IMA Journal of Management Mathematics (2003) **14**, 251–269

The master bay plan problem: a solution method based on its connection to the three-dimensional bin packing problem

ANNA SCIOMACHEN† AND ELENA TANFANI‡

Dipartimento di Economia e Metodi Quantitativi (DIEM), Università di Genova, Via Vivaldi 5, 16126 Genova, Italy

This paper addresses the problem of determining stowage plans for containers in a ship, referred to as the Master Bay Plan Problem (MBPP).

MBPP is NP-complete. We present a heuristic method for solving MBPP based on its relation with the three-dimensional bin packing problem (3D-BPP), where items are containers and bins are different portions of the ship. Our aim is to find stowage plans, taking into account structural and operational constraints related to both the containers and the ship, that minimize the time required for loading all containers on board.

A validation of the proposed approach with some test cases is given. The results of real instances of the problem involving more than 1400 containers show the effectiveness of the proposed approach for large scale applications.

Keywords: stowage plan; three-dimensional bin packing problem; heuristic algorithm.

1. Introduction

The stowage of a containership is one of the problems that has to be solved daily by any company which manages a container terminal (Thomas, 1989). In the past, stowage plans for containers were produced by the Captain of the ship; today, the maritime terminal has to establish the master bay plan, in accordance with the stowage instruction of the ship co-ordinator representing the company holding the ship.

The ship planning problem involves different objectives, such as optimal space allocation, optimal synchronization among dispatching operations and minimization of the berthing time (Atkins, 1991).

Formally, the stowage planning problem, known as the Master Bay Plan Problem (MBPP), consists of determining how to stow a set C of n containers of different types into m available locations of a containership, with respect to some structural and operational constraints, related to both the containers and the ship, while minimizing the time required for loading all containers on board.

MBPP is a NP-complete combinatorial optimization problem.

Interesting heuristics for dealing with the container loading problem can be found in Bischoff & Mariott (1990), Bischoff & Ratcliff (1995), Bortfeldt & Gehring (2001), Davies & Bischoff (1999), Gehring & Menscher e Meyer (1990), Gehring & Bortfeldt (1997) and Raidl (1999) among others. Terno *et al.* (2000) consider many practical restrictions arising

†Email: sciomach@economia.unige.it

‡Email: etanfani@economia.unige.it

in the multi-pallet loading problem, such as weight, placement and stability conditions. Crainic *et al.* (1993) present stochastic models for the allocation of empty containers.

The application of Mathematical Programming models to MBPP has been investigated in Botter & Brinati (1992), Chen *et al.* (1995) and Imai *et al.* (2002), but many simplifications to the problem make the proposed Linear Integer Programming models unsuitable for practical applications with respect to our specific problem.

Ambrosino *et al.* (2002) split the set of bays of a containership and solve a 0/1 Linear Programming model for MBPP by using a Branch & Bound algorithm for each subset of bays, looking successively for the global stability of the ship by performing multi-exchanges.

Rule-based decision systems for dealing with MBPP are presented in Ambrosino & Sciomachen (1998), where a constraint satisfaction approach is used for defining and characterizing the space of feasible solutions without an objective function to optimize, and in Wilson & Roach (2000), where the potential of applying the theory of artificial intelligence to cargo stowage problems is explored.

Avriel & Penn (1993) consider the minimum shift problem, that is the problem of finding stowage plans with the smallest number of shifts, and Avriel *et al.* (2000) connect it to the colouring of circle graphs problem.

In this paper we show the connection between MBPP and 3D-BPP and introduce a heuristic procedure, which is based on the framework of the exact branch-and-bound algorithm for 3D-BPP presented by Martello *et al.* (2000).

A description of the practical details of MBPP is given in Section 2. The main commonalties and differences between 3D-BPP and MBPP are shown in Section 3. The main steps of our algorithm for MBPP are presented in Section 4. In Section 5 we report some computational experiments performed with real life instances involving more than 1400 containers, showing the effectiveness of the proposed approach in terms of quality of the solutions and computational time. Finally, in Section 6 we give some concluding remarks and outline future work.

2. Definition of MBPP

The following points outline the main constraints which must be considered for the stowage planning process for an individual port (Ambrosino *et al.*, 2004).

- The constraints related to the structure of the ship are focused on the type, size and weight of the containers to be loaded. Each location of the ship is addressed by the following identifiers (see Fig. 1): (a) *bay*, that gives its position related to the cross section of the ship (counted from bow to stern); (b) *row*, that gives its position related to the horizontal section of the corresponding bay (counted from the centre to the outside); (c) *tier*, that gives its position related to the vertical section of the corresponding bay (counted from the bottom to the top of the ship).

- As far as the type of containers, we consider dry containers, dry high cube containers, open top containers, platforms, reefers and tank containers; the specifications of such containers are given in Table 1. The exterior dimensions of containers conforming to ISO standards are 10, 20, 30 and 40 feet long $\times$ 8 feet 6 inches high and 8 feet depth.

- Standard locations are generally sought for dry 20′ containers and denoted as one Twenty Equivalent Unit (TEU) location. Containers of 40′ require two contiguous locations of 20′. Moreover, smaller containers cannot be stacked above larger ones, since each container needs to be fixed by four twisters to the upper corners of the container below it. Note that in Figure 1 the location denoted by A represents a standard TEU location, while those denoted by B are devoted to the stowage of 40′ containers.
- Locations of reefers are usually defined in advance by the ship co-ordinator, and are generally near plugs in order to maintain the required temperature during transportation. Hazardous containers and tanks are predestined too by the harbour-master's office which authorizes their loading.
- Weight constraints force the weight of a stack of containers to be less than a given tolerance value; moreover, the weight of a container located in a tier cannot be greater than the weight of the container located below it in the same row and bay.
- Operational and security constraints are related to the weight distribution on the ship. In particular, after any loading/unloading operations different kinds of equilibrium have to be checked, namely: cross equilibrium, that is the weight on the right side of the ship must be equal, within a given tolerance, to the weight on the left side of the ship; horizontal equilibrium, that is the weight on the stern must be equal, within a given tolerance, to the weight on the bow; vertical equilibrium, that is the weight on each tier must be greater than the weight on the tier immediately over it.
- Finally, destination constraints give a general rule which suggests loading first those containers having as destination the final stop of the ship and, consequently, loading last those containers that have to be unloaded first. Without considering the unloading port some containers loaded last could be necessarily moved for enabling the unloading operations of some others. The 'shifting moves' are very unproductive for any maritime company since they increase the berthing time at port thus affecting the cost of the whole trip of the containership.

3. The connection between 3D-BPP and MBPP

As has been already said, the proposed heuristic procedure for solving MBPP is based on its relation with the three-dimensional bin packing problem.

Given a set of n rectangular-shaped *items*, each one characterized by width w_j, height h_j, and depth d_j, ($j \in J = \{1, \ldots, n\}$), and a set of identical three-dimensional containers (*bins*) having width W, height H, and depth D, 3D-BPP consists of orthogonally packing all items into the minimum number of bins. Considering *items* as *containers* and *bins* as different *portions of the ship* (see Section 4.1), our aim is to connect 3D-BPP to MBPP and smoothing out all the differences between the problems.

The basic idea of our solution method for MBPP starts from the exact branch-and-bound algorithm for 3D-BPP presented by Martello *et al.* (2000), where it is assumed that (a) items may not be rotated, (b) items are packed with each edge parallel to the corresponding bin edge, (c) all data are positive integers such that $w_j \leqslant W$, $h_j \leqslant H$, and $d_j \leqslant D, \forall\ j \in J$.

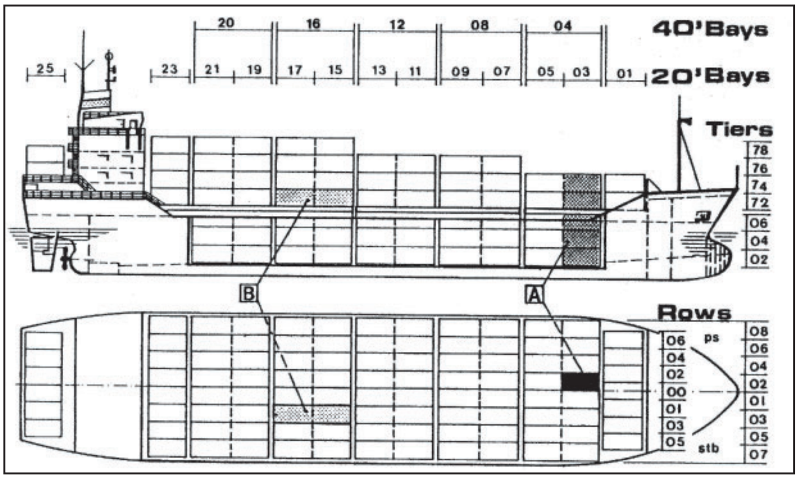

FIG. 1. Cross, vertical and horizontal sections of a standard containership.

TABLE 1 *Container specifications*

Container type	Container sizes	Container applications
Dry	$10' \times 8' \times 8'6''$, $20' \times 8' \times 8'6''$, $30' \times 8' \times 8'6''$, $40' \times 8' \times 8'6''$	General purpose container
Dry high cube	$20' \times 8' \times 9'6''$, $40' \times 8' \times 9'6''$	High container for over height and voluminous cargo
Open top	$20' \times 8' \times 8'6''$, $40' \times 8' \times 8'6''$	Removable tarpaulin and roof bows. Easy top loading of over sized cargo
Platform	$20' \times 8' \times 2'$, $40' \times 8' \times 2'$	For extra length and heavy cargo
Reefer	$20' \times 8' \times 8'6''$, $40' \times 8' \times 8'6''$, $40' \times 8' \times 9'6''$	For cooling, freezing of foods or chemicals
Tank	$20' \times 8' \times 8'6''$	For transportation of liquid chemicals

These assumptions are necessary for MBPP. In particular, assumptions (a) and (b) are not to be underestimated. These strong restrictions are usually not considered in the bin packing problem and items can be rotated (see e.g. Bischoff & Mariott, 1990; Faina, 2000; Gehring & Menscher e Meyer, 1990; George & Robinson, 1980; Mohanty *et al.*, 1994 and Pisinger, 1998). Assumption (b) is not really important in packing practice but is absolutely necessary for the definition of stowage plans since containers have to be stowed only in one orthogonal direction.

The mentioned enumerative algorithm for 3D-BPP iteratively solves associated subproblems in which all items of a given subset $J' \subseteq J$ have to be packed into a single bin (if it is possible), with the aim of maximizing the total volume of the packed items. In particular, a procedure called *main branching tree* assigns items to bins without specifying their actual position, and a branch-and-bound algorithm, called *onebin*, verifies whether a subset of items $J' \subseteq J$ can be placed inside a single bin and, if it is the case, finds the best filling of the single bin using items belonging to J'.

A considerable effort is required in order to ensure feasibility for MBPP optimal 3D-BPP solutions; in fact, we have to consider all structural and operational constraints, related to both the containers and the ship, that any final stowage plan has to satisfy. Moreover, our goal is to minimize the total loading time, which is one of the most important productivity indicators of maritime terminal competitiveness.

In practice, even if there is a general commonalty between MBPP and 3D-BPP, there are many practical objections.

First, analysing the input data we can see that 3D-BPP considers only cubic bins, while in our problem the shape and the structure of the ship are important factors that have to be taken into account, as well as the characteristics of the containers (see Section 2). Moreover, the dimensions of the items are different from the sizes of the containers. Note that, even if in 3D-BPP it is usually assumed that a larger number of objects having different sizes than in MBPP is given, the exact algorithm proposed by Martello *et al.* (2000) is in any case an efficient basis for MBPP, where only the types of the containers shown in Table 1 are considered.

Second, the 3D-BPP algorithm under consideration starts to position the items from the back left bottom corner and continues to fill the bin in a vertical pattern; instead, in the case of stowage plans it would be better to follow horizontal patterns due to the stability of the ship. Moreover, in the 3D-BPP formulation the presence of empty spaces between items is allowed.

Third, because of the characteristics of 3D-BPP, largest items are packed first, as they have more difficulties in their placement; this order of preference applied to our problem arises in stowing first the largest containers and, consequently, $10'$ and $20'$ containers are often positioned over them, thus violating the size constraints (see Section 2).

Finally, in 3D-BPP the position of the items is given in a three-dimensional Cartesian space, with the origin of the axis located at the back left bottom corner, while for the definition of stowage plans we have to know the exact position of the containers, as has been explained in Section 2.

4. The proposed heuristic algorithm for MBPP

The heuristic procedure presented here is intended to exploit the potential of the 3D-BPP algorithm proposed by Martello *et al.* (2000) and find solutions structurally and operationally feasible for the stowage of a containership. Our objective function is the minimization of the total loading time, that is given by the sum of the times required for loading all containers in their assigned location. We assume that the handling operations are performed by yard cranes, whose exemplifying loading times are reported in Table 2; we can see that the value between two contiguous locations increases when we move from the quay side (odd rows) going to the bottom (see Fig. 1), since the locations are more difficult to be reached. Note that any other linear function for the loading times, which depends on the location where a container is put, can be properly used.

We consider the constraints described in Section 2 and use the positioning pattern, and the corresponding 'enumeration' of the containers, followed in the case of 3D-BPP for 'packing' subsets of containers into different portions of the ship, such that the feasibility of the solution is not violated.

The main phases of our approach for finding feasible solutions are now stated.

TABLE 2 *Example of loading times (in seconds) as a function of row and tier indices*

	T02	T04	T06	T72	T74
R04	190	180	170	160	150
R02	180	170	160	150	140
R01	170	160	150	140	130
R03	160	150	140	130	120

4.1 *The complex structure of the ship*

The shape of a ship is different from a standard six-face solid, which is utilized as the bin in 3D-BPP. In fact, we have to distinguish the hold, the upper deck, the bow, the stern and some particular zones where it is not possible to stow containers; moreover, we have to consider the particular 'slanting' shape of the hold. We split the ship into different sections in order to be able to distinguish the above components; each section can hence be considered as a bin.

Consider, as a numerical example, how we can split the hold of a 512 TEU containership, with four tiers, eight rows and 16 TEU bays, into different sections. Figure 2 show the tiers $(02, 04, 06, 08)$ of the ship, the $20'$ standard bays $(01, 03, 05, \dots, 31)$, the corresponding $40'$ bays $(02, 06, 10, \dots, 30)$ and the rows $(08, 06, 04, 02, 01, 03, 05, 07)$. The 'dark' slots are not allowed for stowing containers (note that, because of the natural 'slanting' shape of the hold, the lowest tiers are narrower in the external parts).

We call the biggest parallelepiped shape portion of the ship the *main section*. Usually, the main section consists of about 80% of the total available stowing area (TEUs) of the ship. The main section itself is successively split into k homogeneous sections, denoted *normal sections* N_i, $i = 1, 2, \dots, k$, according to the stability requirements. We derive value k from (1), where b and r are, respectively, the number of bays and rows of the main section:

$$k = \frac{br}{4}. \tag{1}$$

Let T be the number of TEUs available in any normal section N_i, $i = 1, \dots, k$.

We call the other portions of the ship *special sections*; they consist of the remaining parts of the ship and are considered and loaded separately since they correspond to (see Fig. 2): (a) the lowest tier/tiers that can be to small for belonging to the main section; (b) portions of the sides of the main section at different tiers; (c) bays where it is possible to stow only a few $10'$ and $20'$ containers. Special sections S_j, $j = 1, 2, \dots, q$, are numbered according to an increasing value of their tier and row, while alternating their bay index, due to stability reasons.

4.2 *The destination constraints*

The destination constraints are considered by using the ship co-ordinator's instructions, that assign the containers with a given destination to predetermined groups of locations (identified by bays), in such a way that any section is stowed only with containers having the same destination.

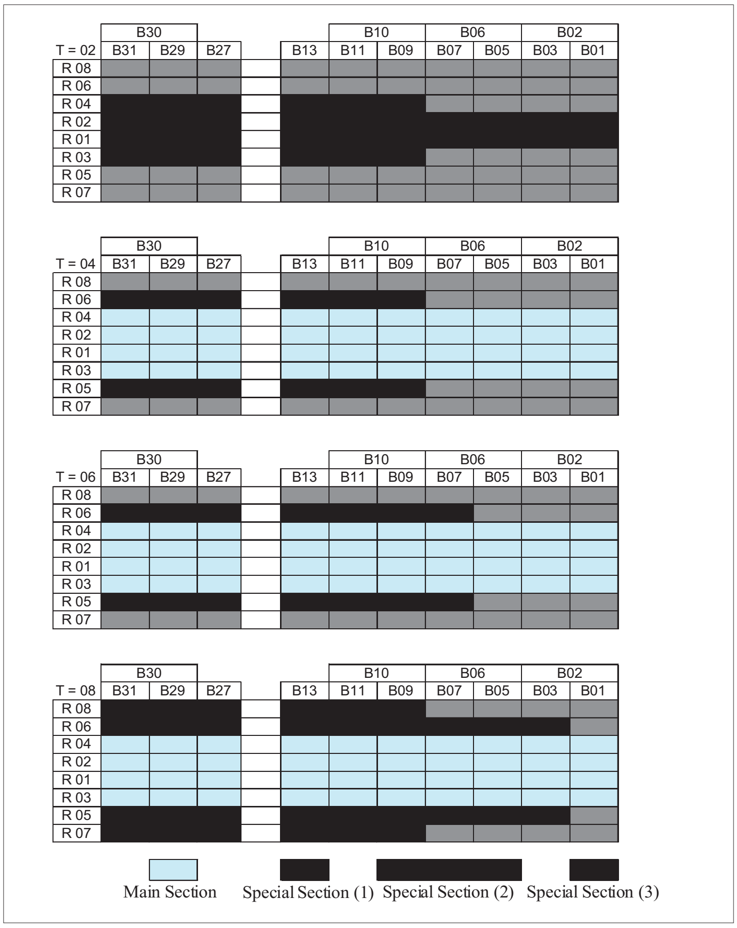

FIG. 2. Normal and special sections of a 512 TEU hold containership.

We split the set C of all containers to be stowed in the ship into p subsets C_h, $h = 1, \dots, p$, where p is the number of different ports visited by the ship and C_h is the set of containers having port h as final destination. All containers are hence grouped together according to their destination, such that $\bigcup_{h=1,\dots,p} C_h = C$ and $C_g \cap C_h = \varnothing, \forall\ h \neq g, h, g = 1, \dots, p$.

Let T_h be the total number of TEUs related to the containers belonging to C_h, $h = 1, \dots, p$. We compute the number k_h of normal sections required to stow all containers belonging to C_h, $h = 1, \dots, p$, as

$$k_h = \left\lceil \frac{T_h}{T} \right\rceil. \tag{2}$$

Note that if $\sum_{h=1}^{p} k_h > k$ some special sections among the q available ones are required for stowing containers belonging to C_h, $h = 1, \dots, p$. In this case we try to minimize the number of sections (bins) used for stowing all containers by following the criterion used by the exact 3D-BPP algorithm. Note that the minimization of the special sections filled implies the minimization of our objective function, that is the loading time, since they are generally located in different zones of the ship that could be far from each other; hence, an additional time, for positioning the yard cranes along the quay, could be required for loading containers in S_j, $j = 1, 2, \ldots, q$.

After this partitioning phase, we assign containers with destination h, $h = 1, \ldots, p$, to k_h normal sections according to stability considerations and using the bay assignment procedure proposed in Ambrosino *et al.* (2002).

4.3 *The ship stability constraints*

We balance and share out the total weight of the loaded containers and satisfy the horizontal and cross equilibrium of the ship by assigning a priori a given number of containers to each normal section on the basis of their weight. For simplicity, let us consider set C_h, $h = 1, \ldots, p$, and suppose that normal sections N_i, $i = 1, \ldots, k_h$, are devoted to the stowage of containers having port h as final destination; the same consideration holds for all the other sections.

We split set C_h into three subsets $C_{h(w)}$, $w = 1, 2, 3$, such that $C_{h(1)}$ consists of 'light' containers (up to 15 tons), $C_{h(2)}$ consists of 'medium' containers (ranging from 15 to 25 tons) and $C_{h(3)}$ consists of 'heavy' containers (more than 25 tons). Then, we assign χ_i containers to N_i such that

$$\chi_i = \sum_{w=1}^{3} \left\lfloor \frac{|C_{h(w)}|}{k_h} \right\rfloor. \tag{3}$$

Note that $\chi = \sum_{i=1}^{k} \chi_i \leqslant n$. If $\chi < n$ we assign the remaining containers to different normal sections according to the destination and stability requirements and providing that $|W_i - W_j| \leqslant \Delta$, $i \neq j$, $i, j = 1, \ldots, k_h$, where W_i is the total weight of the containers in section N_i and Δ is a given tolerance value between the weight of different sections of the ship.

Recalling that the 3D-BPP algorithm starts to position the items from the left bottom corner of the bin (the origin (0, 0, 0)), and follows a vertical pattern concentrating the items near this starting point, now in each normal section N_i, $i = 1, \ldots, k_h$, we have to find the origin point to start loading containers. We use an alternate criterion that, starting from N_1, determines the origin of each normal section according to the following pattern (see Fig. 3 in Section 4.6 and Fig. 5 in Section 5.1):

Normal section	Origin
N_1: smallest bays, even rows	left bottom corner
N_2: same bays, odd rows	right top corner
N_3: next bays, odd rows	left bottom corner
N_4: same bays, even rows	right top corner
etc.	

Successively, through an axis rotation we consider the x axis as the depth (instead of the width), the y axis as the width (instead of the height) and the z axis as the height (instead of the depth). Consequently, since the filling pattern follows a '$y \to x \to z$' sequence, the stowage of the containers will be first by width, then by depth and finally by height, so achieving the horizontal pattern.

4.4 *The size and weight constraints of the containers*

In order to avoid putting smaller containers above larger ones, the containers assigned to N_i, $i = 1, \ldots, k$, are sorted in an increasing order of their size and in a decreasing order of their weight, such that we choose first, for being loaded in the lowest tiers, the smallest heaviest containers. By using this loading criterion we avoid the violation of the size constraints (see Section 2) and the presence of empty spaces between containers (that could occur by applying the 3D-BPP algorithm). Moreover, by using this ordering rule we follow the 'from bottom to top' order utilized by the 3D-BPP algorithm as loading sequence and satisfy the vertical equilibrium constraints. Note that, in this way, it is also easy to check the weight tolerance of a stack of three containers.

4.5 *The main steps of the algorithm*

The first phase of the proposed solution method is the implementation of a procedure for the acquisition of all input data related to the containers (their destination, size and weight) and the structure of the ship. Then, the main steps of the algorithm can be synthesized as follows.

Step 1. Identify the main section of the ship and compute according to (1) k normal sections (bins) N_i, $i = 1, \ldots, k$; consequently, derive those special sections S_j, $j = 1, \ldots, q$, that, following conditions (a), (b) and (c) given in Section 4.1, cannot belong to the main section;

Step 2. Split set C of containers into p subsets C_h, one for any given destination, and compute from (2) the number k_h of normal sections required for stowing containers with destination h, $h = 1, \ldots, p$;

Step 3. Split C_h, $h = 1, \ldots, p$, into three subsets according to the weight of the containers belonging to it and assign χ_i containers given by (3) to normal section N_i, $i = 1, \ldots, k_h$;

Step 4. In normal section N_i, $i = 1, \ldots, k$, find the origin point following the criterion given in Section 4.3;

Step 5. Sort the containers assigned to N_i, $i = 1, \ldots, k$, in an increasing order of their size and in decreasing order of their weight, such that the smallest containers are loaded first, starting from the origin and following the pattern used by the 3D-BPP algorithm for the effective positioning of the containers;

Step 6. If all containers assigned to N_i, $i = 1, \ldots, k$, are loaded in the corresponding bin go to Step 8;

TABLE 3 *Containers specifications and ordered list of each section*

Port	Type	Section	Weight	Order in section list	Port	Type	Section	Weight	Order in section list
1	$10'$	N_1	5	12	2	$10'$	N_2	10	12
1	$10'$	N_1	10	13	2	$10'$	N_2	10	13
1	$10'$	N_3	5	12	2	$10'$	N_4	5	12
1	$10'$	N_3	10	13	2	$10'$	N_4	10	13
1	$20'$	N_1	5	3	2	$20'$HC	N_2	15	5
1	$20'$	N_1	10	4	2	$20'$HC	N_2	15	6
1	$20'$	N_1	10	5	2	$20'$	N_2	15	7
1	$20'$	N_1	10	6	2	$20'$	N_2	15	8
1	$20'$	N_1	10	7	2	$20'$	N_2	15	9
1	$20'$	N_1	15	8	2	$20'$	N_2	20	10
1	$20'$	N_1	15	9	2	$20'$	N_2	25	11
1	$20'$	N_1	15	10	2	$20'$	N_4	15	5
1	$20'$	N_1	15	11	2	$20'$	N_4	15	6
1	$20'$	N_3	5	3	2	$20'$	N_4	15	7
1	$20'$	N_3	5	4	2	$20'$	N_4	15	8
1	$20'$HC	N_3	10	5	2	$20'$	N_4	20	9
1	$20'$HC	N_3	10	6	2	$20'$	N_4	20	10
1	$20'$	N_3	10	7	2	$20'$	N_4	25	11
1	$20'$	N_3	15	8	2	$40'$	N_2	10	1
1	$20'$	N_3	15	9	2	$40'$HC	N_2	20	2
1	$20'$	N_3	15	10	2	$40'$	N_2	25	3
1	$20'$	N_3	15	11	2	$40'$	N_2	25	4
1	$40'$	N_1	10	1	2	$40'$	N_4	20	1
1	$40'$HC	N_1	10	2	2	$40'$HC	N_4	25	2
1	$40'$HC	N_3	10	1	2	$40'$	N_4	25	3
1	$40'$	N_3	10	2	2	$40'$	N_4	25	4

Step 7. Let $\bar{C}$ be the set of all containers left out from the stowage in the normal sections and sort it in decreasing order according to their size. Start to put containers in special section S_j, $j = 1, \ldots, q$, as in Section 4.2, starting from the origin computed as in Section 4.3;

Step 8. Return the final positions of the containers, originally given in the Cartesian coordinates (x, y, z), in terms of their bay, row and tier indices (see Fig. 1) and give a picture of the master bay plan.

Note that Steps 1–8 of our heuristic algorithm are used first for stowing the hold and successively for the upper deck, thus guaranteeing that the heaviest containers are positioned in the lowest tiers.

4.6 *A simple example*

To give an idea of how the proposed approach for MBPP works, let us present a simple case study. We are involved with a prototype of a 64 TEU containership with four bays,

MASTER BAY PLAN

Tier 02

	Bay 01	Bay 03	Bay 05	Bay 07
Row 04	15	15	20	5 | 10
Row 02	5 | 10	15	20	25
Row 01	20	10 | 10	15	15
Row 03	15	25	5 | 10	15

Tier 04

	Bay 01	Bay 03	Bay 05	Bay 07
Row 04	10	10	15	15
Row 02	15	10	15	15
Row 01	15	15	10	10
Row 03	15	15	15	10

Tier 06

	Bay 01	Bay 03	Bay 05	Bay 07
Row 04	5	5	12,5	12,5
Row 02	10	5	12,5	12,5
Row 01	12,5	12,5	5	5
Row 03	12,5	12,5	5	5

Tier 72

	Bay 01	Bay 03	Bay 05	Bay 07
Row 04			12,5	12,5
Row 02	5	5	10	10
Row 01	10	10		
Row 03	10	10	5	5

WEIGHT CONFIGURATION

Total 370	370	Total
140 N_1	235 N_4	375
230 N_2	135 N_3	365

Weight distribution

60	115	
102,5	180	92,5
115	75	

%

8,108	15,541	
13,851	24,324	12,500
15,541	10,135	

Total weight: 740

Trasversal comparison

60	115
80	120
115	60
115	75

FIG. 3. Final MBP and global weight configuration of the simple case.

four rows and four tiers, where we have to stow $n = 52$ containers split into $10'$, $20'$, $40'$ standard dry containers, and $20'$ and $40'$ height cube; the ship visits two different ports, namely ports 1 and 2 (see Table 3).

Following Steps 1–8 of our algorithm, we first identify the main section of the prototype ship, that, due to the small size, represents the whole ship; therefore, according to (1), we identify four normal sections (N_1, N_2, N_3, N_4).

Secondly, we split set C into two subsets C_1 and C_2, one for each given destination, and, according to (2), assign sections N_1 and N_3 to containers destined to port 1, and sections N_2 and N_4 to containers destined to port 2. Successively, we find the origin points depicted in Fig. 3.

Finally, we assign containers to each section without specifying their actual position, and give their ordered list that is reported in Table 3.

Figure 3 reports the master bay plan obtained at the end of Steps 5 and 8, and the corresponding weight configuration. In Fig. 4 we give its 3D representation.

5. Computational experiments

The proposed heuristic algorithm has been implemented in the C programming language and used to solve real life instances of MPBB in order to test the feasibility and optimality

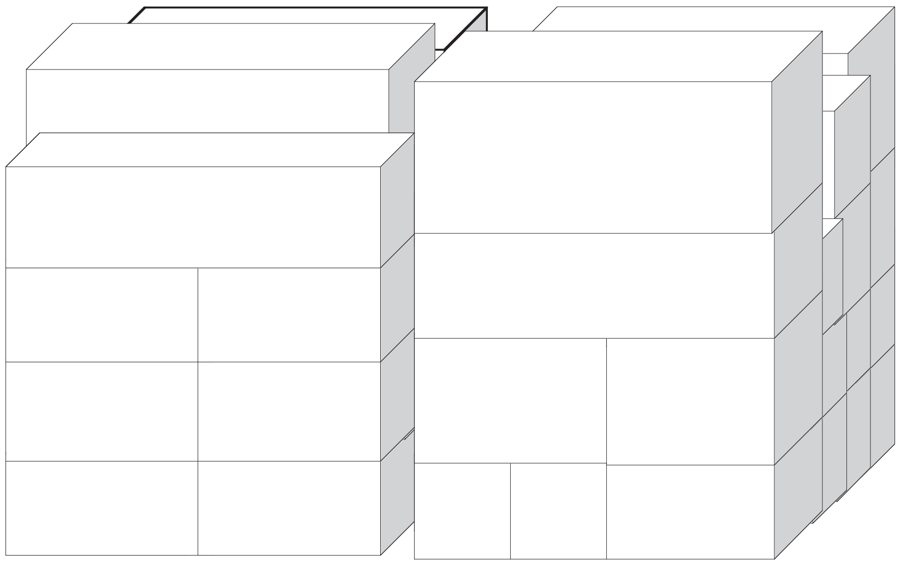

FIG. 4. Final 3D representation of the same MBP.

of the solutions obtained. Here we present two different computational experiments regarding:

(1) the comparison between the solutions of 18 instances related to a 198 TEU containership obtained with the present approach and the optimal solutions of the same instances solved with the 0–1 Linear Programming model for MBPP reported in Ambrosino *et al.* (2004);
(2) the analysis of the results of large scale instances related to a 1800 TEU containership, located in a maritime terminal in Genoa, and the comparison with the operative reality.

5.1 *First computational experiments*

The first computational experiments refer to the test instances reported in Table 4 about the stowage of the Chiwaua ship, that is a 198 TEU containership, with 11 bays, four rows and five tiers (three in the hold and two in the upper deck, respectively). Such instances differ from each other for the number of containers to be loaded, ranging from 100 to 163, their size (10′, 20′, 40′, 20′HC and 40′HC) and weight (light, medium and heavy), the number of ports to be visited, that is either 2 or 3, and the number of TEUs to load on board, ranging from 138 to 188. Column *Full* gives the ship occupation level, as percentages, when all containers are loaded; note that we give a 100% occupation level when 188 TEUs are loaded, since, conventionally, 10 TEU locations are operatively always let free for security and possible emergency reasons.

Together with the ship profile, which contains the information related to both structural and operational constraints of the ship, the terminal has the bay plan configuration that is

TABLE 4 *Input data of 18 instances of the Chiwaua containership*

Instance	Container characteristics													% Full
	TEU	n	Size					Weight			Destination			
			10′	20′	40′	20HC	40HC	l	m	h	1	2	3	
1	138	100	0	62	38	0	0	46	50	4	47	53	0	73·40
2	165	120	0	75	45	0	0	52	64	4	55	65	0	87·77
3	165	123	6	70	40	2	5	55	64	4	57	66	0	87·77
4	170	130	0	90	40	0	0	60	66	4	62	68	0	90·43
5	175	130	0	85	45	0	0	58	68	4	62	68	0	93·09
6	175	132	4	80	38	3	7	60	68	4	64	68	0	93·09
7	180	140	0	100	40	0	0	62	74	4	61	79	0	95·74
8	180	150	0	120	30	0	0	70	76	4	65	85	0	95·74
9	185	130	0	75	55	0	0	60	66	4	62	68	0	98·40
10	185	140	0	95	45	0	0	58	78	4	65	75	0	98·40
11	185	140	0	95	45	0	0	62	73	5	50	40	50	98·40
12	185	143	6	90	43	2	2	65	73	5	50	40	53	98·40
13	186	135	0	84	51	0	0	59	72	4	60	75	0	98·94
14	186	139	8	75	45	5	6	63	72	4	63	76	0	98·94
15	188	148	0	108	40	0	0	66	78	4	71	77	0	100·00
16	188	148	0	108	40	0	0	68	76	4	50	50	48	100·00
17	188	158	0	128	30	0	0	72	82	4	75	83	0	100·00
18	188	163	10	115	26	8	4	77	82	4	77	86	0	100·00

useful for establishing the stowage in the available locations of the ship and understanding its shape. Looking at these documents we define the partition of the structure of the Chiwaua ship and split it into different sections, that are used for stowing the containers. As we have already said, this phase is really important for the quality and feasibility of the final master bay plan. First, we identify the main section by searching for the largest parallelepiped area; then, we split the main section into $k = 8$ normal sections and find the origin of each normal section on the basis of stability and weight considerations according to the alternate criterion given in Section 4.3 (see Fig. 5). The other $q = 10$ special sections result from the remaining spaces of the ship and consist of the lowest tiers and lateral sea-side bays. Note that in such special sections it is difficult to put containers; we put there 40′ containers, thus reducing the number of placements and consequently the loading time.

Table 5 reports the results of the instances of the Chiwaua ship obtained by using our heuristic and gives a comparison between the optimal solutions obtained with the 0–1 Linear Programming model for MBPP (column 0–1), reported in Ambrosino *et al.* (2004), and the solutions obtained by using the present algorithm (column OUR). All computational experiments have been performed on a PC Pentium II of 350 Mhz.

It is interesting to analyse the difference between the optimal loading times (column 0–1) and those corresponding to the stowage plans obtained with our heuristic algorithm (column OUR), that is, on average, 34 min, corresponding to an average optimality gap of 10·75%, with a maximum value of 39 min and a minimum value of 28 min.

With reference to the computational times, it can be easily seen that, in our heuristic procedure, they are almost irrelevant; in fact in all instances they are always less than 1

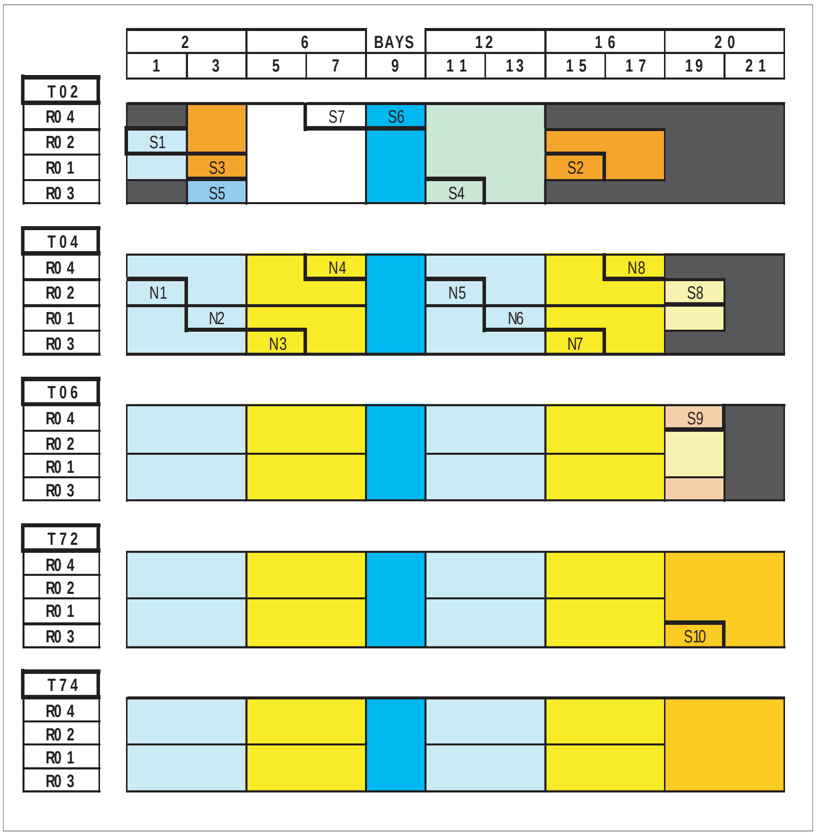

FIG. 5. Normal and special sections of the Chiwaua ship.

second. Instead, we can note how, in the case of the exact Linear Programming model, the CPU time grows noticeably with the number of containers loaded, because of the NP-hard nature of MBPP. It is worth mentioning that the planning office of the terminal that has provided us the data of the Chiwaua containership takes from about one hour to one and a half hours to manually compile the corresponding master bay plans. Operatively, the time required to compile stowage plans increases with respect to the number of containers to be stowed; therefore, the faster the solution time of any supporting software program the lower is the berthing time (which is an important performance index of any maritime terminal) required for loading containers on board.

5.2 *Second computational experiments*

In this section we present a case study which motivates our analysis about MBPP. In particular, we consider the Europe containership, which is a 'client' of the maritime terminal in Genoa (Italy). It is a 1800 TEU containership, i.e. 10 times bigger than the Chiwaua ship before analysed, with a more complex shape and structure.

TABLE 5 *Comparison of the loading times and the CPU times*

Instance	TEU	*n*	% Full	Loading time (min)				Computatonal time (min)		
				0–1	OUR	Δ	Δ%	0–1	OUR	Δ%
1	138	100	73·40	231·90	263·67	31·77	13·70%	2·85	0·009	99·7%
2	165	120	87·77	278·70	311·83	33·13	11·89%	8·28	0·012	99·9%
3	165	123	87·77	286·10	317·90	31·80	11·11%	8·58	0·012	99·9%
4	170	130	90·43	302·60	335·00	32·40	10·71%	10·32	0·011	99·9%
5	175	130	93·09	301·00	333·33	32·33	10·74%	5·76	0·009	99·8%
6	175	132	93·09	306·70	338·65	31·95	10·42%	6·21	0·013	99·8%
7	180	140	95·74	324·40	358·50	34·10	10·51%	10·08	0·012	99·9%
8	180	150	95·74	348·20	383·33	35·13	10·09%	14·27	0·009	99·9%
9	185	130	98·40	302·10	330·45	28·35	9·38%	11·02	0·011	99·9%
10	185	140	98·40	324·90	359·00	34·10	10·50%	15·38	0·010	99·9%
11	185	140	98·40	324·60	358·83	34·23	10·55%	14·53	0·012	99·9%
12	185	143	98·40	332·40	365·78	33·38	10·04%	15·20	0·012	99·9%
13	186	135	98·94	312·00	346·50	34·50	11·06%	8·56	0·013	99·8%
14	186	139	98·94	322·34	355·40	33·06	10·26%	5·52	0·016	99·7%
15	188	148	100·00	342·40	379·17	36·77	10·74%	13·32	0·012	99·9%
16	188	148	100·00	342·30	379·67	37·37	10·92%	13·17	0·009	99·9%
17	188	158	100·00	366·80	405·33	38·53	10·50%	23·88	0·013	99·9%
18	188	163	100·00	378·20	417·21	39·01	10·31%	25·82	0·012	100·0%
Average	178·44	137·17	94·92	318·20	352·20	34·00	10·75%	11·82	0·01	99·87%

All structural and operational information about the Europe ship are available to interested readers.

In the first computational experiments on the Europe ship we test our approach looking for the master bay plan referring to 15 cases, reported in Table 6; the instances differ from each other for the number of containers to load on board, ranging from 715 to 1413, corresponding to a ship occupation level, as percentages, ranging from 52·50 to 100%. We increase the number of heavy containers, until 50% of the total number of containers are loaded (see instance 13), and change the size and the number of ports to be visited, which is either 2 or 3. Note that there are only 20′ and 40′ dry containers, which in real cases represent about 98% of the total TEUs loaded on board.

The results of the above instances, reported in Table 7, show that, as in the Chiwaua ship, the computational times are not influenced by the ship dimensions.

Within the second group of computational experiments, we have used specific professional software developed for the Europe ship by the terminal for checking the stability, equilibrium, draft, inclinations and torsion of the ship with the stowage plans obtained with our heuristic approach for the instances reported in Table 8. Starting from these input data we compare the values of some quality indicators between the stowage plans obtained by our algorithm and those that have been established by the planning office of the terminal that has provided us with the data of the containers to be loaded. Note that the exact Linear Programming model used in the first computational experiments is impractical for these larger instances.

TABLE 6 *Input data of 15 instances of the Europe ship*

Instance	Containers characteristics															% Full
	TEU	n	Size				Weight			Destination (TEU)						
			$20'$	%	$40'$	%	l	m	h	1	%	2	%	3	%	
1	945	715	485	68	230	32	215	429	71	463	49	482	51	–	–	52·50
2	1022	762	502	66	260	34	228	458	76	501	49	521	51	–	–	56·78
3	1120	820	520	63	300	37	246	492	82	549	49	571	51	–	–	62·22
4	1218	898	578	64	320	36	270	541	87	600	49	618	51	–	–	67·67
5	1320	980	640	65	340	35	295	589	96	450	34	514	39	356	27	73·33
6	1380	1090	800	73	290	27	327	654	109	676	49	704	51	–	–	76·67
7	1386	984	582	59	402	41	296	590	98	680	49	706	51	–	–	77·00
8	1415	1069	723	68	346	32	321	642	106	481	34	524	37	410	29	78·61
9	1420	1060	700	66	360	34	318	636	106	696	49	724	51	–	–	78·89
10	1528	1202	876	73	326	27	361	721	120	749	49	779	51	–	–	84·89
11	1522	1138	754	66	384	34	341	683	114	524	34	586	39	412	27	84·56
12	1627	1215	803	66	412	34	365	729	121	804	49	823	51	–	–	90·39
13	1691	1289	887	69	402	31	172	512	605	835	49	856	51	–	–	93·94
14	1724	1331	938	70	393	30	400	799	132	588	34	669	39	467	27	95·78
15	1800	1413	1026	73	387	27	424	848	141	882	49	918	51	–	–	100·00

TABLE 7 *Results of the above instances*

Instance	TEU	n	Total loading time (min)	Containers handled per hour
1	945	715	2979·67	28·80
2	1022	762	3149·33	29·03
3	1120	820	3339·83	29·46
4	1218	898	3600·67	29·93
5	1320	980	3861·50	30·45
6	1380	1090	4250·83	30·77
7	1386	984	3864·00	30·56
8	1415	1069	4153·00	30·89
9	1420	1060	4110·83	30·94
10	1528	1202	4595·67	31·39
11	1522	1138	4371·00	31·24
12	1627	1215	4595·33	31·73
13	1691	1289	4839·00	31·97
14	1724	1331	4994·83	31·98
15	1800	1413	5245·33	32·33

For an exact comparison of the results we recall the following assumptions that have been taken in our resolution approach:

- we consider only three classes of weight (l, m, h);
- we have at most three destination ports;
- we have only ISO standard dry containers to be loaded (see Table 1);

TABLE 8 *Input data of two real instances*

Case	Containers characteristics															
	TEU	n	Size				Weight						Port (TEU)		% Full
			20′	%	40′	%	l	%	m	%	h	%	1	2	
1	1442	1070	698	65·23	372	34·77	247	23	446	42	377	35	690	752	80·11
2	1665	1233	801	64·96	432	35·04	209	17	290	24	734	60	841	823	92·50

TABLE 9 *Comparison between some feasibility indicators for the Europe ship*

Case 1

MBPP Solutions	G°M	Draught FWD	Draught AFT	Draught Mean	Propeller water line (submersion)	Rolling time	Stability range
Terminal	2·62 m	6·93 m	7·30 m	7·12 m	56%	15·6 sec	61·0°
Our	4·15 m	5·90 m	8·46 m	7·18 m	73%	12·3 sec	70·0°

Case 2

MBPP Solutions	G°M	Draught FWD	Draught AFT	Draught Mean	Propeller water line (submersion)	Rolling time	Stability range
Terminal	1·74 m	8·75 m	8·78 m	8·76 m	77%	18·1 sec	56·0°
Our	2·06 m	8·05 m	9·26 m	8·63 m	84%	16·7 sec	58·0°

- we assume that the ship starts its journey in the port for which we are studying the problem.

The values of the indicators in the official stowage plans produced by the planning office and those obtained by the proposed approach are reported in Table 9.

The first index ‘G° m’ refers to the stability of the ship and represents the height of the halfcentre; in general it is necessary that this index assumes positive value, with a minimum value of 0·15 m, meaning that the ship is perfectly balanced. The *Draught* indicator shows the inclinations of the ship with its three values *Fwd*, *Aft*, *Mean* regarding respectively the bow, the stern and the central side; in good stowage plans these values have to be almost the same. The ‘Propeller submersion’ is usually greater than 100%, but the above values are influenced by the fact that we did not consider the weight of the ballast, the fuel and the other equipment provided. The *Rolling time* index gives the oscillation of the ship during navigation in normal weather conditions. Finally, the *Stability range* is used for checking the positioning of the cargo and has to be greater than 50°.

Note that all the above values for the stowage plans obtained by our solution method are positive and feasible thus confirming the goodness of the results.

As a further consideration we have computed the total loading time for the above cases showing the very good performance of our algorithm in the minimization of the loading time (see Table 10).

6. Concluding remarks

In this paper we have presented a heuristic algorithm for MBPP based on its connection to 3D-BPP. The proposed solution method, although it does not reach optimal values, has very

TABLE 10 *Comparison between loading times*

Case	Loading time (min)		Δ%
	Our	Planning Office	
1	4169·3	4199·5	−0·72%
2	4648·3	4754·8	−2·24%

good performances in terms of both solution quality and computational time. In particular, the most important consideration about the performance of our heuristic algorithm is the possibility of finding stowage plans for big size ships and more than 1400 containers to be loaded. Therefore, we believe that the proposed approach is very valuable and that one of the future directions of this research should be its application on the evaluation of the impact of the ship system requirements in the whole organization of the yard.

Aknowledgements

The authors wish to thank the anonymous referees for their fruitful comments and suggestions.

REFERENCES

AMBROSINO, D. & SCIOMACHEN, A. (1998) A constraints satisfaction approach for master Bay plans. *Maritime Engineering and Ports*. (G. Sciutto & C. A. Brebbia, eds). Boston: WIT Press, pp. 155–164.

AMBROSINO, D., SCIOMACHEN, A. & TANFANI, E. (2002) A Bay assignment procedure for the loading problem. *Proc. 3th Int. Congress on Maritime Technological Innovations andResearch*. Bilbao, pp. 7–15.

AMBROSINO, D., SCIOMACHEN, A. & TANFANI, E. (2004) Stowing a containership: the master Bay plan problem. *Transport. Res.*, **38**, 81–99.

AVRIEL, M. & PENN, M. (1993) Exact and approximate solutions of the container ship stowage problem. *Comput. Indust. Engng*, **25**, 271–274.

AVRIEL, M., PENN, M. & SHPIRER, N. (2000) Container ship stowage problem: complexity and connection to the colouring of circle graphs. *Discr. Appl. Math.*, **103**, 271–279.

ATKINS, W. H. (1991) *Modern Marine Terminal Operations and Management*. Oakland, CA: Boyle.

BISCHOFF, E. E. & MARIOTT, M. D. (1990) A comparative evaluation of heuristics for container loading. *Eur. J. Oper. Res.*, **44**, 267–276.

BISCHOFF, E. E. & RATCLIFF, M. S. W. (1995) Issues on the development of approaches to container loading. *Int. J. Manage. Sci.*, **23**, 377–390.

BORTFELDT, A. & GEHRING, H. (2001) A hybrid genetic algorithm for the container loading problem. *Eur. J. Oper. Res.*, **131**, 143–161.

BOTTER, R. C. & BRINATI, M. A. (1992) Stowage container planning: a model for getting an optimal solution. *IFIP Trans. B (Appl. Tech.)*, **B-5**, 217–229.

CHEN, C. S., LEE, S. M. & SHEN, Q. S. (1995) An analytical model for the container loading problem. *Eur. J. Oper. Res.*, **80**, 68–76.

CRAINIC, G., GENDREAU, M. & DEJAX, P. (1993) Dynamic and stochastic models for the allocation of empty containers. *Oper. Res.*, **41**, 102–126.

DAVIES, A. P. & BISCHOFF, E. E. (1999) Weight distribution considerations in container loading. *Eur. J. Oper. Res.*, **114**, 509–527.

FAINA, L. (2000) A global optimization algorithm for the three dimensional packing problem. *Eur. J. Oper. Res.*, **126**, 340–354.

GEHRING, M. & MENSCHER E MEYER, M. A. (1990) Computer-based heuristic for packing pooled shipment containers. *Eur. J. Oper. Res.*, **44**, 277–288.

GEHRING, M. & BORTFELDT, A. (1997) A genetic algorithm for solving container loading problem. *Int. Trans. Oper. Res.*, **4**, 401–418.

GEORGE, J. A. & ROBINSON, D. F. (1980) A heuristic for packing boxes into a container. *Comput. Oper. Res.*, **7**, 147–156.

IMAI, A., NISHIMURA, E., PAPADIMITRIU, S. & SASAKI, K. (2002) The containership loading problem. *Int. J. Marit. Econ.*, **4**, 126–148.

MARTELLO, S., PISINGER, D. & VIGO, D. (2000) The three-dimensional bin packing problem. *Oper. Res.*, **48**, 256–267.

MOHANTY, B. B., MATHUR, K. & IVANCIC, N. J. (1994) Value considerations in three dimensional packing. A heuristic procedure using the fractional knapsack problem. *Eur. J. Oper. Res.*, **74**, 143–151.

PISINGER, D. (1998) A tree search heuristic for the container loading problem. *Ricerca Operativa*, **28**, 31–47.

RAIDL, G. R. (1999) A weight-coded genetic algorithm for the multiple container packing problem. *Proceedings of the 1999 ACM Symp. on Applied Computing*. San Antonio, TX, pp. 291–296.

TERNO, J., SCHEITHAUER, G., SOMMERWEISS, U. & RIEHME, J. (2000) An efficient approach for multi-pallet loading problem. *Eur. J. Oper. Res.*, **123**, 372–381.

THOMAS, B. J. (1989) *Management of Port Maintenance: a Review of Current Problems and Practices*. London: HMSO.

WILSON, I. D. & ROACH, P. (2000) Container stowage planning: a methodology for generating computerised solutions. *J. Oper. Res. Soc.*, **51**, 248–1255.