An Optimization Approach for the Train Load Planning Problem in Seaport Container Terminals

Daniela Ambrosino, Davide Anghinolfi, Massimo Paolucci, and Silvia Siri

Abstract In this work an optimization approach for defining loading plans for trains in seaport container terminals is presented. The problem consists in defining the assignment of containers of different length, weight and value to wagon slots of a train, in order to maximize the total value loaded on the train and to minimize unproductive movements, both in the stacking area and of the crane during the loading process. Due to the difficulty in solving this problem for real scenarios, a MIP heuristic solution approach based on a randomized matheuristics is proposed. Computational results are presented and discussed, showing the effectiveness of the proposed heuristic solution method.

Keywords Train Loading Problem · MIP heuristics

1 Introduction

The landside transport planning represents a crucial process in seaport container terminals for its impact on congestion and pollution and for the stronger rules imposed to terminals regarding the time spent by import and export containers in the stacking areas. Container terminals have to reach automation and efficiency in their whole organization. Optimization methods have been applied to the decision problems arising in many processes in container terminals, as shown in the surveys by Steenken et al. (2004) and Stahlbock and Voss (2008). This chapter presents an optimization approach for defining loading plans for trains.

D. Ambrosino
Department of Economics and Business Studies (DIEC), University of Genoa, Genova, Italy
e-mail: ambrosin@economia.unige.it

D. Anghinolfi · M. Paolucci (✉) · S. Siri
Department of Informatics, Bioengineering, Robotics, and Systems Engineering (DIBRIS), University of Genoa, Genova, Italy
e-mail: davide.anghinolfi@unige.it; massimo.paolucci@unige.it; silvia.siri@unige.it

J. E. Hernández et al. (eds.), *Collaborative Logistics and Intermodality*,
https://doi.org/10.1007/978-3-030-50958-3_7

Some research studies focus on train load planning problems in landside intermodal terminals. Bostel and Dejax (1998) propose some models and heuristics for container allocation problems on trains arising in rail-rail terminals with rapid transfer yards. Corry and Kozan (2006, 2008) face the problem of assigning containers to train slots considering, in their second work, different types of containers and loading patterns and minimizing the weighted sum of number of wagons and equipment working time. They propose metaheuristics, such as local search and simulated annealing, to solve the problem in practical applications. Only in a more recent work Bruns and Knust (2012) consider real weight constraints for wagons.

Studies on the Train Load Planning Problem (TLPP) arising in seaports started with Ambrosino et al. (2011) that, inspired by Bruns and Knust (2012), extended their models for a sequential train load by including the reshuffles in the stacking area, a crucial aspect in maritime container terminals. A more general train loading model, not imposing sequential loading but allowing also unproductive movements of the crane, is presented in Ambrosino et al. (2013) and in Ambrosino and Siri (2014). In the first work this general model is used for comparing different train loading policies and stacking strategies. In the second work different models are analyzed to identify the most suitable one for solving real problems in maritime container terminals (i.e., providing good and applicable solutions in an acceptable CPU time). Due to the difficulty in solving the model presented in Ambrosino and Siri (2014) for real scenarios, this chapter proposes a Mixed Integer Programming (MIP) heuristic approach consisting in a Randomized Neighborhood Search (RANS) algorithm firstly introduced by Anghinolfi and Paolucci (2011) and successfully applied to different logistic problems by Ambrosino et al. (2011) and by Anghinolfi and Paolucci (2014).

The chapter is organized as follows. Section 2 introduces the problem and its mathematical formulation. Section 3 describes the MIP heuristic approach, Sect. 4 reports the experimental results and conclusions are drawn in Sect. 5.

2 Problem Definition and Mathematical Formulation

The considered problem is inspired by a real case of an Italian port. The TLPP considers only one train at a time and, assuming shuttle trains directed to an inland port, neglects the destination of containers. Containers are characterized by length, weight and a priority value reflecting their importance. Containers that must be loaded on trains are stored in stacks of different heights in a specific stacking area close to the railway yard. From there, they are moved near the tracks with trailers and finally loaded on a train by a crane. Each train is composed of a set of wagons of different types, i.e., with different length and weight capacity, different possible configurations in terms of number and length of slots and weight capacity of each slot (more details are given in Ambrosino and Siri (2014)). The assignment of containers to wagon slots accounts for length and weight constraints.

The crane usually starts loading the train from the first wagon and goes on along the train without changing direction. Consequently, reshuffles may occur when the crane needs to load on the currently served wagon a container that is located below other containers in a stack. Another possibility, which allows to reduce reshuffles in the yard, is that the crane proceeds to the next wagon leaving a slot free, and it moves back later if a suitable container for such slot becomes available on the top of a stack; this latter movement represents an unproductive operation of the crane.

Unproductive operations, i.e., reshuffles in the stacking area and unproductive moves of the crane, represent a cost for the terminal and slow down the loading operations. For this reason, when dealing with the TLPP, they must be minimized. Moreover, the load plan should be realized in order to maximize the train utilization; in this chapter this goal is achieved by maximizing the total value of loaded containers.

The MIP formulation proposed in Ambrosino and Siri (2014) for solving the TLPP described above is here briefly reported. First of all, let us introduce the notation used in the multi-objective MIP model.

Let C denote the number of containers in the stacking area, W the number of wagons of the train and S the number of train slots. For each container $i = 1,\ldots,C$, w_i is the weight (expressed in tons), λ_i the length (i.e., 20′ or 40′), π_i the value. As for the length, containers are stored in homogeneous stacks of 20′ or 40′. The relative position of containers in the stacks is given by $\gamma_{i,j}$, $i, j \in \{1,\ldots,C\}$, where $\gamma_{i,j} = 1$ if container i is over container j in a stack, and $\gamma_{i,j} = 0$ otherwise. Let Q represent the height of the stacks (i.e., the maximum number of tiers). For each wagon $\omega = 1,\ldots,$ W, S_ω is the subset of slots, B_ω the subset of weight configurations, and ϖ_ω the weight capacity. Moreover, $B_{s,\omega}$ is the subset of weight configurations for slot s of wagon ω, μ_s the length of slot s (i.e., 20′ or 40′), ρ_s the position of slot s in the train (expressed in TEUs) with respect to the first slot of the first wagon, $\delta_{b,s}$ the weight capacity of slot s in the weight configuration b, and $\overline{\Omega}$ the weight capacity of the train. When loading the train, the maximum number of loading operations (T) is equal to the TEU capacity of the train, which corresponds to loading only 20′ containers. The actual number of operations executed depends on the cargo composition (number of 20′ and 40′ containers loaded) and it is equal to the number of containers loaded on the train, that is not greater than T.

Finally, α and β are, respectively, the unitary reshuffling and crane movement costs.

The decisions are related to:

- The choice of a configuration b for each wagon ω (variables $f_{\omega,b} \in \{0,1\}$, $\omega = 1,\ldots,W$, $b \in B_\omega$);
- The assignment of a container i to a slot s at operation t (variables $x_{i,s,t} \in \{0,1\}$, $i = 1,\ldots, C$, $s = 1,\ldots, S$, $t = 1,\ldots,T$); obviously, container i can be assigned to slot s only if $\lambda_i = \mu_s$.

The number of reshuffles is accounted by means of variables $y_{i,j} \in \{0,1\}$ defined for $i, j \in \{1,\ldots,C\}$ such that $\gamma_{i,j} = 1$ then, $y_{i,j} = 1$ if container i is reshuffled to load container j. Variables $z_t \geq 0$, $t = 2,\ldots,T$ model the unproductive distance traveled by the crane due to operation t; assuming the crane initially positioned over the first wagon (e.g., on the left end of the train), z_t equals the distance in TEUs covered backward by the crane (e.g., from right to left) between operation $t-1$ and t. Finally, technical variables $u_t \geq 0$, $t = 2,\ldots,T$ are introduced for not computing the return of the crane to its starting position after the last loading operation as an unproductive movement. The adopted MIP formulation is the following:

$$\min \alpha \cdot \sum_{\substack{i,j\in\{1,\ldots C\}:\\ \gamma_{i,j}=1}} y_{i,j} + \beta \cdot \sum_{t=2}^{T} z_t - \sum_{i=1}^{C} \pi_i \cdot \sum_{s=1}^{S} \sum_{t=1}^{T} x_{i,s,t} \tag{1}$$

$$\text{s.t.} \quad \sum_{s=1}^{S} \sum_{t=1}^{T} x_{i,s,t} \leq 1 \quad i = 1, \ldots, C \tag{2}$$

$$\sum_{i=1}^{C} \sum_{s=1}^{S} x_{i,s,t} \leq 1 \quad t = 1, \ldots, T \tag{3}$$

$$\sum_{i=1}^{C} \sum_{t=1}^{T} x_{i,s,t} \leq 1 \quad s = 1, \ldots, S \tag{4}$$

$$\sum_{b\in B_\omega} f_{w,b} = 1 \quad \omega = 1, \ldots, W \tag{5}$$

$$\sum_{i=1}^{C} \sum_{t=1}^{T} w_i \cdot x_{i,s,t} \leq \sum_{b\in B_{s,\omega}} \delta_{b,s} \cdot f_{\omega,b} \quad \omega = 1, \ldots, W \quad s \in S_\omega \tag{6}$$

$$\sum_{i=1}^{C} \sum_{s\in S_\omega} \sum_{t=1}^{T} w_i \cdot x_{i,s,t} \leq \varpi_\omega \quad \omega = 1, \ldots W \tag{7}$$

$$\sum_{i=1}^{C} \sum_{s=1}^{S} \sum_{t=1}^{T} w_i \cdot x_{i,s,t} \leq \overline{\Omega} \tag{8}$$

$$\sum_{s=1}^{S} \sum_{t=1}^{T} t \cdot x_{i,s,t} - \sum_{s=1}^{S} \sum_{t=1}^{T} t \cdot x_{j,s,t} \leq T \cdot y_{i,j} + T\left(\sum_{s=1}^{S} \sum_{t=1}^{T} x_{i,s,t} - \sum_{s=1}^{S} \sum_{t=1}^{T} x_{j,s,t}\right) \quad i,j \in \{1,\ldots C\} : \gamma_{i,j} = 1 \tag{9}$$

$$z_t \geq \sum_{i=1}^{C} \sum_{s=1}^{S} \rho_s \cdot x_{i,s,t-1} - \sum_{i=1}^{C} \sum_{s=1}^{S} \rho_s \cdot x_{i,s,t} - u_t \quad t = 2, \ldots T \tag{10}$$

$$u_t \leq T\left(\sum_{i=1}^{C} \sum_{s=1}^{S} x_{i,s,t-1} - \sum_{i=1}^{C} \sum_{s=1}^{S} x_{i,s,t}\right) \quad t = 2, \ldots T \tag{11}$$

The objective function (1) minimizes a weighted sum of costs corresponding to reshuffles in the stacking area and unproductive crane movements, and maximizes the total value of the loaded containers. Constraints (2)–(4) regard the assignment of containers to train slots. Thanks to (2), each container can be assigned at most to one slot; constraints (3) require that at most one container-slot assignment is done for each operation, and (4) guarantee that at most one container can be loaded in each slot. Constraints (5)–(8) impose weight restrictions. In particular, for each wagon, a given weight configuration must be chosen, as imposed by (5), and constraints (6), (7) and (8) represent the weight capacity constraints for each slot, each wagon and for the whole train, respectively. Constraints (9)–(11) ensure that the reshuffling variables $y_{i,j}$ and the variables z_t and u_t related to the crane movement are correctly computed. It is important to remember that container i is re-handled if, when operation t is executed, a container j, located in the stacking area under i, is loaded on the train while container i has not yet been loaded. Thus, in constraints (9), if, for a pair of containers i and j such that $\gamma_{i,j} = 1$, j is loaded before i, the left hand side assumes a positive value, forcing variable $y_{i,j}$ to be positive; note that the second term in the right hand side of (9) is used for not considering the loading of i as a cause of reshuffling if container j is not loaded on the train.

2.1 A Simple Initialization Heuristics

Since in some test cases it has been observed that the MIP solver hardly finds a first feasible solution different from the trivial one (which corresponds to load nothing on the train), a simple procedure to generate a non trivial starting solution has been designed. Such a procedure assigns one container per wagon so that unproductive movements of the crane, as well as reshuffles in the storage area, do not occur. The procedure considers the wagons in sequence and iteratively scans the top of the stacks of containers in the stacking area searching for a container compatible (i.e., for length and weight) with the available slots on the current wagon. Whenever such a container is found, it is removed from the stack and assigned to the relevant wagon slot and the next wagon is considered. If a container compatible with the current wagon is not found, the wagon remains empty and the procedure goes on to consider the next wagon. Then, the initial solution loads a number of containers equal to the number of wagons in the best case, while it loads nothing (trivial feasible solution) in the worst case. Anyway, in the experimental tests, the worst case never occurs when applying the initialization heuristics.

3 The MIP Heuristic Solution Approach

Constraints (2)–(4) and (7) of the TLPP model are typical constraints of the Generalized Assignment Problem, which is a classical combinatorial optimization problem known to be NP-hard. Thus also the TLPP is NP-hard. Due to the difficulty in solving this model, the solution approach based on the RANS heuristics introduced by Anghinolfi and Paolucci (2011) is here described and applied to the TLPP.

The RANS heuristics is a simple iterative search algorithm that starts from a first feasible incumbent solution x^c for the original MIP problem and iterates the following three steps until the maximum time limit is reached:

1. Variables fixing by random choices. A partially fixed MIP sub-problem is defined by fixing the values of a subset of k randomly selected binary and integer variables equal to the ones in x^c. The parameter k is initialized equal to the 10% of total number of binary and integer variables.
2. Local search. The sub-problem is solved by a MIP solver fixing t_{mip} as maximum allowed time. The t_{mip} parameter is set equal to $\max\{T_{\min}, 2 \cdot t_{rel}\}$, where $T_{min} = 30$ s and t_{rel} is the time needed to solve the linear relaxation of the original problem. If a new best solution is found, the incumbent x^c is updated.
3. Parameter adjustment. If a new best solution is found in at most $t_{mip}/2$, then k is increased as $k = k \cdot 1.1$; otherwise k is reduced as $k = k \cdot 0.9$ and a new iteration starts.

RANS operates at the higher level as an iterated local search: steps 1 and 3 define the area in the solution space that is explored in step 2 by a local search. The solution neighbourhood used by RANS is randomly defined by hard fixing a subset of incumbent variable values. The dimension of such neighbourhood is controlled by k that is adjusted depending on the experienced difficulty in solving sub-problems (if the condition in step 3 is verified, the sub-problem is considered easily solved). In this way the exploration is terminated in a reasonable short time and the choice of the initial value of k becomes not critical. The self-tuning mechanism used for k makes also the choice of the T_{min} value not critical, since it allows reducing the neighbourhood size so that the sub-problems can be easily solved. This kind of self-tuning used for k is similar to the adaptation of the fraction of variables to be hard fixed in the mutation phase of the polishing MIP heuristics (Rothberg 2007).

4 Experimental Analysis

The proposed approach has been evaluated by considering 30 instances, randomly generated with reference to a real Italian case study. These instances are divided in six groups (A,. . .,F) that differ for the number of containers stored in the yard (30-40-50) and for the height of the stacks in the yard ($Q = 4$ or 6). In each instance, 60% of containers are 20′ long. 20′ containers have a π value equal to 10, 15 or 20, randomly assigned with equal probabilities, whereas these values are doubled for 40′ containers. 20′ container weights, expressed in tons, are uniformly distributed in $U[6, 24]$, whereas 40′ container ones in $U[10, 30]$. Containers are stored in stacks in accordance with their length. The train is composed of 15 wagons and its maximum load is 900 tons. The wagon composition of the train is randomly generated assuming three types of wagons with different maximum weight, available slots and number of alternative configurations. The train capacity (T) ranges from 33 to 42 TEUs. In the computational tests α and β are fixed to 5 in order to better represent the real operative scenario, in which the cost of these unproductive movements is almost equivalent. It is important to note that these weights have to be defined having in mind that the main aim is to load the train; some preliminary tests for tuning these weights have been executed.

RANS has been implemented in C++ on a 2.4GHz Intel Core 2 Duo E6600, 4GB RAM notebook, and Cplex 12.5 is the MIP solver used.

Table 1 shows the dimensions of the instances in terms of number of variables and constraints of the proposed model and the values of the objective function obtained by the MIP solver for different CPU time limits, i.e., 600 s, 1200 s, 1800 s, 3600 s and 14,400 s.

From Table 1 it is apparent that the objective (1) cannot assume values strictly greater than zero, as these are dominated by the trivial zero-cost solutions corresponding to not loading any container on the train. For shorter computational times the MIP solver was not able to find a solution different from the trivial one for several instances and even after 4 h of computation only the zero-cost solution was found for instance 26.

Figure 1 shows the behaviour in time of the three components of the objective function produced by the MIP solver together with the trend of the optimality gap for the six groups of instances.

The trend of the most relevant objective, i.e., the value of the loaded containers, is non-decreasing, whereas a non-monotonic trend can be observed for the cost of reshuffling and of unproductive movements of the crane. This is due to the fact that a possible increase in such components may lead to an improvement of the overall objective. This is also confirmed by the behaviour of the optimality gap that is monotonically decreasing in time. However, the considered instances appear difficult to solve since the gap for most of the instance groups is quite high even after 4 h of computation.

Table 2 compares the average results obtained by RANS over 5 runs with the ones of the MIP solver, fixing for both methods a maximum computation time which ranges from 600 to 3600 s.

Table 2 provides the percentage deviations of the results of RANS (computed as 100·(*RANS_obj – MIPsolver_obj)/|MIPsolver_obj*|) aggregated for the six groups of instances and, in the last row, shows the average percentage deviations over all the set of instances. Table 2 clearly shows the effectiveness of RANS, which was able to find better solutions even with a short computation time. Then, Table 3 summarizes the percentage deviations of the RANS results after 10 min with respect to the MIP solver ones for all the different available computation times. Here the values in boldface denote a prevalence in the average of RANS over the MIP solver, and it is easy to note that only for three instance groups and after 4 h of computation the MIP solver provided better results.

Further tests investigated the quality of the solution generated by the initialization heuristics and the possible benefits of this initialization for the MIP solver. Table 4 shows the percentage deviations of the MIP solver and RANS results with respect to

Table 1 The MIP solver results

Instances	Variables	Constraints	600 s	1200 s	1800 s	3600 s	14,400 s
1	53,154	373	–100	–370	–415	– 477.5	–517.5
2	36,059	362	–295	–600	–627.5	–665	–642.5
3	25,807	318	–470	–515	–515	–555	–555
4	38,969	351	–237.5	–270	–280	–407.5	–597.5
5	33,190	340	–300	–397.5	–502.5	–520	–510
6	30,772	372	–232.5	–140	–140	–505	–505
7	62,400	438	0	–190	–110	–150	–470
8	36,092	394	–120	–195	–390	–467.5	–530
9	33,386	383	–497.5	–502.5	–550	–565	–565
10	55,185	394	–77.5	–80	–315	–307.5	–375
11	29,125	356	–552.5	–605	–605	–605	–605
12	47,991	389	–280	–505	–510	–552.5	–552.5
13	40,909	367	–330	–450	–515	–405	–405
14	31,108	367	–550	–610	–610	–610	–610
15	55,617	378	0	–125	–142.5	–402.5	–457.5
16	55,651	414	–100	–100	–110	–337.5	–557.5
17	48,026	425	–130	–130	–130	–192.5	–540
18	48,294	403	0	–60	–65	–225	–535
19	55,644	403	0	–97.5	–115	–377.5	–430
20	51,762	436	–85	–205	–210	–210	–407.5
21	48,210	368	–50	–180	–302.5	–530	–530
22	59,534	379	–262.5	–300	–362.5	–415	–585
23	83,495	401	0	–200	–260	–277.5	–492.5
24	69,455	401	0	0	–285	–370	–297.5
25	100,716	467	–40	–40	–40	–150	–447.5
26	65,817	438	0	0	0	0	0
27	63,887	449	0	–70	–400	–440	–422.5
28	59,968	460	–80	–80	–302.5	–412.5	–375
29	64,633	471	–100	–100	–247.5	–367.5	–520
30	51,120	438	–225	–225	–267.5	–557.5	–562.5

the initial solution, pointing out that, although quite simple, the initialization heuristics produced average results better than the ones of the MIP solver after half an hour of computation for groups B and F, and even after an hour for groups D and E.

Table 5 provides an overall comparison of the RANS results after 600 s with the ones produced after 1 h by the MIP solver with and without the heuristics starting solution initialization. The columns of Table 5 show the average values for the six groups of instances for the loaded value (L), the number of reshuffles (R), the number of unproductive crane movements (U) and the percentage TEU occupancy on the train (O). In addition, Table 5 shows, for the initialized MIP and the RANS, the percentage deviation of the objective values from the MIP solver ones (D). The heuristics initialization produced an overall benefit in the objective function results obtained by the MIP solver, even if there is a worsening both in the overall

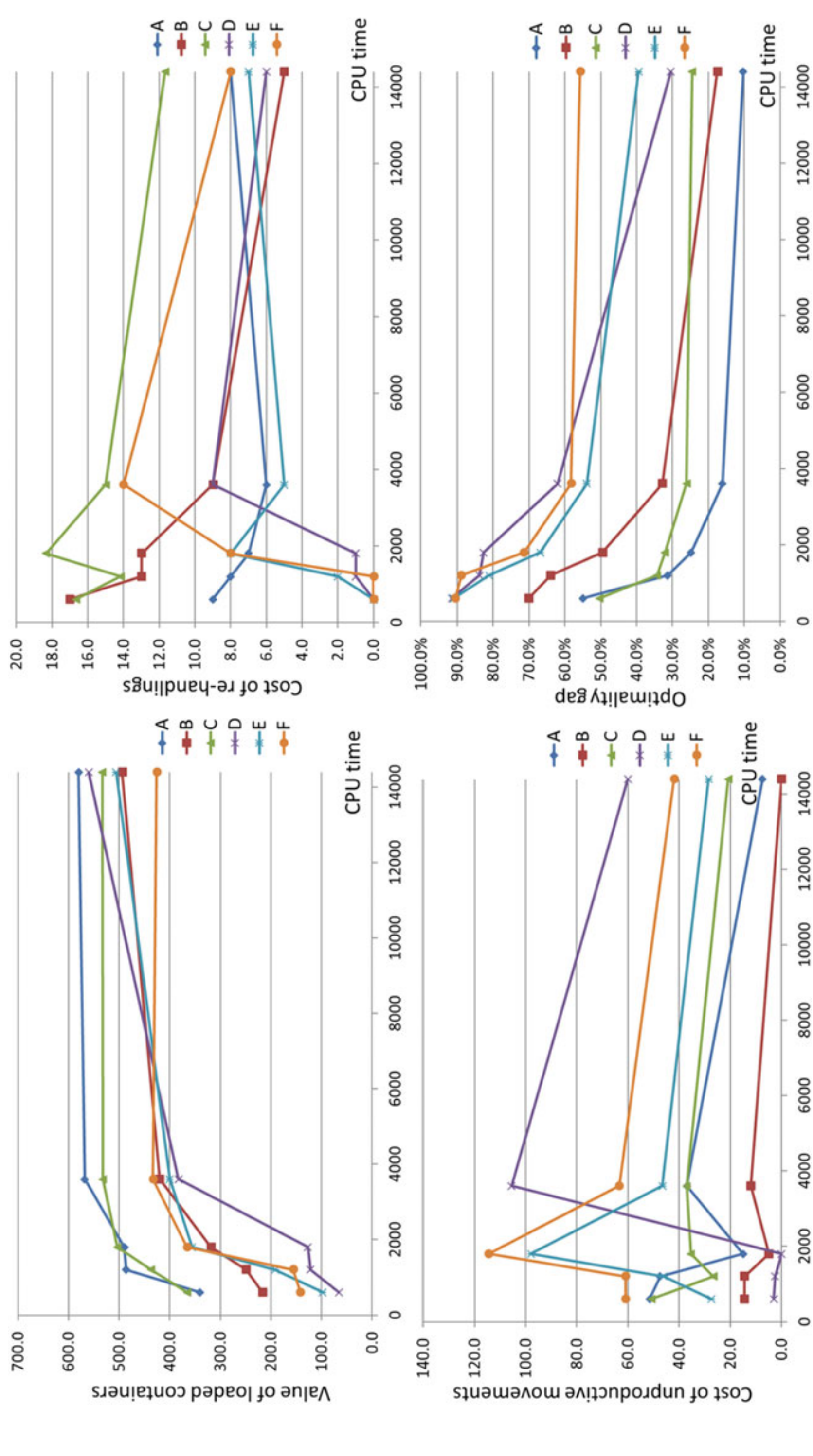

Fig. 1 The trends of the three objectives and the optimality gap obtained by the MIP solver

Table 2 Percentage deviations between RANS and MIP solver results

Groups	600 s	1200 s	1800 s	3600 s
A	–137.40	–33.80	–24.24	–6.36
B	–175.09	–175.19	–116.62	–46.76
C	–40.33	–69.34	–56.65	–15.63
D	–360.29	–370.59	–346.60	–113.16
E	–814.71	–446.06	–340.59	–97.51
F	–400.86	–459.51	–92.18	–35.51
Averages	–285.28	–245.25	–165.25	–53.07

Table 3 Percentage deviations of the RANS results after 600s w.r.t. the MIP solver results

Groups	600 s	1200 s	1800 s	3600 s	14,400 s
A	**–137.40**	**–30.55**	**–20.32**	**–1.55**	7.91
B	**–175.09**	**–153.31**	**–94.21**	**–28.09**	14.35
C	**–40.33**	**–60.66**	**–47.65**	**–7.55**	**–4.62**
D	**–360.29**	**–349.76**	**–315.58**	**–86.61**	5.06
E	**–814.71**	**–459.69**	**–345.90**	**–89.69**	**–19.48**
F	**–400.86**	**–463.37**	**–85.75**	**–25.97**	**–19.45**
Averages	**–285.28**	**–237.98**	**–153.84**	**–40.39**	**–2.13**

Table 4 Percentage deviations of the initialization heuristic solutions w.r.t. the MIP solver and RANS results

	MIP Solver				RANS
Groups	600 s	1200 s	1800 s	3600 s	600 s
A	–44.97	17.78	24.14	35.55	35.82
B	–100.65	–87.40	–49.89	2.51	24.57
C	9.68	10.62	17.36	34.61	38.93
D	–190.41	–177.10	–156.76	–20.52	37.34
E	–428.49	–202.15	–134.66	–3.38	42.86
F	–227.92	–260.22	–19.62	19.29	36.44
Averages	–142.23	–108.21	–54.40	11.07	35.99

loaded value and in the percentage occupancy of the train. On the other hand, the average RANS results are the best for the overall objective, the loaded value and train occupancy, paying this with a worse value for the two less important objectives. Finally, note that the 5% confidence interval for the average percentage deviations of RANS results in 600 s from MIP solver results in 1 h is [–66.49%, –14.29%], whereas from the initialized MIP solver is [–36.85%, –20.33%], thus denoting that in both cases RANS produced, on average, statistically significant better results.

Although the dimensions assumed for the yard in the previous experimental analysis are representative of the real case study used as reference, two final tests were performed in order to evaluate the ability of the proposed method to scale for larger yards. Therefore, two additional scenarios, denoted by *Medium* (M) and *Large* (L), respectively characterized by 100 and 500 containers in the yard, were randomly generated. The MIP model for scenario M includes 167,395 variables and 635 constraints, whereas the one for scenario L presents 675,887 variables and 2046 constraints. Table 6, analogously to Table 5, shows the comparisons between the

Table 5 The overall comparison of the MIP solver results with the RANS ones

	MIP solver (3600 s)				Initialized MIP solver (3600 s)					RANS (600 s)				
Grp	L	R	U	O (%)	L	R	U	O (%)	D (%)	L	R	U	O (%)	D (%)
A	518	1.4	1.9	87.31	448	0	0.4	75.38	13.41	537	8	7.5	90.55	−1.55
B	431	3	4.6	75.26	483	1.6	0.3	68.43	−14.66	532	22	96	93.24	−28.09
C	577	3.8	6.3	92.68	502	1.4	0.8	70.79	14.74	552	16	35.5	90.52	−7.55
D	396	1.6	14.7	68.03	405	3.8	1.8	70.18	−54.64	502	10	15	84.62	−86.61
E	438	2	6.7	68.22	365	0.6	1.5	60.87	−18.79	573	8	20.5	92.50	−89.69

Table 6 The comparison between MIP solver and RANS results for the larger yard scenarios

	MIP solver (3600 s)				Initialized MIP solver (3600 s)					RANS (600 s)				
Scenario	L	R	U	O (%)	L	R	U	O (%)	D (%)	L	R	U	O (%)	D (%)
M	30	0	0	8.11	280	0	0	54.05	−833.33	640	10	70	100.00	−1766.67
L	120	0	0	15.79	407	0	0	63.16	−239.17	760	0	55	100.00	−487.50

results obtained for the two new scenarios by the MIP solver (with and without initialization) in 3600 s and RANS in 600 s. The greater difficulty in finding good solutions for these new scenarios is highlighted by the low train occupancy levels obtained by the MIP solver. Better results are produced when the solver started from the solutions generated by the initialization heuristics, even if the final levels of occupancy of the train capacity are still quite unsatisfying. Note that the improvement obtained in 1 h by the MIP solver with respect to the starting solutions are only 3.70% for scenario M and 33.44% for scenario L. On the other hand, even for these larger cases, RANS shows its ability to find high quality results in an acceptable short time, in particular being able to exploit all the available train load capacity.

5 Conclusions

This chapter discusses a solution approach based on a MIP heuristics to the MIP model proposed for the train load planning problem at seaport terminals. Such heuristics performs a randomized iterative local search exploring a sequence of solution neighbourhoods by defining and solving MIP sub-problems. Experimental tests performed on a set of random instances, generated with reference to a real terminal context, showed the difficulty in solving the presented MIP model and the effectiveness of the proposed heuristic method to find good solutions in an acceptable computation time.

References

Ambrosino, D., & Siri, S. (2014) Models for train load planning problems in a container terminal. In J. Freire de Sousa & R. Rossi (Eds.), *Computer-based modelling and optimization in transportation*, Advances in intelligent systems and computing. Cham: Springer.

Ambrosino, D., Bramardi, A., Pucciano, M., Sacone, S., & Siri, S. (2011). Modeling and solving the train load planning problem in seaport container terminals. In *Proceedings of the 7th IEEE conference on automation science and engineering* (pp. 208–213).

Ambrosino, D., Caballini, C., & Siri, S. (2013). A mathematical model to evaluate different train loading and stacking policies in a container terminal. *Maritime Economics & Logistics, 15*, 292–308.

Anghinolfi, D. & Paolucci, M. (2011, May 4–6). A new MIP heuristic based on randomized neighborhood search. In *Proceedings of the VII ALIO–EURO—Workshop on applied combinatorial optimization, Porto, Portugal* (pp.84–87).

Anghinolfi, D., & Paolucci, M. (2014). A general purpose Lagrangian heuristic applied to the train loading problem. *Procedia—Social and Behavioral Sciences, 108C*, 37–46. Elsevier.

Anghinolfi, D., Paolucci, M., Sacone, S., & Siri, S. (2011). Freight transportation in railway networks with automated terminals: a mathematical model and MIP heuristic approaches. *European Journal of Operational Research, 214*(3), 588–594.

Bostel, N., & Dejax, P. (1998). Models and algorithms for container allocation problems on trains in a rapid transshipment shunting yard. *Transportation Science, 32*, 370–379.

Bruns, F., & Knust, S. (2012). Optimized load planning of trains in intermodal transportation. *OR Spectrum, 34*(3), 511–533.

Corry, P., & Kozan, E. (2006). An assignment model for dynamic load planning of intermodal trains. *Computers & Operations Research, 33*, 1–17.

Corry, P., & Kozan, E. (2008). Optimised loading patterns for intermodal trains. *OR Spectrum, 30*, 721–750.

Rothberg, E. (2007). An evolutionary algorithm for polishing mixed integer programming solutions. *INFORMS Journal on Computing, 19*, 534–541.

Stahlbock, R., & Voss, S. (2008). Operations research at container terminals: A literature update. *OR Spectrum, 30*, 1–52.

Steenken, D., Voss, S., & Stahlbock, R. (2004). Container terminal operation and operations research—A classification and literature review. *OR Spectrum, 26*, 3–49.