \in H,$$

(A.14)

$$v_{mk} \in \{0, 1\}, \quad \forall k \in K, \ m \in M_h,$$

(A.15)

$$x_{ik} \in \{0, 1\}, \quad \forall i \in I_h, \ k \in K,$$

(A.16)

$$y_{jkl} \in \{0, 1\}, \quad \forall j \in J_h, \ l \in L, \ k \in K,$$

(A.17)

$$z_k \in \{0, 1\}, \quad \forall k \in K, \tag{A.18}$$

$$\tau_k \in \{0, 1\}, \quad \forall k \in K. \tag{A.19}$$

The objective function (A.1) seeks to maximize the utilization of the rail cars. Constraint (A.2) ensures that when rail car k is assigned hub h, one can either load two laden $20'$ containers on the rail car, or none. Constraint (A.3) states that each laden $20'$ container can be loaded on 1 rail car only. Constraints (A.4) and (A.5) are the parallels of constraints (A.2) and (A.3) for *empty* $20'$ containers, whereas constraint (A.6) prohibits the mixing of empties and laden $20'$ containers on a rail car. Constraint (A.7) guarantees that each $40'$ container is assigned to at most one location on the rail cars. Constraint (A.8) states that only when rail car k is assigned hub h, one can load a $40'$ container (i.e., destined for hub h) to the top position on the rail car. Similarly, at most one $40'$ container (i.e., destined for hub h) can be loaded at the bottom of rail car k, if the rail car is destined for hub h and no $20'$ containers are loaded on the rail car, cf. constraints (A.9) and (A.10). In (A.11), the weight capacity restrictions of the rail cars are stated. Constraint (A.12) ensures that the top container is lighter than the bottom container(s) within a tolerance factor of γ_k. For instance, when $\gamma_k = 1.1$, this indicates that the weight of the container on the top on rail car k is at most 10% higher than the weight of the bottom container(s). Constraint (A.13) ensures that each rail car is assigned a hub. Finally, constraints (A.14) to (A.19) state the binary nature of the decision variables. For a more detailed discussion of this model, the reader is referred to Ng and Talley [13].

Data Availability

The data used in the current research is from the port authority we worked with. Due to the non-disclosure agreement, we cannot reveal all the details. However, we would be happy to provide the details to reviewers for review purpose.

Disclosure

The contents of the article remain the sole responsibility of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] Z. Song, W. Tang, and R. Zhao, "Implications of economies of scale and scope for round-trip shipping canvassing with empty container repositioning," *Annals of Operations Research*, pp. 1–31, 2020.

[2] D.-Y. Lin, C.-J. Juan, and M. Ng, "Evaluation of green strategies in maritime liner shipping using evolutionary game theory," *Journal of Cleaner Production*, vol. 279, Article ID 123268, 2021.

[3] Y. Wang and S. Wang, "Deploying, scheduling, and sequencing heterogeneous vessels in a liner container shipping route," *Transportation Research Part E: Logistics and Transportation Review*, vol. 151, Article ID 102365, 2021.

[4] M. Ng, "Vessel speed optimisation in container shipping: a new look," *Journal of the Operational Research Society*, vol. 70, no. 4, pp. 541–547, 2019.

[5] K. Fagerholt and H. N. Psaraftis, "On two speed optimization problems for ships that sail in and out of emission control areas," *Transportation Research Part D: Transport and Environment*, vol. 39, pp. 56–64, 2015.

[6] K. H. Park, Y. T. Chang, and J. S. L. Lam, "Optimal emission control under public port rivalry: a comparison of competitive and cooperative policy," *Maritime Transport Research*, vol. 1, Article ID 100005, 2020.

[7] J. Qi, S. Wang, and C. Peng, "Shore power management for maritime transportation: status and perspectives," *Maritime Transport Research*, vol. 1, Article ID 10000, 2020.

[8] F. Bruns and S. Knust, "Optimized load planning of trains in intermodal transportation," *Spectrum*, vol. 34, no. 3, pp. 511–533, 2012.

[9] D. L. Pacanovsky, C. T. Jahren, R. N. Palmer, R. R. Newman, and D. Howland, "A decision support system to load containers to double-stack rail cars," *Civil Engineering Systems*, vol. 11, no. 4, pp. 247–261, 1995.

[10] Y.-C. Lai, C. P. L. Barkan, and H. Önal, "Optimizing the aerodynamic efficiency of intermodal freight trains," *Transportation Research Part E: Logistics and Transportation Review*, vol. 44, no. 5, pp. 820–834, 2008.

[11] M. Lang, J. Przybyla, and X. Zhou, "Loading containers on double-stack cars: multi-objective optimization models and solution algorithms for improved safety and reduced maintenance cost," 2011, http://www.civil.utah.edu/%7Ezhou/container_scheduling.pdf.

[12] A. Upadhyay, W. Gu, and N. Bolia, "Optimal loading of double-stack container trains," *Transportation Research Part E: Logistics and Transportation Review*, vol. 107, pp. 1–22, 2017.

[13] M. Ng and W. K. Talley, "Rail intermodal management at marine container terminals: loading double stack trains," *Transportation Research Part C: Emerging Technologies*, vol. 112, pp. 252–259, 2020.

[14] H. Zhao and C. Zhang, "An online-learning-based evolutionary many-objective algorithm," *Information Sciences*, vol. 509, pp. 1–21, 2020.

[15] Z.-Z. Liu, Y. Wang, and P. Q. Huang, "AnD: a many-objective evolutionary algorithm with angle-based selection and shift-based density estimation," *Information Sciences*, vol. 509, pp. 400–419, 2020.