Noname manuscript No.
(will be inserted by the editor)

Optimization Approaches for Defining Storage Strategies in Maritime Container Terminals

Daniela Ambrosino · Haoqi Xie

Received: date / Accepted: date

Abstract In maritime container terminals, yards have a primary role in permitting the efficient management of import and export flows. In this work, a mixed 0/1 linear programming model and a heuristic approach are proposed for defining storage rules in order to minimize the space used in the export yard. The minimization of land space is pursued by defining the rules to allocate containers into the bay-locations of the yard, in such a way to minimise the number of bay-locations used and the empty slots within them.

The main aim of this work is to propose a solution approach for permitting to the yard manager to compare yard storage strategies for different transport demands, in such a way to be able to evaluate and, eventually, change the storage strategy when the characteristics of the transport demand change. Computational experiments, based on both real instances and generated ones, are presented. All instances are derived by a case study related to an Italian terminal.

Keywords export containers · storage policies · optimization model · heuristic approach · yard space minimization

1 Introduction and literature review

Maritime container terminals are generally recognised as crucial intermodal change nodes in the logistic chains, managing the greater part of the world sea trade, i.e., about 80% of the world one (UNCTAD, 2018).

Daniela Ambrosino
Dep. of Economic and Business Studies, Università di Genova, Via Vivaldi,5, 16126 Genova, Italy
Tel.: +39-010-2095494
E-mail: ambrosin@economia.unige.it

Haoqi Xie
Dep. of Economic and Business Studies, Università di Genova, Via Vivaldi,5, 16126 Genova, Italy

The storage yards have a primary role in permitting the efficient management of import and export flows [2] and, in the recent years, thanks to advancements in quay side equipment and technologies, seems the bottleneck of port operations has moved from quay side to yard side [16]. This means that, the typical operations performed in the yard, such as the storage and the retrieval of containers, dispatching and routing of material handling equipment, must be managed improving their efficiency, in such a way not to compromise the efficiency of the whole terminal system and the competitiveness of the terminal in the whole logistic chain.

The yard, the intermediate area between the frontier and the backward of a terminal used to store, control and handle the containers, occupies a considerable part in a terminal. The yard is usually divided into some segmentation for the inbound and outbound containers based on the process of import and export, respectively. Referring to the location of the input/output container points, two different configurations of layout are possible: Asian and European layout [18].

The container yard is basically divided into numerous *Blocks*, each one composed by a given number of *Bays*.

Each bay is formed by several *Rows*; containers are thus stacked in *Tiers* [10]. In modern container terminals, the maximum tier to stock a container in a block is 4 and the utilization ratio ranges from 70% to 90% [15]. The identification of a container position in the yard is based on these 3 indicators: *Bay*, *Row* and *Tier*.

As far as the storage strategies are considered, the most part of literature is devoted to export containers. Many terminals store containers in the yard in accordance with their loading vessels. In this case, the terminal has to assign sub-blocks to vessels and than organize the storage of containers inside every sub-block. This problem is known as the yard template planning ([11], [22]) and it represents a tactical level decision problem. The yard template planning has been generally solved under deterministic assumptions (i.e. the number of containers to load on a vessel is known).

At operational level, given a yard template, the terminal solves the storage allocation problem ([21], [9]) generally based on the sub-blocks. Some authors refer to these sub-blocks as loading clusters. In [20] the authors model the choice of loading clusters in such a way to obtain a more flexible allocation strategy for organizing the space in the export yard. They describe in detail the concept of loading cluster and loading operations, the link between these two important activities. In [17] and [4] the authors try to develop more flexible yard management by determining simultaneously the size of the loading clusters and their allocation to specific blocks. A loading cluster is a stretch of bays in a specific yard block. The terms loading is used to stress the importance of coordinating the bay configuration in the yard with the slots on the ship in a given bay. The ideal situation is to have the container in the same yard stack to be put in the same ship bay. Thus, the yard manager has to optimize the choice of loading cluster when considering their loading operations. In [3] the authors optimize the yard template and the yard storage allocation problems simultaneously.

More recent papers deal with the robust yard template facing uncertainty [22]. In [12] the authors evaluate how the block widths affects the terminal performances thanks to a discrete event simulation model, while [13] show how the length of the blocks in the storage yard affects.

The template of the terminal is organized in accordance with the handling equipment used. In the analysed literature, the terminals use Rubber Tyred Gantry (RTG) cranes. The template of a terminal using reach stackers for picking up export containers is quite different, since the blocks are operated from one side. The pick up operations and the number of re-handles to execute in order to pick up a container is affected by the type of terminal equipment used.

In this paper, we consider an European layout where import and export yards are independent, and we deal with export standard containers. Handling operations in the export yard are executed thanks to reach stackers.

The yard template is given, this means that the export yard is organized in blocks of different capacities and, for each vessel there is a subset of dedicated blocks, that is the containers that will be loaded on that vessel must be stored in the dedicated subset of blocks. Containers can be stored in the dedicated blocks in accordance with different storage strategies. Since now we only consider the subset of blocks dedicated to a vessel and the containers that must be loaded on it.

Each container is characterized by its type, size, weight and destination; these characteristics are important when defining the storage strategy. The ideal rule is to store together containers having the same characteristics, in order to reduce the operation time and avoid bottleneck in the terminal when loading the ship [21], [14]. This strategy is known as *consignment* strategy. Note that, this strategy requires large storage space (for example more than random policies [7]), but on the other hand permits to improve the storage yard operations during the vessel containers loading in terms of productivity of both pick up operations in the bays and movements of material handling equipment among bays. Note that when a random policy is used, generally in European layout terminals, another strategy follows to improve the efficiency in the loading process; this strategy can be either a pre-marshaling strategy that permits to reorganize the container stacking beforehand, in order to reduce reshuffles, or a re-marshaling strategy that permits to move containers from their current storage location to a location closer to their vessel.

Among the papers dealing with the storage allocation for export containers, in [6] a consignment strategy, based on Light, Medium and Heavy weight classes, destination and size,is used to decide an exact slot for an individual container; [5] tries to increase the loading operation efficiency by considering the travel distances of equipment, while in [8] the optimal storage location is determined taking into consideration the container handling schedules.

In [19] four rules to determine the number of blocks to allocate the groups of export containers are proposed. Rules are fixed and the main aim is to optimize the movements of yard equipment and the distance between the yard and the quay. Moreover, the authors evaluate the influence of the yard size on the efficiency of loading operations.

In [1] an optimization model for defining storage strategies for export containers in proposed. The authors focus on the definition of rule of consignment strategy with the aim of minimizing the space used.

Starting from the model proposed in [1], in the present paper a new formulation for defining the best allocation of containers to storage spaces is proposed, simultaneously defining the best consignment strategy to use.

The main purpose of this work is to propose a solution approach, based on a mathematical model, being able to determine the set of best rules for defining which

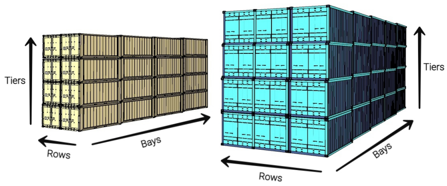


Fig. 1 Diagram of bay-locations: yellow block with 8 capacity bay-locations; light blue block with 12 capacity bay-locations

containers to store together while determining the loading cluster for each group of containers. From a managerial point of view, the proposed approach permits to the yard manager to compare storage strategies and in particular, it should be used by the yard manager to evaluate and, eventually, change the current storage strategy when the characteristics of the transport demand change.

The reminder of this paper is organized as follows. The problem under investigation is described in Section 2. The 0-1 linear model is presented in Section 3, while the solution approach is described in Section 4. Finally, the experimental tests are reported in Section 5 and conclusions and future works are outlined in Section 6.

2 Problem under investigation

2.1 The general contest

Let us consider an export yard and the blocks dedicated to a particular vessel. Blocks are characterized by different capacities, depending on bays, rows and tiers. Generally, the number of rows ranges from 2 to 5, while 4 tiers are considered.

A *bay-locations* is a set of cells belonging to the same bay of a block. Thus, the capacity of a bay-locations varies according to the number of rows in the block; the capacities are 8, 12, 16 and 20 containers. Figure 1 represents two different blocks with several bays. The yellow part is a block composed by 4 bays, each one characterized by 2 rows, thus the capacity of each bay-locations is 8. Meanwhile, the blue part is a block composed by bay-locations with 3 rows, thus, the capacity of each bay-locations is 12.

Note that we refer to 20' bay-locations. For the storage of 40' containers, two contiguous 20' bay-locations are required.

Figure 2 gives a sample of possible usages of a block characterized by 6 bay-locations, having both 20' and 40' containers to store. The possible usages are: i) the whole block is used for the storage of 20' containers (e.g., 6 bay-locations are occupied by 20' containers); ii) the whole block is used for the storage of 40' containers (e.g., 6 bay-locations are used for composing 3 40'-bay-locations and are occupied by 40' containers); iii) the block is used for stowing both 20' and 40' containers, thus either 4 bay-locations of 20' containers and 1 40'-bay-locations for 40' containers, or 2 bay locations of 20' containers and 2 40'-bay-locations for 40' containers.

Summarizing, the yard consists in a given number of 20' bay-locations (here called simply bay-locations) of different capacities.

Let B be the set of bay-locations, and Q the set of bay-locations capacities, B_q is the subset of bay-locations having capacity q, and $B = \cup_{q \in Q} B_q$.

The yard manager assigns containers to the bay-locations following the storage rules adopted by the terminal.

The storage rules consist in a list of characteristics that containers may have to be stored together. These rules permit to have homogeneous containers in each bay-locations, i.e., to be able to pick them up in sequence for their loading on board of the vessel, and to optimize the work of the reach stackers during their pick-up in the yard (it is preferred to complete the pick-up process in a bay-locations and empty it before moving the reach stacker to another bay-locations).

The most common characteristics used when defining a storage strategy are the following:

Size: 20 feet (20') and 40 feet (40') containers; only stacks (and bay-locations) of one size (i.e., either 20' or 40') are permitted.

Type: standard containers, 20' and 40' box and 40' HC containers. (special containers follow different rules derived directly from the particular requirements for their storage: plugs for reefers, special locations for hazardous and out of huge etc)

Destination: containers are grouped in accordance with their destination. In fact, containers, on board, are generally grouped for homogeneous port of discharge, i.e., either a bay of the vessel, or a part of it, is dedicated to store containers for the same destination.

Weight: containers stored in the same bay have similar weights for respecting the requirement of safety, generally saying that the weight of the container stored in a given tier has to be no greater than the weight of the container stored in tier below it, within a given tolerance. Many terminals group containers according to the weight classes, i.e., containers belonging to the same weight class can be stored together; the most com-

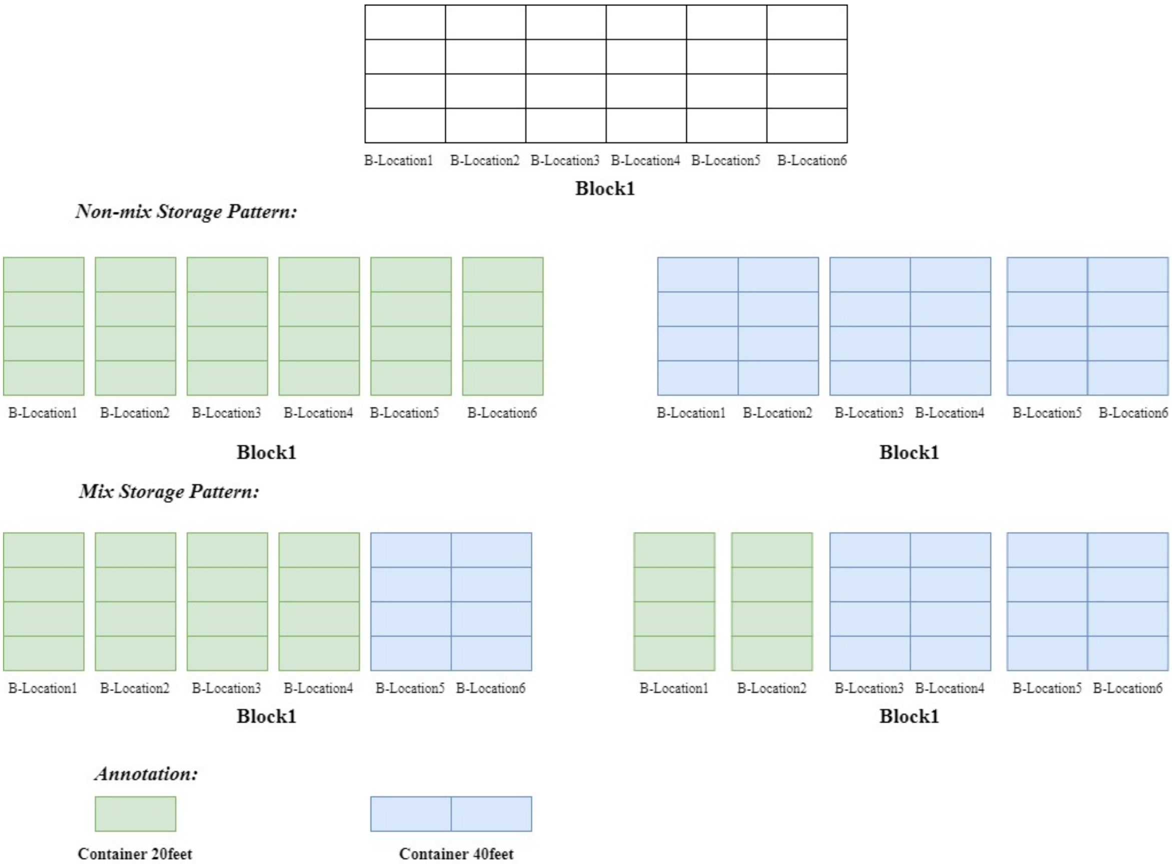


Fig. 2 Storage patterns for containers in a single block

mon configuration used is based on three classes: Light (from 5 tons to 15 tons); Medium (from 15 tons to 25 tons); Heavy (over 25 tons).

It is easy to understand that the elements more impacting on the space utilization are the following:

- the yard template and layout of blocks: the capacities of the bay-locations and the number of bay-locations of each capacity;
- the consignment strategy: the characteristics used and the rules adopted impact on the number of containers of each group, and the number of groups to manage (as in Figure 3). The number of groups to manage corresponds to the required *patterns* of bay-locations.

In Figure 3 containers are grouped in accordance with their destination, their type (Box or High Cube), their size and their weight class. For each destination nine patterns must be managed. The higher is the number of patterns to manage the higher is the yard space required. The yard space required is also function of the bay-locations capacities. How it is possible to act on these elements to reduce the yard space without penalising the efficiency of loading operations?

The number of classes used for defining the storage rules has a direct impact on the number of patterns, while acting on the weight limitation of each class it is possible to modify the number of containers in each patter, that is the number of the required bay-locations for each pattern. Hence, these two elements might have a great impact on the space used in the yard.

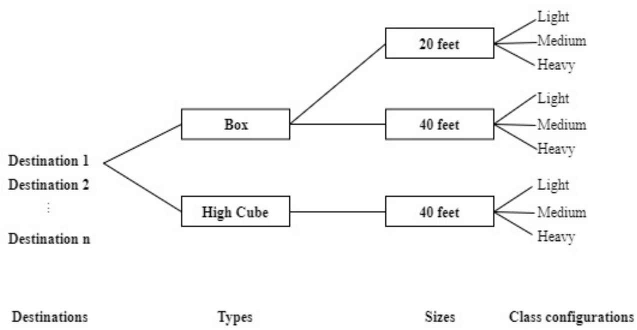


Fig. 3 Decision tree for defining patterns of bay-locations.

We will refer to the number of class of weight as *Class configuration.* Each class configuration can be characterized by many weight limitations, here called *Weight configurations*.

Let us give an example. Let us consider the more common class configuration with three classes: Light, Medium and Heavy. For this configuration two alternative weight configurations can be considered 3-15/15-25/25-33 and 3-12/12-22/22-33.

2.2 The analysed contest

As explained above, when implementing the storage strategy the key elements are the class configuration (i.e. the number of classes used to split containers) and the weight limitations of each class, together with the characteristics of containers such as destination, type and size.

In very general terms, the problem under investigation can be described as follows. Given the export yard blocks characterized by a set of bay-locations of different capacities dedicated to store the containers waiting for their loading on a specific vessel, given a set of containers representative of the average transport demand for the considered vessel, the problem consists in deciding the *Class configuration* and the *Weight configuration* to use for grouping the containers in order to minimize the space used in the export yard.

Let us now introduce the problem in more details.

As far as the yard is considered, the following elements are given: the set of bay-locations of the blocks and the number of bay-locations having a given capacity q; the set of the possible class configurations and the set of weight configurations of each class, together with the weight limits of each weight configuration. An example of possible class and weight configurations is reported in Table 1. Let us consider three class configurations, i.e. F_2, F_3 and F_4 each having two weight configurations associated with. If class configuration F_2 with only two weight classes is used for the storage strategy, then either weight configuration W_1 or W_2 can be chosen.

Moreover, each container to store in the yard is characterized by its size, type, weight, and destination.

The problem consists in deciding the assignment of each container to a specific bay-locations, while simultaneously determining the class configuration of the blocks dedicated to the vessel under investigation, and the characteristics of each bay-locations in terms of destination, type, size, capacity (among the set of available capacities in the blocks of the yard) and the weight limitations among the set of weight limitations for the chosen class configuration, in order to respect the yard capacity, minimize both the number of bay-locations and the total empty slots.

Note that this problem emerges at tactical level for defining rules to use in the operative contests. These rules are not fixed once and ever adopt; the idea is to modify them following the trend of the export flow demand. Consider, for example, a service served by the terminal, and suppose that the number of containers for a destination of this service increases; we are interested to observe which groups of containers increase (i.e., 20' box containers or heavy 20' ones, etc.). Only in this way, we are able to decide if the existing rules are adequate or not, and in this latter case, how to modify them thanks to optimization approach. The model and the solution approach useful for doing that analysis are presented in the following sections.

3 The mathematical model

In this section, a basic 0-1 linear programming model to solve the problem described in Section 2.2 is presented. The useful notation is the following:

B the set of bay-locations
Q the set of bay-locations capacities (i.e. 8,12,16,20)
F the set of the possible class configurations
W the set of weight configurations
P the set of weight limits
C the set of containers
D the set of ports of destination
H the set of types/heights (Box, High Cube)
S the set of sizes (20, 40 feet)
n_q the number of bay-locations having capacity $q, \forall q \in Q$
d_i destination of container $i, \forall i \in C$
s_i size of container $i, \forall i \in C$
h_i height of container $i, \forall i \in C$
w_i weight of container $i, \forall i \in C$
u_p weight upper bound of weight limit $p, \forall p \in P$
l_p weight lower bound of weight limit $p, \forall p \in P$
$\delta_{fw} \forall f \in F, \forall w \in W$, $\delta_{fw} = 1$ if the weight configuration w belong to class configuration f, 0 otherwise
$\gamma_{wp} \forall w \in W, \forall p \in P$, $\gamma_{wp} = 1$ if the weight limits p belong to configuration w, 0 otherwise
α weight used in the objective function for penalising the empty slots in the bay-locations.

Let us introduce the following decision variables:
$x_{ij} \in \{0,1\}, \forall i \in C, \forall j \in B$, $x_{ij} = 1$ if container i is stored in bay-locations j
$y_{j,d,h,s,q} \in \{0,1\}, \forall j \in B, \forall d \in D, \forall h \in H, \forall s \in S, \forall q \in Q$, $y_{j,d,h,s,q} = 1$ if in bay-locations j, with capacity q, are stored containers for destination d, with height h and size s
$c_{pj} \in \{0,1\}, \forall p \in P, \forall j \in B$, $c_{pj} = 1$ if weight limits p are assigned to bay-locations j

Table 1 An example of possible yard class and weight configurations.

Weight	Class Config. F_2	Class Config. F_3	Class Config. F_4
W_1	$p_1(0-15), p_2(15-33)$	-	-
W_2	$p_3(0-20), p_4(20-33)$	-	-
W_3	-	$p_5(0-12), p_6(12-22), p_7(22-33)$	-
W_3	-	$p_8(0-15), p_9(15-25), p_{10}(25-33)$	-
W_5	-	-	$p_{11}(0-12), p_{12}(12-20), p_{13}(20-28), p_{14}(28-33)$
W_6	-	-	$p_{15}(0-10), p_{16}(10-18), p_{17}(18-25), p_{18}(25-33)$

$a_f \in \{0,1\}, \forall f \in F$, $a_f = 1$ if class configuration f is chosen for the yard storage
$b_w \in \{0,1\}, \forall w \in W$, $b_w = 1$ if weight configuration w is chosen
$z_j \geq 0, \forall j \in B$, number of empty slots in bay-locations j.

The resulting model is the following:

$$Min \sum_{j\in B}\sum_{d\in D}\sum_{h\in H}\sum_{s\in S}\sum_{q\in Q} y_{j,d,h,s,q} + \alpha \sum_{j\in B} z_j \tag{1}$$

Subject to:

$$\sum_{j\in B} x_{ij} = 1 \quad \forall i \in C \tag{2}$$

$$\sum_{d\in D}\sum_{h\in H}\sum_{s\in S}\sum_{q\in Q} y_{j,d,h,s,q} \leq 1 \quad \forall j \in B \tag{3}$$

$$\sum_{i\in C} x_{ij} \leq \sum_{d\in D}\sum_{h\in H}\sum_{s\in S}\sum_{q\in Q} q y_{j,d,h,s,q} \quad \forall j \in B \tag{4}$$

$$\begin{aligned}&\sum_{j\in B}\sum_{d\in D}\sum_{h\in H}\sum_{s\in S:s_i=20} y_{j,d,h,s,q}\\ &+2\sum_{j\in B}\sum_{d\in D}\sum_{h\in H}\sum_{s\in S:s_i=40} y_{j,d,h,s,q} \leq n_q \quad \forall q \in Q\end{aligned} \tag{5}$$

$$\sum_{f\in F} a_f = 1 \tag{6}$$

$$\sum_{w\in W} b_w = 1; \tag{7}$$

$$\sum_{w\in W} \delta_{wf} b_w \leq a_f \quad \forall f \in F \tag{8}$$

$$\sum_{p\in P} c_{pj} \leq 1 \quad \forall j \in B \tag{9}$$

$$\sum_{p\in P} \gamma_{pw} c_{pj} \leq b_w \quad \forall j \in B \, \forall w \in W \tag{10}$$

$$w_i x_{ij} \leq \sum_{p\in P} u_p c_{pj} \quad \forall i \in C, \forall j \in B \tag{11}$$

$$w_i x_{ij} \geq \sum_{o\in P} l_p c_{pj} - 1000(1-x_{ij}) \quad \forall i \in C, \forall j \in B \tag{12}$$

$$\sum_{d\in D}\sum_{h\in H}\sum_{s\in S}\sum_{q\in Q} q y_{j,d,h,s,q} - \sum_{i\in C} x_{ij} \leq z_j \quad \forall j \in B \tag{13}$$

Equation (1) is the objective function that minimizes the number of bay-locations used and penalises the empty slots in the bay-locations.

Thanks to constraints (2) each container must be stored in one bay-locations. Constraints (3) assign at most one destination, one size, one type and one capacity to each bay-locations. Constraints (4) verify the number of containers assigned to the bay-locations is less or equal to the capacity assigned to it.

The yard capacity, in terms of number of bay-locations of the different capacities available (i.e. 8, 12, 16, 20 containers), is verified thanks to (5).

Constraints (6), (7) and (8) refer to the choice of a class configuration together with a weight configuration.

Only one couple of weight limits can be assigned to each bay locations (9), and thanks to (10) the weight limits are assigned to each bay-locations in accordance with the choice of the weight configuration chosen for the blocks of the yard. Thanks to (11) and (12) a container can be assigned to a bay-locations only if its weight is within the maximum and minimum weight limitations imposed to the bay-locations by the weight configuration assigned to it thanks to (9). In (13) the number of empty slots in each bay-locations is computed.

Since the proposed model (1)-(13) can be solved up to optimally only for small instances (as shown in Section 5) we propose a heuristic procedure as describe in next section.

4 Heuristic approach

In order to solve the real size problems for defining the best storage strategy rules, we propose a heuristic approach based on the model (1)-(13). From computational results (see Section 5) it is clear that the number of destinations and the different capacities of bay-locations have a great impact on the CPU time. Due to these considerations and to the fact that, each consignment strategy can not be constructed without grouping containers with respect to their destination and size, we propose a solution approach that decomposes the problem in sub-problems. In particular, we solve model (1)-(13) for each size of containers (i.e. 20' and 40') and for each destination. Moreover, we relax the capacity constraints due to the layout of the yard; thus, we suppose to have an unlimited number of bay-locations of 16 and 20 containers capacity. The union of the obtained solutions can produce a global unfeasible solution with respect to constraints (5) that verify the yard capacity, in terms of number of bay-locations of the different capacities available. Moreover, the obtained solution should present different class configurations for different destinations (i.e. a violation of constraint (6)) and, as a consequence, different weight limitations. In the proposed heuristic only the unfeasibility with respect to the yard capacity is eliminated.

The main steps of the proposed heuristic procedure are the following:

Step1: construct a solution and verify its feasibility;

Step2: remove unfeasibility by new assignment of containers belonging to the bay-locations used in more quantities than available with respect to the real yard layout.

Before describing the solution approach, let us recall and add the required notation:
n_q number of available (20')bay-locations of capacity q
u_q number of (20')bay-locations of capacity q used in the current solution
E_q number of bay-locations of capacity q used in excess with respect to the available ones (n_q)
A_q number of bay-locations of capacity q left with respect to the available ones (n_q)
β used to indicate the size of containers (20' and 40')
L_q list of bay-locations of capacity q used in the current solution
z_j number of empty slots in bay-locations j
m_j number of containers stored in the bay-locations j

Step1: construct a solution and verify its feasibility

After having solved model (1)-(13) for each destination d and for each size s, we construct, by the union of the obtained solutions, the complete current solution $\hat{x}$, characterized by a given number of used bay-locations with capacity 16 and 20 containers.

For verifying, in Step 1, the feasibility of the complete solution $\hat{x}$, it is necessary to compute the number of 20'-bay-locations used for each capacity q (i.e. 16 and 20) and compare them with n_q.

This check is detailed in the following procedure describe in c-like.

Check feasibility:

For each capacity q, set $u_q = 0$
  Set $\beta \in \{1, 2\}$
    If the bay-location j is used for 20' containers
    Set $\beta = 1$, then compute u_q=u_q + β;
    End If
    If the bay-location j is used for 40' containers
    Set $\beta = 2$, then compute u_q=u_q + β;
    End If
End For
For every capacity q, calculate E_q and A_q:
    If $n_q - u_q$ >0;
    Set $A_q = n_q - u_q$:
    End If
    If $s_k - u_k$<0
      Set $E_q = |n_q - u_q|$
      Create the list L_q
      put L_q in descendent order with respect to the empty slots (z_j)
    End If
  If E_q>0 for at least one q
    the current solution is unfeasible.
    Go-to Step 2
  otherwise : Stop- the current solution is feasible
End For

Note that when E_q is set equal to $|n_q - u_q|$ this means that the current solution $\hat{x}$ has exceeded the total capacity of the yard in terms of bay-locations of capacity q. Thus, all bay-locations with capacity q are put in the list L_q in descendent order with respect to their empty slots (z_j) in such a way to try to reduce the number of bay-locations with capacity q used, starting from those having the largest number of empty slots, as better explained in the following Step2.

Step2: obtaining feasibility

When the current solution $\hat{x}$, characterized by a given number of used bay-locations with capacity 16 and 20 containers does not respect the yard layout, i.e., there is at least one $E_q > 0$, then it is necessary to modify the usage of bay-locations in the yard.

Since one of the aim is to minimize the number of empty slots in the bay-locations the idea is to replace the bay-locations with large numbers of empty slots with bay-locations of different and more adequate capacities (thus able to reduce the number of empty slots). For example suppose to have a bay-locations of capacity 20 containers with 9 empty slots. If in the yard is available a bay-locations with 12 containers capacity we can change them and reduce the number of empty slots from 9 to 1. In case there is no a bay-locations with capacity greater than 11 containers, and we need to remove the 11 containers assigned to the bay-locations with capacity 20 containers, we can try to split the 11 containers in two bay-locations, the most adequate among the available in the yard; for example we can use two bay-locations with capacity 8 containers (if available in the yard).

These ideas, used to modify the current unfeasible solution in order to obtaining feasibility, are detailed in the following procedure describe in c-like.

Obtaining feasibility by removing bay-locations over used:

For each capacity q, such that $E_q > 0$
 While $E_q > 0$
 Select the first element of the list L_q
 Let be $\hat{j}$ the bay-locations selected
 Search in the yard the most adequate bay-locations
  Let be:
  j^{new} the selected bay-locations
  q^{new} its capacity
 If $q^{new} >= m_{\hat{j}}$
  Update the solution
  Set $L_q = L_q - \hat{j}$
  $E_q = E_q - 1$, $A_q^{new} = A_q^{new} - 1$
 otherwise:
  Search the 2 most adequate bay-locations
  for receiving - by splitting - $m_{\hat{j}}$
   Let be:
   j^{new1}, j^{new2} the selected bay-locations
   q^{new1}, q^{new2} their capacities
   Compute:
   $m_{j^{new1}} = q^{new1}$ and $z_{j^{new1}} = 0$
   $m_{j^{new2}} = m_{\hat{j}} - q^{new1}$ and $z_{j^{new2}} = q^{new2} - m_{j^{new2}}$
   If $q^{new2} >= m_{j^{new2}}$
    Update the solution: set
    $L_q = L_q - \hat{j}$
    $E_q = E_q - 1$
    $A_q^{new1} = A_q^{new1} - 1$, $A_q^{new2} = A_q^{new2} - 1$
   otherwise:
    Stop- there is not a feasible solution
End For

The search in the yard among the available bay-locations is realised by comparing the capacity of available bay-locations with $m_{\hat{j}}$ that is the number of containers stored in bay-locations $\hat{j}$, in order to minimizing the empty slots in the new selected bay-locations. If there is not a bay-location with capacity greater than $m_{\hat{j}}$, the most adequate bay-locations is anyway selected among the bay-locations with the largest capacity. In this case, the idea is to split $m_{\hat{j}}$ in two bay-locations, thus it is necessary to select the two most adequate bay-locations, again with the aim of minimizing the number of empty slots. If it is not possible to split $m_{\hat{j}}$, this means that we are not able to construct a feasible solution starting from the $\hat{x}$, having the current layout. Note that, it is not possible to modify the layout on demand, that is, in any case a difficulty in storing containers emerges. But in the terminal under investigation can often happen that a part of a block originally dedicated to another vessel, is used for solving a critical - temporarily situation.

5 Computational experiments

To verify the effectiveness of the proposed model and to justify the choice of the heuristic approach, we conduct a series of numerical experiments. In the first campaign we try to show the behaviour of the model in different circumstances, i.e. when permitting more freedom degree to choose the storage strategy, when considering different yard layout and increasing the instance size in terms of number of containers. Moreover, we try to show the benefits for the terminal yard manager of having more freedom degree in choosing storage strategies, evaluating the impact on the space utilization. In the second campaign, we test the model by using a particular scenario (i.e. Scenario3) by increasing both the number of containers and the number of destination.

The model introduced in Section 3 and the solution approach described in Section 4 have been implemented in MPL (Mathematical Programming Language) and spreadsheet Excel, and solved by the commercial solver GUROBI on a device with Intel Core i7, 2.6 G Hz, Memory 16G. All experiments have been conducted by using instances generated by having in mind the real cases solved by a container terminal of an Italian port. Also some instances derived by a case study are reported to validate the proposed approach.

5.1 First experimental campaign

In the first campaign we use small-scale instances. In particular, we refer to instances SS, characterized by

Table 2 Layout characteristics for SS and MS instances

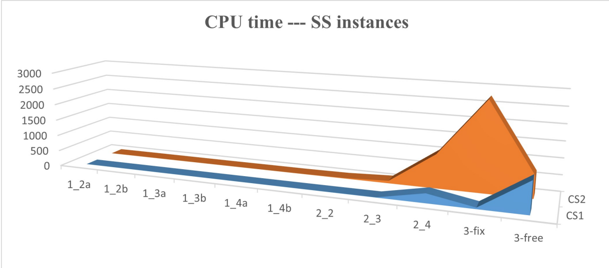
CS1	cap 12	cap 16	cap 20
SS	0	10	10
MS	0	30	30
CS2	**cap 12**	**cap 16**	**cap 20**
SS	8	8	8
MS	15	15	15

86 containers to load on the same vessel, and instances MS with 320 containers. The main aim of this first campaign is to investigate the behaviour of the model presented in Section 3 either when fixed and predetermined weight configuration and class configuration are used or different weight and class configurations can be chosen. For doing this analysis we investigate four scenarios of increasing difficulty and complexity. Details of parameters of each scenario are specified in Figure 9 reported in the Appendix. In particular, in Scenario1 only one fixed class configuration is used, with fixed weight limits. Scenario2 permits to chose the weight limits for a given class configuration. Scenario3 permits to choose among different class configurations, each one characterized by fixed weight limits. The last scenario, Scenario4, represents the larger degree of flexibility: it is possible to choose both the best class configuration and weight configuration. Note that model proposed in Section 3 permits to face this problem, the more general case for defining the best storage strategy.

Moreover, in this analysis we suppose to have two different terminal layouts in terms of capacity of the bay-locations. In fact, thanks to an historical data analysis of the terminal under investigation, we know that bay-locations with capacity of 12, 16, and 20 are in common usage. Thus, we compare performances under two different situations, a capacity set 1 (named CS1) in which the blocks of the terminal are composed by bay-locations with a capacity of 16 and 20 containers, and a capacity set 2 (named CS2), in which bay-locations have capacity 12, 16, and 20 containers. In particular, the terminal layout characteristics in terms of quantity of bay-locations of each capacity are summarized in Table 2.

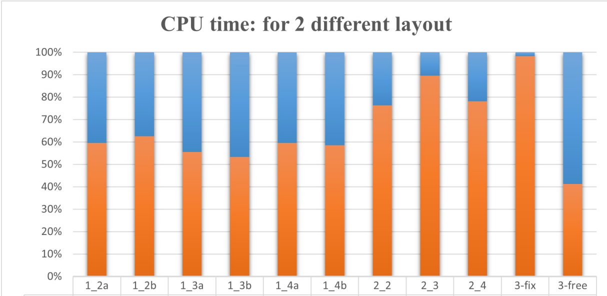
In this step, for the first scenario, the maximum CPU Time is set as 3600 seconds, for the second scenario, the maximum CPU Time is 10800 seconds, while for the last two scenarios, the time limit is fixed to 14400 seconds.

The detailed results of the different scenarios solved are reported in Appendix in Figure 10 and in the following Figures where are shown by some graphs, in such a way to read them easily. In particular we can note from



	1_2a	1_2b	1_3a	1_3b	1_4a	1_4b	2_2	2_3	2_4	3-fix	3-free
CS1	3.26	3.37	3.92	3.04	3.48	3.7	5.23	8.93	289	46.68	968
CS2	4.79	5.64	4.9	3.48	5.12	5.23	16.82	77	1034	2742	679

Fig. 4 CPU time trend for small instances SS



	1_2a	1_2b	1_3a	1_3b	1_4a	1_4b	2_2	2_3	2_4	3-fix	3-free
SS CS1	3.26	3.37	3.92	3.04	3.48	3.7	5.23	8.93	289	46.68	968
SS CS2	4.79	5.64	4.9	3.48	5.12	5.23	16.82	77	1034	2742	679

Fig. 5 Small instances SS - CPU Time comparison for layout CS1 and CS2

graph in Figure 4 that all SS instances can be solved up to optimally in the four analysed scenarios. More flexibility in the class and weight configuration choice required more CPU time, and from the graph it is easy to note that instances characterized by layout CS2 are more time consuming. This conclusion is more clear in the second graph shown in Figure 5.

In case of medium size instances MS, model (1)-(13) can be used to solve up to optimally only instances characterized by the simple layout (SC1) with both the class configuration and the weight configurations fixed. The trend of the CPU time and Gap are reported in the graph in Figure 6.

Finally, we have investigated how the space is used in the yard when the different layouts are implemented. From the results reported in Figure 10 we can obtain the graphs depicted in Figures 7 and 8. We can note that the number of empty slots are lower when in the yard are available bay-locations with 12, 16 and 20 container capacity.

By fixing a-priori either the number of classes to use or the weight limitations for each class can cause a inefficient usage of the space in the yard. In fact, if we consider SS instances we can note that, without optimizing the storage strategy can be generated even

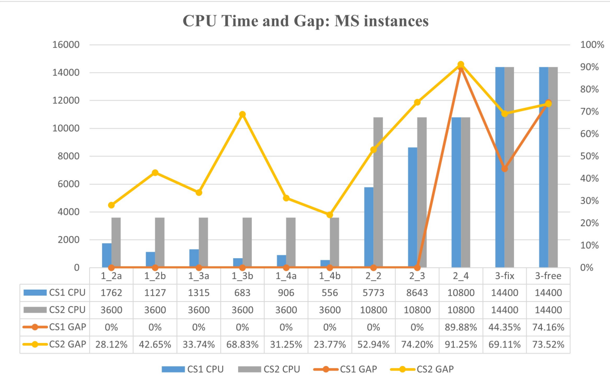


Fig. 6 Trend of CPU Time and Gap for MS instances

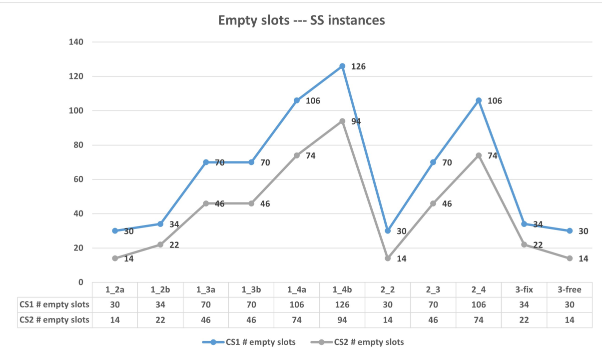


Fig. 7 Empty slots in the different scenario: small instances SS

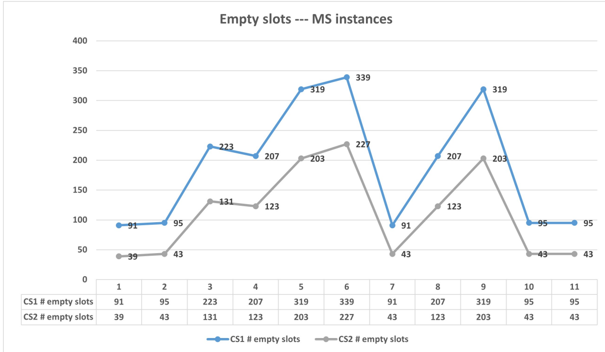


Fig. 8 Empty slots in the different scenario: medium instances MS

Table 3 Comparison of two groups in the second step

CNTR	dest.	CPU	bay-loc.	empty s.	Conf.	Gap
200	1	85s	14	80	F1	0%
	4	48.43s	24	280	F1	0%
	8	99s	35	500	F1	0%
400	1	265s	23	60	F1	0%
	4	272s	32	240	F1	0%
	8	399s	42	440	F1	0%
800	1	7200s	42	40	F1	40.24%
	4	7200s	52	240	F1	44.52%
	8	7200s	66	520	F1	41.95%
1600	1	14400s	82	40	F1	34.42%
	4	14400s	96	320	F2	79.06%
	8	14400s	116	720	F2	85.08%

94 empty slots [126] while in the optimal solution only 14 empty slots [30] are present when layout CS2 [CS1] is used. These numbers grow for instances MS. We can also note that the number of empty slots increases more than 100 percent by passing from layout CS2 to CS1, in case of optimal solutions.

Obviously, small capacity is more attractive in order to reduce vacancy in every bay-location. However, the amounts of bay-locations generated remain almost same under two different capacity sets. The influence denoted by the variety of weight configuration can be deemed as small on the space utilization and bay-locations utilization.

5.2 Second experimental campaign

This campaign is executed by using the proposed model for solving different instances for Scenario 3 with a particular layout of the yard, that is only one bay-locations capacity is present, i.e. 20 containers bay-locations capacity. The effectiveness of the proposed model is evaluated for increasing size instances in terms of number of containers and in terms of destinations. We conduct experiments with 200, 400, 800, and 1600 containers.

CPU Time limit is fixed 3600 seconds for instances with 200 and 400 containers, while it is 7200 seconds for instances with 800 containers and 14400 seconds for the largest instances with 1600 containers. Results and detailed information are listed in Table 3.

Looking at Table 3 we can note that all instances with 200 and 400 containers have been solved up to optimality requiring an average CPU time of 77 and 311 seconds, respectively. Larger instances present high gaps even if the time limit has been increased up to 7200 and 14400 seconds for instances with 800 and 1600 containers, respectively. We can note that larger instances with only one destination present lower gaps than those with either 4 or 8 destinations.

Summarizing, from both the results shown in Table 3 and in Figures 5 and 6 (where CS1 and CS2 were compared), we can say that the number of different capacities of bay-locations strongly affects the proposed model performances. Moreover, it is clear that the number of destinations is another impacting factor on the capability of the model to solve the real size instances. However, containers are generally (almost always) grouped according to their destination, implying that it is possible to decompose the storage problem. Thus, from the above considerations we have decided to implement the solution approach described in Section 4. In the next sub-section we compare the solution obtained by the proposed model and the solution approach, with the solution used by the terminal under investigation for some real instances.

5.3 Some real case instances

As final test, we present a comparison of the results obtained by using the storage strategy solution of the proposed model, of the proposed solution approach for tree real instances, two belonging to the set of SS and MS and the last one that is larger, characterized by 646 containers and 9 different destinations. In this case we can compare the obtained solutions also with the storage strategy adopted by the terminal under investigation.

The comparison in Table 4 shows that either proposed model or solution approach outperforms the current storage plan of the terminal under investigation in terms of empty space and bay-locations. The solutions obtained by using the proposed heuristic approach grant a saving that ranges from 7% to 56% of empty slots, and from 16% to 53% of bay-locations used. Comparing results of the model and the solution approach we can note that generally the heuristic approach is worst in terms of empty slots. The proposed approach seems to be promising and helpful for yard storage managers.

6 Conclusions

In this paper we have tried to implement a solution approach for helping yard manager in defining the best storage strategies to use for minimizing the space used. We have shared the obtained results with the maritime terminal under investigation that really appreciate this approach having a lots of problems due to the lack of space in the yard. The idea is to solve this problem each time there is a significant change in the transport demand that can require a change in the storage strategy. The storage strategy defined, in particular the number of classes and the weight limitations are inserted as parameters in the TOS (terminal operation system) of the terminal that manage the real time storage of the flow of containers reaching the terminal by trains and by trucks. The proposed approach provides maximum freedom to terminal manager choosing different storage strategies in accordance with numerous requests. It permits to decide the most appropriate combination of characteristics and configurations for reorganizing the storage plan granting a better space utilization.

As future work it should be interesting to extend this problem in such a way to consider it in a dynamic way. A vessel of a service can visit the terminal for example once a week, and can occur that containers arrive too early at the terminal and, since they have to be loaded on the vessel of the next week, they have to wait; thus, it is necessary to manage together containers that must be loaded in two different vessels of the same service. This is the new request of the terminal we are involved with.

Declarations

Funding

This research has been supported by MUR- Italy - project PRIN2015 research program: SPORT: Smart PORt Terminals

Conflict of interest

The authors declare that they have no conflict of interest

Availability of Data and Material

Real data are not available for privacy reasons. Generated instances are available on request.

References

1. Ambrosino, D., Xie, H.: An optimization model for defining storage strategies for export yards in container terminals: A case study. In: E. Lalla-Ruiz, M. Mes, S. Voß (eds.) Computational Logistics - 11th International Conference, ICCL 2020, Enschede, The Netherlands, September 28-30, 2020, Proceedings, *Lecture Notes in Computer Science*, vol. 12433, pp. 119–132. Springer (2020). DOI 10.1007/978-3-030-59747-4_8. URL https://doi.org/10.1007/978-3-030-59747-4_8
2. Carlo, H.J., Vis, I.F., Roodbergen, K.J.: Storage yard operations in container terminals: Literature overview, trends, and research directions. European Journal of Operational Research **235**(2), 412 – 430 (2014)
3. Han, Y., Lee, L.H., Chew, E.P., Tan, K.C.: A yard storage strategy for minimizing traffic congestion in a marine container transshipment hub. OR spectrum **30**(4), 697–720 (2008)
4. He, J., Tan, C., Yan, W., Huang, W., Liu, M., Yu, H.: Two-stage stochastic programming model for generating container yard template under uncertainty and traffic congestion. Advanced Engineering Informatics **43**, 101032 (2020). DOI https://doi.org/10.1016/j.aei.2020.101032

Table 4 Comparison among the solutions obtained by the proposed approaches and the terminal solution

	Proposed model		Solution approach		Terminal Sol.	
Instances	Empty space	Bay-locations	Empty space	Bay-locations	Empty space	Bay-locations
SS	14	7	26	7	30	15
MS	39	25	71	25	65	30
real inst.	-	-	58	44	134	89

5. Kim, K.H., Park, K.T.: A note on a dynamic space-allocation method for outbound containers. European Journal of Operational Research **148**(1), 92 – 101 (2003)
6. Kim, K.H., Park, Y.M., Ryu, K.R.: Deriving decision rules to locate export containers in container yards. European Journal of Operational Research **124**(1), 89–101 (2000)
7. Koster, R.D., Le-Duc, T., Roodbergen, K.J.: Design and control of warehouse order picking: A literature review. European Journal of Operational Research **182**(2), 481 – 501 (2007)
8. Kozan, E., Preston, P.: Mathematical modelling of container transfers and storage locations at seaport terminals. In: Container Terminals and Cargo Systems, pp. 87–105. Springer (2007)
9. Lee, C., Tan, H.: An optimization model for storage yard management in transshipment hubs. OR Spectrum **28**, 539–561 (2006). DOI 10.1007/s00291-006-0045-4
10. Legato, P., Canonaco, P., Mazza, R.M.: Yard crane management by simulation and optimisation. Maritime Economics & Logistics **11**(1), 36–57 (2009)
11. Moorthy, R., Teo, C.: Berth management in container terminal: The template design problem. OR Spectrum **28**, 495–518 (2006)
12. Petering, M.E.: Effect of block width and storage yard layout on marine container terminal performance. Transportation Research Part E: Logistics and Transportation Review **45**(4), 591–610 (2009)
13. Petering, M.E., Murty, K.G.: Effect of block length and yard crane deployment systems on overall performance at a seaport container transshipment terminal. Computers & Operations Research **36**(5), 1711–1725 (2009)
14. Saanen, Y., Dekker, R.: Intelligent stacking as way out of congested yards part 1. Port Technology International **31**, 87–92 (2007)
15. Sha Mei, Z.X.Q.T.b.Y.D.y.Q.H.l.: Study on layout optimization and simulation of container yard [j]. Industrial Engineering and Management **2** (2013)
16. Tan, C., He, J., Wang, Y.: Storage yard management based on flexible yard template in container terminal. Advanced Engineering Informatics **34**, 101 – 113 (2017). DOI https://doi.org/10.1016/j.aei.2017.10.003
17. Tan, C., He, J., Yu, H.: Mathematical modeling of yard template regeneration for multiple container terminals. Advanced Engineering Informatics **40**, 58 – 68 (2019). DOI https://doi.org/10.1016/j.aei.2019.03.004
18. Wiese, J., Suhl, L., Kliewer, N.: Mathematical models and solution methods for optimal container terminal yard layouts. OR spectrum **32**(3), 427–452 (2010)
19. Woo, Y.J., Kim, K.H.: Estimating the space requirement for outbound container inventories in port container terminals. International Journal of Production Economics **133**(1), 293 – 301 (2011)
20. Yu, H., Zhang, M., He, J., Tan, C.: Choice of loading clusters in container terminals. Advanced Engineering Informatics **46**, 101190 (2020). DOI https://doi.org/10.1016/j.aei.2020.101190
21. Zhang, C., Liu, J., Wan, Y.w., Murty, K., Linn, R.: Storage space allocation in container terminals. Transportation Research Part B: Methodological **37**, 883–903 (2003)
22. Zhen, L.: Container yard template planning under uncertain maritime market. Transportation Research Part E: Logistics and Transportation Review **69**, 199 – 217 (2014)

Appendix

Scenario	Test ID	Status of class configuration	Status of weight configuration	Classes	Weight limits
Scenario1	1-2a	1 fixed	fixed	Light/Heavy	0-15/15-33
	1-2b			Light/Heavy	0-20/20-33
	1-3a			Light/Medium/Heavy	0-12/12-22/22-33
	1-3b			Light/Medium/Heavy	0-15/15-25/25-33
	1-4a			Light/Medium/Heavy/Extra	0-12/12-20/20-28/28-33
	1-4b			Light/Medium/Heavy/Extra	0-10/10-18/18-25/25-33
Scenario2	2-2	1 fixed	free	Light/Heavy	0-15/15-33 or 0-20/20-33
	2-3			Light/Medium/Heavy	0-12/12-22/22-33 or 0-15/15-25/25-33
	2-4			Light/Medium/Heavy/Extra	0-12/12-20/20-28/28-33 or 0-10/10-18/18-25/25-33
Scenario3	3-fix	free	fixed	Light/Heavy or Light/Medium/Heavy or Light/Medium/Heavy/Extra	0-20/20-33 0-15/15-25/25-33 0-10/10-18/18-25/25-33
Scenario4	3-free	free	free	Light/Heavy or	0-15/15-33 or 0-20/20-33
				Light/Medium/Heavy or	0-12/12-22/22-33 or 0-15/15-25/25-33
				Light/Medium/Heavy/Extra	0-12/12-20/20-28/28-33 or 0-10/10-18/18-25/25-33

Fig. 9 Characteristics' scenarios

Status of class configuration	Status of weight configuration	Test ID	Layout	Instance	# empty slots	# bay-locations used	CPU(s)	GAP
1 fixed	all fixed	1_2a	CS1	SS	30	7	3.26s	0%
			CS2	SS	14	7	4.79s	0%
		1_2b	CS1	SS	34	7	3.37s	0%
			CS2	SS	22	7	5.64s	0%
		1_3a	CS1	SS	70	9	3.92s	0%
			CS2	SS	46	9	4.90s	0%
		1_3b	CS1	SS	70	9	3.04s	0%
			CS2	SS	46	9	3.48s	0%
		1_4a	CS1	SS	106	11	3.48s	0%
			CS2	SS	74	11	5.12s	0%
		1_4b	CS1	SS	126	12	3.70s	0%
			CS2	SS	94	12	5.23s	0%
1 fixed	fixed	1_2a	CS1	MS	91	25	1762s	0%
			CS2	MS	39	25	3600s	28.12%
		1_2b	CS1	MS	95	25	1127s	0%
			CS2	MS	43	25	3600s	42.65%
		1_3a	CS1	MS	223	32	1315s	0%
			CS2	MS	131	32	3600s	33.74%
		1_3b	CS1	MS	207	31	683s	0%
			CS2	MS	123	31	3600s	68.83%
		1_4a	CS1	MS	319	37	906s	0%
			CS2	MS	203	37	3600s	31.25%
		1_4b	CS1	MS	339	38	556s	0%
			CS2	MS	227	38	3600s	23.77%
1 fixed	free	2_2	CS1	SS	30	7	5.23s	0%
			CS2	SS	14	7	16.82s	0%
		2_3	CS1	SS	70	9	8.93s	0%
			CS2	SS	46	9	77s	0%
		2_4	CS1	SS	106	11	289s	0%
			CS2	SS	74	11	1034s	0%
1 fixed	free	2_2	CS1	MS	91	25	5773s	0%
			CS2	MS	43	25	10800s	52.94%
		2_3	CS1	MS	207	31	8643s	0%
			CS2	MS	123	31	10800s	74.20%
		2_4	CS1	MS	319	37	10800s	89.88%
			CS2	MS	203	37	10800s	91.25%
free	fixed	3-fix	CS1	SS	34	7	46.68s	0%
			CS2	SS	22	7	2742s	0%
free	fixed	3-fix	CS1	MS	95	25	14400s	44.35%
			CS2	MS	43	25	14400s	69.11%
free	free	3-free	CS1	SS	30	7	968s	0%
			CS2	SS	14	7	679s	0%
free	free	3-free	CS1	MS	95	25	14400s	74.16%
			CS2	MS	43	25	14400s	73.52%

Fig. 10 Results of different scenarios