Modeling and solving the train load planning problem in seaport container terminals

Daniela Ambrosino, Andrea Bramardi, Marco Pucciano, Simona Sacone and Silvia Siri

Abstract—In this paper we present two mathematical formulations and a heuristic approach for the train load planning problem of import containers at a seaport container terminal. The problem consists of determining how to assign a set of containers of different length and weight to the wagons of a train in order to satisfy capacity constraints of both the wagons and the train, while minimizing the rehandling operations in the storage area where containers are waiting for being loaded on trains and maximizing the train utilization. Some computational results will be reported in the paper in which the heuristic approach is compared with the solution of the mathematical programming formulation.

I. INTRODUCTION AND LITERATURE REVIEW

Nowadays several transportation systems are used to meet the transport demand for containers (i.e. containers are shipped by trucks, trains and vessels). Containers are transshipped from one mode of transportation to another one either at ports or at container terminals. The transshipment process must be characterized by quickness and efficiency: it is affected by the different type of handling equipment used, by the control system and by the degree of coordination among actors operating in the terminal.

Container terminals are therefore very complex systems that require the development of quantitative methods to support the relevant decisions. In [1] the authors distinguish among three levels of planning and control in making decisions to obtain an efficient terminal: the strategic, tactical and operational level. The authors present an overview on decision problems arising at the three different levels. Other recent surveys [2] and [3] are focused on operations research methods applied to container terminals; the authors divide the developed optimization approaches according to the different processes in a seaport terminal: ship planning (i.e. berth allocation, stowage planning and crane split), storage and stacking planning, and transport optimization (divided in quayside, landside, and crane movements). With respect to this classification, this work is devoted to landside transport optimization and presents an optimization approach for the definition of loading plans for trains.

As highlighted in [2], a loading plan indicates on which wagon a container has to be placed; this decision generally depends on the destination, type and weight of the container, the maximum load of the wagon and the train composition. Also the container location in the storage area can influence the loading plan. Moreover, the loading plan concerns operational decisions that obviously are affected by strategic and tactical ones regarding both how the inter-terminal transport of containers has been organized (that is the handling equipment for the storage area and for the trains) and how the stowage area is managed (i.e. the stowage strategy used).

We consider the case in which the loading plan is performed by the terminal operator with the aim of optimizing both the pick-up operations in the storage area where containers are waiting for being loaded on trains and the load of each train. In a recent work [4] related to a distributed agent system for the port planning and scheduling, the train schedule for container transportation and the yard storage are considered but no details are reported for this problem.

In the literature, few research studies have been devoted to the load planning problem and these studies are referred to landside intermodal terminals rather than to seaports. In [5] the load train is considered as a subproblem when decomposing the scheduling problems arising during operations of a transshipment yard. In [6] the authors propose some models and heuristic methods for container allocation problems on trains, referring to rail-rail terminals with rapid transfer yards, whereas in [7] the authors consider a terminal where containers are transferred to and from trucks on a platform adjacent to the rail tracks provided with a short-term storage area. They propose several techniques for defining the assignment of containers to slots of a train while minimizing container handling time and optimizing the weight distribution of the train. They solve the load plan problem considering only one type of container and without including in the model weight restrictions for the wagons. In a following work [8] they include more types of containers and the objective regards the minimization of the train length.

The load planning problem in intermodal terminals is also studied in [9]; the authors propose three different integer linear programming formulations for solving the problem of loading containers on wagons of a train by choosing the configuration of the wagons in order to maximize the utilization of the train and minimize total costs including the transportation costs for loading containers and set up costs for changing the configuration of wagons. Many types of containers have to be loaded and many weight restrictions related to wagons configurations are considered. The authors use different solvers for the IPs models.

In this paper we propose two mathematical formulations and a solution approach for solving the load planning problem in a seaport terminal (i.e. only import containers are
considered) with the aim of minimizing the loading time and maximizing the train utilization. The paper is organized as follows: in Section II we properly define the problem under investigation; in Section III two mathematical formulations are proposed and then solution approaches are described in Section IV. Experimental results and some conclusive remarks are given in Section V.

II. PROBLEM DEFINITION

The problem under investigation is the train load planning of import containers on one track. This is the problem of seaport container terminals when they have to plan the train loading for shuttle trains directed to the inland port. Containers are stocked in the yard following a very simple policy (depending for instance on their weight and length); the destination of containers is not taken into account in the load planning problem because it is the same for all containers: the inland terminal. Moreover, the planning problem considers only one track, i.e. only one train at a time.

The present work is inspired from a real case of an Italian port but it can be easily extended to many other cases. The considered case study refers to a container terminal in which the railway yard works as sketched in Fig. 1. The transfer of containers from the stocking area (where they are stacked in blocks of four rows and until the forth tier) to the train is realized as follows: a reach stacker takes a container from the stocking area and puts it on a tractor; then, the tractor moves it near the railway tracks where it is taken by an overhead traveling crane that loads it on the train.

![Fig. 1. A sketch of the railway yard.](image)

In the present train load planning problem, we suppose that the overhead traveling crane loads the train sequentially (starting from the first wagon onwards) and some rehandling operations in the stocking area are allowed. Rehandles are unproductive movements affecting the time and cost of loading operations, so they must be minimized. Thus, more formally, the problem under investigation can be defined as follows: given a set of containers characterized by different weight, length, commercial value and stowage position in the yard, given a set of wagons of a train characterized by weight and length capacity, the problem is to determine how to assign containers to wagons in order to satisfy the capacity constraints, while maximizing the train utilization and minimizing the rehandling cost in the stoking area.

When dealing with real loading problems, the real weight constraints are stricter than simply considering a maximum weight capacity for each wagon and train. Generally, more than one container can be placed on board of a wagon and different weight restrictions have to be satisfied depending on the physical configurations of wagons. Details on different wagon structures are reported in [9].

Just to give an idea of the different possible configurations that may arise for a particular type of wagon, in Fig. 2 a real load pattern is reported (this is a real one used in the Italian freeways). In this case there are three possible “load configurations” (corresponding respectively to three 20’ containers, two 20’ containers, and one 40’ container with one 20’ container). For each load configuration, a set of “weight configurations” is defined, indicating the maximum weight for each slot. For example, for the case of loading a 40’ container together with a 20’ one, there are nine different weight configurations permitted (if for instance the 40’ container has a weight of 25 t, the weight of the 20’ container must be no more than 26 t). Thus the problem includes also a load configuration and a weight configuration decision for each wagon.

![Fig. 2. An example of a real load pattern.](image)

III. MATHEMATICAL FORMULATIONS

In this section we present two different formulations (in the following denoted as M1 and M2) for the train loading problem (TLP). Let us introduce the following notation:

- \( \mathcal{C} \) set of containers in the stocking area;
- \( \mathcal{W} \) set of wagons (of the train under investigation);
- \( \lambda_i \) length of container \( i \in \mathcal{C} \);
- \( \omega_i \) weight of container \( i \in \mathcal{C} \);
- \( \pi_i \) cost of not loading container \( i \in \mathcal{C} \), that takes into account urgency and importance of the container (this term can be considered as the container commercial value);
- \( \gamma_{i,j} \), \( i,j \in \mathcal{C}, i \neq j \), relative position of each container with respect to the others; in particular, \( \gamma_{i,j} = 1 \) indicates that container \( i \) is located below \( j \) (of course, this happens only if \( i \) and \( j \) are in the same stack); otherwise it is \( \gamma_{i,j} = 0 \);
- \( \alpha \) unitary rehandling cost in the stocking area;
- \( T \) maximum number of tiers;
- \( L_w \) length of wagon \( w \in \mathcal{W} \);
• $\overline{W}$, weight capacity of wagon $w \in \mathcal{W}$;
• $\overline{T}$, weight capacity of the train.

The cost terms are defined so that the costs $\pi_i$, $i \in \mathcal{C}$, of not loading containers are always much higher than the cost $\alpha$; this assures that the train is loaded as much as possible. The first formulation M1 concerns the planning problem of loading containers on trains in which each wagon can be different from the others in terms of length and weight capacity, while M2 concerns the more complex problem in which wagons can be used in different load and weight configurations (as those reported in Fig. 2).

Let us introduce the first formulation with the following decision variables:

• $x_{i,w} \in \{0,1\}$, equal to 1 if container $i$ is assigned to wagon $w$;
• $y_{i,w} \in \{0,1\}$, equal to 1 if container $i$ is rehandled when wagon $w$ is loaded.

The TLP can be then stated with the following 0-1 linear programming formulation.

**Model 1 (M1):**

$$\min \sum_{i \in \mathcal{C}} \sum_{w \in \mathcal{W}} \alpha \cdot y_{i,w} + \sum_{i \in \mathcal{C}} \pi_i \cdot \left(1 - \sum_{w \in \mathcal{W}} x_{i,w}\right)$$  \hspace{1cm} (1)

s.t.

$$\sum_{w \in \mathcal{W}} x_{i,w} \leq 1 \quad i \in \mathcal{C}$$  \hspace{1cm} (2)

$$\sum_{i \in \mathcal{C}} \lambda_i \cdot x_{i,w} \leq \overline{\omega}_w \quad w \in \mathcal{W}$$  \hspace{1cm} (3)

$$\sum_{i \in \mathcal{C}} \omega_i \cdot x_{i,w} \leq \overline{\omega}_w \quad w \in \mathcal{W}$$  \hspace{1cm} (4)

$$\sum_{i \in \mathcal{C}} \sum_{w \in \mathcal{W}} \omega_i \cdot x_{i,w} \leq \overline{\Omega}$$  \hspace{1cm} (5)

$$\sum_{j \in \mathcal{C}: \gamma_{j,i} = 1} x_{j,w} \leq T \cdot \left( y_{i,w} + \sum_{h \in \mathcal{W}: h \leq w} x_{i,h}\right) \quad i \in \mathcal{C} \quad w \in \mathcal{W}$$  \hspace{1cm} (6)

$$x_{i,w} \in \{0,1\} \quad i \in \mathcal{C} \quad w \in \mathcal{W}$$  \hspace{1cm} (7)

$$y_{i,w} \in \{0,1\} \quad i \in \mathcal{C} \quad w \in \mathcal{W}$$  \hspace{1cm} (8)

The cost function (1) includes rehandling costs and penalty costs for not loaded containers. Thanks to constraints (2), each container $i$ is assigned at most to one wagon. Constraints (3), (4) and (5) ensure that the train loading plan respects both length and weight capacity of the wagon and weight capacity of the train. Constraints (6) define rehandling movements, i.e. $y_{i,w}$ variables. Specifically, the meaning of constraints (6) can be explained as follows. If a container $j$, among those containers that are located under container $i$ in the stoking area, is assigned to wagon $w$, then two possibilities arise: either container $i$ has already been assigned to a wagon $h \leq w$ (and then $\sum_{h \in \mathcal{W}: h \leq w} x_{i,h} = 1$) or it is necessary to rehandle it (and so $y_{i,w} = 1$).

As already introduced, particular load and weight restrictions are related to physical wagon configurations. Thus we modified the previous formulation in order to both introduce the configuration decisions and take into account weight restrictions. In the following sections solution approaches are presented only for model M2, whose formulation therefore can be clearly discussed only starting from model M1. The additional notation is the following:

• $c(w)$ type of wagon $w$;
• $\mathcal{S}_w$ set of possible slots for wagon $w$;
• $\mathcal{K}_{c(w)}$ set of load configurations for wagon $w$ of type $c(w)$;
• $B_k$ set of weight configurations for load configuration $k$;
• $\delta_{b,s}$ maximum weight for slot $s$ in the weight configuration $b$;
• $\sigma_{i,k,s}$ compatibility of container $i$ with slot $s$ in load configuration $k$ ($\sigma_{i,k,s} = 1$ if the length of $i$ fits the length $s$ in configuration $k$, equal to 0 otherwise).

In order to clarify the notation, a simple example is reported in Fig. 3 for a wagon $w_1$ of type $c_1$. In this case there are 3 possible slots, then $\mathcal{S}_{w_1} = \{s_1, s_2, s_3\}$. Moreover, the considered wagon presents 2 load configurations, i.e. $\mathcal{K}_{c_1} = \{k_1, k_2\}$, and 7 weight configurations, i.e. $B_{k_1} = \{b_1, b_2, b_3\}$, $B_{k_2} = \{b_4, b_5, b_6, b_7\}$. The maximum weight is defined for each slot and for each weight configuration, as for instance $\delta_{b_1,s_1} = 5$, $\delta_{b_1,s_2} = 15$, $\delta_{b_2,s_1} = 10$, etc. The container fitting is expressed for instance by $\sigma_{i,k_1,s_1} = 1$ if container $i$ is 20'.

![Fig. 3. An example of a load pattern.](image-url)

The decision variables for the new formulation are the following:

• $x_{i,s,w} \in \{0,1\}$, equal to 1 if container $i$ is assigned to slot $s$ of wagon $w$;
• $t_{i,b} \in \{0,1\}$, equal to 1 if weight configuration $b$ is chosen for wagon $w$;
• $y_{i,w} \in \{0,1\}$, equal to 1 if container $i$ is rehandled (i.e. it is moved but not assigned) when wagon $w$ is loaded.

The second model for the TLP is again a 0-1 linear programming formulation.

**Model 2 (M2):**

$$\min \sum_{i \in \mathcal{C}} \sum_{w \in \mathcal{W}} \alpha \cdot y_{i,w} + \sum_{i \in \mathcal{C}} \pi_i \cdot \left(1 - \sum_{w \in \mathcal{W}} \sum_{s \in \mathcal{S}_w} x_{i,s,w}\right)$$  \hspace{1cm} (9)

s.t.

$$\sum_{w \in \mathcal{W}} x_{i,s,w} \leq 1 \quad i \in \mathcal{C}$$  \hspace{1cm} (10)
\[
\sum_{i \in \mathcal{C}} x_{i,s,w} \leq 1 \quad w \in \mathcal{W} \quad s \in \mathcal{S}_w
\]

\[
\sum_{b \in \mathcal{B}_k \cap \mathcal{K}(w)} t_{w,h} = 1 \quad w \in \mathcal{W}
\]

\[
x_{i,s,w} \leq \sum_{k \in \mathcal{K}(w)} \sigma_i,k,s \quad i \in \mathcal{C} \quad w \in \mathcal{W} \quad s \in \mathcal{S}_w
\]

\[
\sum_{i \in \mathcal{C}} \omega_i \cdot x_{i,s,w} \leq \sum_{b \in \mathcal{B}_k \cap \mathcal{K}(w)} \Delta_{b,s} \cdot t_{w,b} \quad w \in \mathcal{W} \quad s \in \mathcal{S}_w
\]

\[
\sum_{i \in \mathcal{C}} \sum_{s \in \mathcal{S}_w} \omega_i \cdot x_{i,s,w} \leq \Pi_w \quad w \in \mathcal{W}
\]

\[
\sum_{i \in \mathcal{C}} \sum_{w \in \mathcal{W}} \sum_{s \in \mathcal{S}_w} \omega_i \cdot x_{i,s,w} \leq \Pi
\]

\[
\sum_{j \in \mathcal{C}: \gamma_j, i = 1} \sum_{s \in \mathcal{S}_w} x_{j,s,w} \leq T \quad \left( y_{i,w} + \sum_{h \in \mathcal{W}, h \leq s \in \mathcal{S}_w} \sum_{i \in \mathcal{C}} x_{i,s,h} \right)
\]

\[
x_{i,s,w} \in \{0, 1\} \quad i \in \mathcal{C} \quad s \in \mathcal{S}_w \quad w \in \mathcal{W}
\]

\[
t_{w,h} \in \{0, 1\} \quad w \in \mathcal{W} \quad b \in \mathcal{B}_k
\]

\[
y_{i,w} \in \{0, 1\} \quad i \in \mathcal{C} \quad w \in \mathcal{W}
\]

Constraints (10) impose that each container is not assigned to more than one slot. Constraints (11) ensure that no more than one container is assigned to each slot. Constraints (12) impose that for each wagon a given weight configuration is chosen. Constraints (13) concern the length compatibility of containers to wagon slots. Constraints (14), (15) and (16) are relevant to the weight capacity for each slot, for each wagon and for the train, respectively. Constraints (17) impose the relation between \( x_{i,s,w} \) and \( y_{i,w} \) variables: if some containers under container \( i \) are assigned to a slot \( s \) of wagon \( w \), either container \( i \) has already been assigned to a wagon \( h \leq w \) (and then \( y_{i,w} = 0 \)) or it is rehandled (and then \( y_{i,w} = 1 \)).

IV. THE HEURISTIC APPROACH

The formulation M2, differently from model M1, takes into account the real physical constraints for the TLP, but its formulation could be hardly solved in short times for many real application cases. Therefore, we have devised a heuristic approach to solve model M2 aiming at finding the best containers to assign to each wagon, considering them sequentially and selecting containers according to their cost and their weight. The proposed algorithm is given by 4 different policies which differ for the criterium used in the choice of containers to load first on wagons (let us call them \( P1, P2, P3 \) and \( P4 \)), each one providing a solution to the train loading problem; then the best solution will be chosen and an improvement phase will be applied to it.

Let us start with policy \( P1 \). First of all, a priority term is associated to each container taking into account the two considered cost terms of model M2 (rehandling cost and not-loading cost). Let us denote with \( \sigma_i \) the priority for container \( i \). Supposing that at the beginning all containers are stored in blocks until tier \( T \), if container \( i \) is placed at tier \( \tau \), its priority \( \sigma_i \) can be computed as:

\[
\sigma_i = \pi_i - \alpha \cdot (T - \tau)
\]

Therefore, the container priority is determined by his commercial value, i.e. \( \pi_i \), minus the rehandling costs that would be paid if it were loaded on the train.

The assignment of containers to wagons will be done by considering containers priority and weight. To this purpose, 20' and 40' containers are gathered respectively in sets \( \mathcal{C}^{20} \) and \( \mathcal{C}^{40} \) in non-increasing order of \( \sigma_i \). Moreover, as regards the weight of containers, some clusters of containers are built in order to ease, successively in the algorithm, the research of load units which fit the constraints on weight capacity for the wagon slots. Specifically, we chose to employ, for the two sets \( \mathcal{C}^{20} \) and \( \mathcal{C}^{40} \), a number of clusters that is respectively \( \sqrt{|\mathcal{C}^{20}|} + 1 \) and \( \sqrt{|\mathcal{C}^{40}|} + 1 \) (this number is often used in literature for the gaussian or uniform distribution because it assures good performances in many different cases).

Moreover, in the algorithm a sequential loading (from the first wagon until the last one) of units is done selecting containers in \( \mathcal{C}^{20} \) and \( \mathcal{C}^{40} \), starting from the first elements (with higher values of \( \sigma_i \)); after each loading, all data structures are updated (i.e. \( \mathcal{C}^{20} \), \( \mathcal{C}^{20} \), \( \mathcal{C}^{40} \)). For each wagon with capacity of 3 TEUs, 5 load configurations are considered: let us denote with \( k_1 \) the 20-20-20 configuration, with \( k_2 \) the 40-20-20 one, with \( k_3 \) the 20-20-20 one, with \( k_4 \) the 40 one and, finally, with \( k_5 \) the 20 one. Since the main objective of the planning problem is to load each wagon as much as possible, these configurations will be evaluated sequentially as follows:

- \( k_1 \) and \( k_2 \) with the same priority (Step 1);
- \( k_3 \) and \( k_4 \) with the same priority (Step 2);
- \( k_5 \) (Step 3).

Therefore, if the assignment of containers to wagons can be done in Step 1, then Step 2 and Step 3 are not evaluated in the algorithm; if instead the assignment cannot be done in Step 1, Step 2 is analysed, and so on. Moreover, if a wagon with a capacity of 2 TEUs is considered, the algorithm starts evaluating Step 2 and, if necessary, Step 3.

For each load configuration, the feasibility in the assignment of containers is evaluated by two sequential checks:

- on wagon weight capacity \( \Pi_w \) (Cond1);
- on the capacity of the weight configurations (Cond2).

Note that in the choice of triples or couples of containers fitting the constraints, the algorithm searches for the suitable containers using the clusters previously created.

In Algorithm 1 the basic pseudocode for policy \( P1 \) is shown (considering the case of wagons with a capacity of 3 TEUs, then considering also \( P2 \)). As for the other policies, the structure of the algorithm remains the same but the priority values are computed in a different way and some randomization is introduced. Let us denote with \( \sigma_i^{20} \)
and $\sigma_i^{40}$ the priority of container $i$ if it is a 20' or a 40' container, respectively, to be used in policies $P_2$, $P_3$ and $P_4$. In general, these priorities can be written as $\sigma_i^{20} = \sigma_i^* (1 + \rho_i)$ and $\sigma_i^{40} = \sigma_i^* (1 + \xi_i)$ where $\sigma_i^*$ is given by (21). The values $\rho_i$ and $\xi_i$ in the three policies are computed as:

- $P_2$: a higher priority to 20' containers is assigned by randomly choosing $\rho_i$ in $[0, 1]$ and setting $\xi_i = 0$, $\forall i$; 
- $P_3$: a higher priority to 40' containers is assigned with $\rho_i = 0$ and $\xi_i$ random in $[0, 1]$, $\forall i$; 
- $P_4$: the priorities assigned are random and change dynamically while the containers are loaded on the wagons, i.e. $\rho_i = \mu \cdot \frac{(nlc^{20} - plc^{20})}{\sigma_i}$ and $\xi_i = \nu \cdot \frac{(nlc^{40} - plc^{40})}{\sigma_i}$ where $\mu$ and $\nu$ are suitably defined random parameters, $nlc$ indicates the number of containers not loaded on the train and $plc$ is the number of containers which could be loaded on the train, considering the available slots of the not processed wagons (note that $nlc$ and $plc$ are dynamical variables that change during the application of the algorithm).

As said before, the heuristic algorithm applies the 4 policies, chooses the best solution (among the 4 solutions found), and tries to improve it. More in detail, denoting with $P^*$ the policy that has provided the best solution, an improvement phase is applied to the solution obtained with $P^*$. The improvement phase has the aim of filling the wagon slots that are still empty by changing some slot-container assignments in order to load more containers. For the characteristics of the algorithm, if there is a empty slot after applying $P^*$, we are sure that there are no containers in the yard that could be placed there. Therefore, the improvement phase is done by trying to put in the empty slots containers that were assigned to other slots by $P^*$ and trying to load containers remained in the yard to fill the new empty slots (i.e. slots assigned by $P^*$ to containers that are now changing slots). Note that the improvement phase of the heuristics is applied only if in the solution of $P^*$ there are still containers in the yard and if there is at least one empty slot on the train.

In order to simplify the computational complexity of this improvement phase (and then to reduce computational times), this evaluation of possibly changing slot-container assignments is done neglecting the rehandling costs; this is motivated by the fact that we assume $\pi_i \gg \alpha_i$, $\forall i$. Under this assumption, increasing the total number of loaded containers certainly causes a decrease of the total cost.

V. COMPUTATIONAL RESULTS AND CONCLUSIONS

In this section some experimental tests are presented in order to analyse and compare the results obtained by applying the heuristic algorithm proposed in Section IV to solve the formulation $M2$. We implemented both the formulation $M2$ and the algorithm with MATLAB and we used Cplex 11.0 to solve the M2 formulation.

We ran the experimental tests on a set of randomly generated problem instances with data inspired from a real case study. We suppose that containers are stored along the railway tracks in blocks of 3 or 4 tiers each; in each row containers have the same length (20' or 40') and different commercial values (i.e. different $\pi_i$). We analysed 5 groups of instances whose main characteristics (number of containers and number of wagons) are reported in Table I. For each group we solved different instances, in which the train composition is random (wagons can be of 5 different types), the commercial values of containers are random (among 3 value levels), the percentage of 20' and 40' containers is random and the weight of containers varies randomly between 5 t and 30 t.
The computational tests were executed on a 2.4 GHz Intel Core Duo 2 computer with 4 GB of RAM. We solved model M2 giving a time limit of 10 minutes to the Cplex solver (this short time is motivated by the fact that in a real application this planning problem should be solved in a relatively short time). Table II shows for each group the average size and CPU values among 30 instances. In this table the average number of variables and constraints of the 0/1IP formulation is reported for each group, and the average computational times (in seconds) needed to solve model M2 and to apply the heuristics are compared. In the column referred to the computational time needed by Cplex, we report in brackets the number of times the solver was stopped by the time limit. The difference in times between the heuristics and the solver is very strong and, obviously, becomes larger for larger instances. Note that the computational time needed by Cplex is not reported for group E since no integer solution was found in 10 minutes; it is interesting to highlight that for this group the average time needed by the heuristic algorithm to find the solution still is very small, i.e. 23.95 seconds.

### TABLE II

<table>
<thead>
<tr>
<th>Group</th>
<th>Variables</th>
<th>Constraints</th>
<th>Cplex time</th>
<th>Heuristics time</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1670</td>
<td>1671</td>
<td>9.51 (0)</td>
<td>7.37</td>
</tr>
<tr>
<td>B</td>
<td>2504</td>
<td>2527</td>
<td>48.20 (2)</td>
<td>10.01</td>
</tr>
<tr>
<td>C</td>
<td>3807</td>
<td>3807</td>
<td>147.31 (4)</td>
<td>13.86</td>
</tr>
<tr>
<td>D</td>
<td>4551</td>
<td>4577</td>
<td>203.92 (3)</td>
<td>12.01</td>
</tr>
<tr>
<td>E</td>
<td>14210</td>
<td>14258</td>
<td>- (30)</td>
<td>23.95</td>
</tr>
</tbody>
</table>

In order to evaluate the quality of the solutions found by the heuristics with respect to the optimal solution provided by Cplex, the two following indicators are compared:

\[
RO = \sum_{i \in C} \sum_{w \in W} y_{i,w}
\]

\[
LR = 1 - \frac{\sum_{i \in C} \pi_i \cdot \left(1 - \sum_{w \in W} \sum_{s \in S_{in}} x_{i,s,w}\right)}{\sum_{i \in C} \pi_i}
\]

The indicator \(RO\) represents the number of rehandling operations, and it allows to evaluate the first term of the cost function; the second indicator \(LR\) represents a load ratio, obtained as the sum of containers loaded on the train (weighted with their commercial value \(\pi_i\)) divided by the sum of all the values \(\pi_i\); in other words, \(LR\) represents the ratio between the commercial value of loaded containers and the total commercial value of containers present in the stocking area before loading the train. This indicator is very useful to compare the heuristics with the solver as regards the second term of the cost.

In Table III the average values of \(RO\) and \(LR\) are reported for the solutions obtained by using Cplex and the heuristic algorithm. Except for group E in which the solver could not find an integer solution and then it is not possible to compare this solution with that obtained by the heuristic algorithm, in the other cases the performances of the algorithm are quite satisfactory, in particular as regards the \(LR\) indicator: the \(LR\) average values obtained with the heuristic approach are not too lower than those obtained with Cplex, indicating that the train loading is realised in a satisfactory way. The main weakness of the algorithm stands instead in the part relevant to the \(RO\) indicator, in which the algorithm performs not so well, determining a high number of rehandling operations for loading the train. Present and future research is then devoted to improve the heuristic approach specifically as regards this part of the cost function.

### TABLE III

<table>
<thead>
<tr>
<th>Group</th>
<th>RO Cplex</th>
<th>RO heuristics</th>
<th>LR Cplex</th>
<th>LR heuristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1.9</td>
<td>0.95</td>
<td>0.95</td>
<td>0.91</td>
</tr>
<tr>
<td>B</td>
<td>3.85</td>
<td>0.83</td>
<td>0.75</td>
<td>0.75</td>
</tr>
<tr>
<td>C</td>
<td>0.42</td>
<td>0.97</td>
<td>0.95</td>
<td>0.95</td>
</tr>
<tr>
<td>D</td>
<td>7.55</td>
<td>20.1</td>
<td>0.88</td>
<td>0.83</td>
</tr>
<tr>
<td>E</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>0.69</td>
</tr>
</tbody>
</table>

In conclusion, the experimental tests reported above show the importance of the heuristic approach, in particular for real applications. Real problem instances are characterized by trains composed of 25 wagons (sometimes also 30) and the containers placed in the stocking area to be loaded on the train could be approximately 80-100. If the maximum time to decide the assignment of containers to wagon slots is 10 minutes, as considered in the experimental tests reported above (and this seems a correct value for real cases), the importance of a heuristic approach seems absolutely relevant since it can find good solutions in very short times.