THE SHIP STOWAGE PLANNING PROBLEM

M. F. Monaco[(a)], M. Sammarra[(b)]

[(a)]Dipartimento di Elettronica, Informatica e Sistemistica – Università della Calabria – Via P. Bucci 41-C, 87036 Rende (CS) – Italy
[(b)] Dipartimento di Elettronica, Informatica e Sistemistica – Università della Calabria – Via P. Bucci 41-C, 87036 Rende (CS) – Italy

[(a)]monaco@deis.unical.it, [(b)]m.sammarra@deis.unical.it

ABSTRACT
In this work we deal with the Ship Stowage Planning Problem. It is a two-step process. The first step is executed by the shipping line, which designs the stowage plan for all ports of a vessel's rotation. At a given port, the stowage plan indicates containers that must be discharged and, for each available slot, a class of containers which can be loaded there. Here we focus on the second step, which pertains to the terminal planners. On the basis of the stowage plan, they must assign exactly one container, of a given class, selected in the yard, to a slot of the same class. The objective is the minimization of the total loading time. We present a Linear Integer Model and a heuristic algorithm, to get feasible solutions for the problem.

Keywords: maritime logistics, container stowing, heuristic

1. INTRODUCTION

Nowadays the world container traffic calls for very large containerships, whose capacity is around ten thousand of TEUs. Moreover each of them visits several hubs, where some containers are discharged and some are loaded. Therefore, during a trip, the *configuration* of a containership's cargo can vary from a port to the next one the ship has to visit (this sequence of ports is called *port rotation*). In this high dynamic scenario, planning the position of containers transported by a containership is a key factor to guarantee a fast turnaround time.

Before discussing how the stowage plan is drawn up, it is instructive to describe, briefly, the structure of a modern containership. It consists, basically, of several *slots* capable of accommodating one TEU. Two adjacent slots are used to stow a 40' container.

Slots belonging to the same cross section of the containership constitute a *bay*, those sharing the same vertical section a *row*, and, finally, those which are on the same horizontal section a *tier* (see Figure 1). Thus, to identify the position of container within a vessel, a three index code (b,r,t) is used. The first index b represents the bay and takes positive numbers, usually increasing from the bow to the stern of the containership. To differentiate 20' containers from 40' containers, odd bay numbers and even bay numbers, respectively, are used. The second index, the row index r, is a positive odd number for rows on the quay-side of the vessel, while rows on the sea-side are indexed by even numbers (the row $r = 0$ is the central one). Finally the tier index t takes even values from 02 (indicating the lowest position) to the maximum allowed for slots located below the deck (hold), while positions on the deck are labelled by indexes starting from 80.

Designing a stowage plan for a containership consists of two sequential processes (Steenken, Voss, and Stalhlbock 2004; Álvarez, 2006). The first step is executed by the shipping line, whose planners have a complete view both of all containers that have to be loaded or discharged during the vessel's trip, and of the cellular structure of the vessel. They provide stowage plans for each port of the vessel's port rotation.

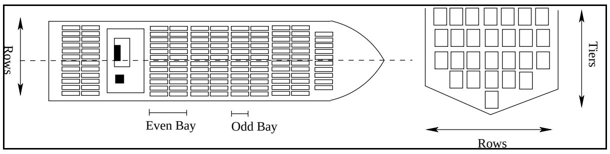

Figure 1: Horizontal and Cross sections of a vessel

At this stage of the stowage planning process, the reference unit is not the specific container, but a class of containers, identified by several attributes: the size, the type, the destination port, the weight. Containers of the different classes are assigned to specific positions within the ship, taking into account a number of constraints, basically related to the ship's capacity and stability. The shipping line's objectives are to minimize the number of on-board shifts during the port rotation and to maximize the ship utilization. The result of this process is a list of documents, which are sent to the terminals' planners of all ports the containership will visit.

A first document summarizes all container moves (loading, discharging, and restowing) that have to be performed at a specific port. Then a series of other sheets gives a detailed view of each ship bay and indicates positions from which containers have to be discharged (*Discharging Plan*) or in which containers have to be loaded (*Loading Plan*). Figure 2.a shows that at the port labelled by "2" 32 containers must be discharged from bay 1. More precisely 16 containers are locate below into the hold and 16 are up the deck. Furthermore Figure 2.a shows that to reach containers into the hold a *hatchcover* (represented in the figure by a thick black line) must be beforehand removed.

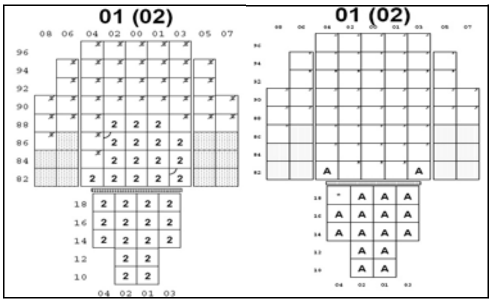

Figure 2: Discharging (a) and Loading (b) Plans.

The *Loading Plan* for bay 1 is depicted in Figure 2.b, from which one can retrieve that at bay 1 15 containers must be stowed into the hold and two must be placed on the deck. All containers stowed into the bay 1 belong to the same class. In particular these containers will be discharged, successively, at the port labelled by "A". As regards this issue, it is worthy to highlight that the *class* of a container incorporates several container characteristics: the port of destination, of course, but also the weight class (*heavy, medium, light*), the dimension (*standard, high cube, oversized*), the type of load (*perishable, dangerous*).

These documents act as guidelines (*Prestow Plans*) for the terminals' planners, which are involved in the second step of the stowage plan design, sometimes referred as *ship loading* or *load sequencing* problems. In the reported example, planners of port "2" must assign exactly one container (identified by a unique code) of class "A", selected in the yard, to each slot of bay 1. In this phase the basic constraint planners have to comply with is that the weight of container stacks must decrease from the hold to the deck.

As regards the objectives the terminal planners have to pursue, they are quite different from the shipping line's ones. In order to speed up the ship loading process, it is possible to minimize the transportation time of containers from the yard to the quay, or minimize the yard reshuffles. Reshuffles, or yard shifts, are very time consuming unproductive moves, which occur whenever some containers on the top of a yard stack have to be removed (and then restacked) in order to pick up a suitable container on the bottom of the same stack.

The paper will address the ship stowage planning problem as it arises at a transhipment container terminal. The strong interaction between the problem under consideration and the yard layout and equipment is evident. We will consider an extensive yard, where containers are moved by a fleet of straddle carriers (*Direct Transfer System - DTS*). In such a context, yard stacks are almost three/four containers height.

Looking at the problem from the point of view of the terminal manager, the objective to minimize will be the ship's total berthing time. The contribution to the ship's berthing time related to the loading process is given by two terms: the total transportation time of containers from yard to quay, including the time wasted by reshuffles, and the total loading time. Since the shipping line's prestow plan is an input datum, as well as the number of cranes allocated to the containership and the sequence of ship bays each of them has to handle, the time the cranes will take to load all the containers into the ship can be considered a constant. It is worth noting that the stowage planning is an offline optimization process, in the sense that the stowage plan is generated by the ship planner before ship loading starts. Hence, the main modelling difficulty, in the case of many cranes working on the same vessel, relies on the reshuffles estimation, since the yard stack configurations vary dynamically during the loading process.

Here we will reduce our attention to the minimization of the transportation times and of yard reshuffles, by formulating the stowage planning problem as an integer linear programming problem. The outline of the paper is as follows. In Section 2 we provide an analysis of related literature. A Linear Integer Model for the stowage problem is presented in Section 3. Section 4 describes a heuristic algorithm and is followed by preliminary results in Section 5. Some conclusions are driven in Section 6.

2. LITERATURE REVIEW

The Ship Stowage Planning Problem (SSPP in what follows) has received, to our knowledge, not much attention by researchers. Moreover in some of previous studies the distinction between the two phases is not so clear, which makes difficult the application in a real context of models and algorithms developed.

(Avriel, Penn, Shpirer, et al. 1998) have studied the SSPP from the point of view of the shipping companies. The authors propose an Integer Linear Model, where the objective is the minimization of on-board shifts, and a heuristic algorithm to get feasible solutions to the problem.

In the paper of (Wilson and Roach 1999) the whole stowage planning problem is considered. First, the authors propose a Branch-and-Bound algorithm for the *Prestow Planning*. Then the second phase is performed by a Tabu Search algorithm. Similar studies are reported in (Wilson and Roach 2000; Wilson, Roach, and Ware 2001).

(Dubrovsky, Levitin, and Penn 2002) propose a genetic algorithm for the problem as described in (Avriel, Penn, Shpirer, et al. 1998), but taking into account constraints on the ship stability.

In (Ambrosino, Sciomachen, and Tanfani, 2004) the ship planning problem at the first port of the port rotation is described. The authors propose a Mixed Integer Program and a three-step heuristic, with the aim of minimizing the total loading time. A different algorithm for the same problem is given in (Ambrosino, Sciomachen, and Tanfani 2006).

In our opinion the most meaningful papers on the SSPP are those of (Álvarez, 2006) and (Kim, Kan, and Ryu, 2004).

In (Álvarez, 2006) the author considers a port where containers are moved by reachstackers and analyzes the problem from the point of view of the terminal managers. The author provides an Integer Program, whose objective function to minimize is a linear combination of the yard reshuffles and of the total distance travelled by the reachstackers. The problem is solved via a Tabu Search algorithm.

The paper by (Kim, Kan, and Ryu, 2004) considers a port where an *Indirect Transfer System* is adopted: a fleet of trucks moves containers between the yard and the quay, while stacking and retrieving of containers in the yard is performed by transfer cranes. The authors present a Nonlinear Integer Model for the load-sequencing problem, which consists in determining, for each transfer crane, a pick-up sequence of containers and, then, for each container, a loading sequence. The objective function, which is a combination of several terms, relates to the maximization of the efficiency of both transfer and quay cranes. The solution of the problem is tackled by a Beam Search algorithm.

3. MATHEMATICAL MODEL

To derive a Linear Integer Model for the SSPP we consider, as input data, the following:

1. the configuration of the ship;
2. the prestow plans, assuming that constraints related to the ship capacity, the cross and longitudinal equilibrium, are satisfied;
3. the number of cranes allocated to the ship, as well as the sequence of bays each crane will handle and, for each bay, the *operational modality* of the crane (sea-to-shore, shore-to-sea, row-wise, stack-wise);
4. the set of all containers whose classes match the classes of ship's lots to be filled, together with their codes, attributes, and positions in the yard;

Thanks to assumptions (1) and (3), it is possible to map each slot (*b,r,t*) in to a sequence position index p^k, if the slot (*b,r,t*) is the *p-th* container in the sequence of slots where the crane *k* has to load containers. Since for each slot the prestow plan indicates the class of container which can be stowed in that slot, the problem reduces to assign a suitable container to each position index p^k subject to side constraints. These further constraints have to guarantee that the weight of containers piled up on the same stack must decrease from the bottom to the top.

We now introduce our main notation. We define:

- N the set of containers as previously defined $(|N| = n_C)$;
- $T \subseteq N$ and $F \subseteq N$ the set of 20' and 40' containers, respectively;
- $\forall i \in N, w_i$ the weight of container i, and c_i its class;
- K the set of cranes assigned to handle a given containership $(|K| = m)$;
- $\forall k \in K, P^k$ the set of slot positions associated to the crane k $\left(|P^k| = n^k \text{ and } \sum_{k \in K} n^k = n_P\right)$;
- $\forall k \in K, P_E^k \subseteq P^k$, the set of slot positions where the crane k can load a 40' container;
- $\forall k \in K, P_O^k \subseteq P^k$, the set of slot positions where the crane k can load a 20'container;
- $\forall k \in K, \forall p \in P^k$, c_p the class of the slot related to the position p;
- $\forall i \in N, k \in K, p \in P^K$, τ_{ip}^k the time needed to transport the container from its location in the yard to the bay corresponding to the position p of crane *k*;
- σ, the time needed to perform the shift of a container in the yard.

Moreover we introduce the following $\{0,1\}$ matrices:

- $\forall k \in K, \Phi^k_{n^k \times n^k}$ whose elements $\varphi_{pq}^k = 1$ if and only if the slots associated to position indexes p and q, (b_p, r_p, t_p) and (b_q, r_q, t_q) respectively, satisfy $b_p = b_q, r_p = r_q, t_p < t_q$;
- $\Gamma_{n_C \times n_C}$, which relates the positions of containers in to the yard. In particular the generic element $\gamma_{ij} = 1$ indicates that containers i and j are in the same yard stack and i lies below j.

By means of the decisional variables

- $x_{ip}^k = 1$ if and only if the container i is assigned to the position p of the crane k;
- $z_{ij} = 1$ if and only if container i is handled before j in a sequence P^k for some $k \in K$

the SSPP can be formulated as follows:

$$\min \sum_{i\in N}\sum_{k\in K}\sum_{p\in P^k} \tau_{ip}^k x_{ip}^k + \sigma \sum_{i\in N}\sum_{j\in N} \gamma_{ij} z_{ij} \quad (1)$$

s.t.

$$\sum_{k\in K}\sum_{p\in P^k} x_{ip}^k \le 1 \qquad \forall i \in N \quad (2)$$

$$\sum_{i\in N} x_{ip}^k = 1 \qquad \forall k \in K, p \in P^k \quad (3)$$

$$x_{ip}^k = 0 \qquad \forall i \in T, k \in K, p \in P_E^k \quad (4)$$

$$x_{ip}^k = 0 \qquad \forall i \in F, k \in K, p \in P_O^k \quad (5)$$

$$x_{ip}^k = 0 \qquad \forall i \in N, k \in K, p \in P^k, c_i \ne c_p \quad (6)$$

$$\sum_{i\in N} w_i x_{ip}^k - \phi_{pq}^k \sum_{j\in N} w_j x_{jq}^k \ge 0 \quad \forall k \in K, p,q \in P^k \quad (7)$$

$$z_{ij} \ge x_{ip}^k + \sum_{\substack{q\in P^k \\ q>p}} x_{jq}^k - 1 \qquad \forall i,j \in N, k \in K, p \in P^k, \gamma_{ij} = 1, p \ne n^k \quad (8)$$

$$x_{ip}^k \in \{0,1\} \qquad \forall i \in N, k \in K, p \in P^k \quad (9)$$

$$z_{ij} \in \{0,1\} \qquad \forall i,j \in N \quad (10)$$

In this model the objective function (1) consists of the containers' transportation time from the yard to the quay plus the time needed to perform possible yard shifts and represents the *variable* share of the total handling time. The latter includes also the containers' loading time which, as discussed in the Introduction, can be considered constant.

Constraints (2) and (3) are the classical assignment constraints (note that we are assuming $n_C \ge n_P$, which is a necessary condition for the model to be feasible), while variable settings (4) to (6) avoid to assign containers to incompatible positions. Decreasing weight of containers on the same stack is ensured by constraints (7). Finally constraints (8) define the z variables and count the yard shifts. As regards constraints (8), one could observe that they do not take into account possible shifts of containers belonging to the same yard stack but handled by different cranes. Nevertheless, although we do not consider these possible shifts, the value we get by the model is not so far from the optimal one. Actually, as previously disclosed, we are considering a *DTS* transshipment port. Since yard stacks are homogeneous with respect to the destination port and since the *crane split,* that is the allocation of quay cranes to the ship bays, is commonly done so that each crane handles whole groups of homogeneous containers, it is unusual that two containers belonging to the same yard stack are handled by different cranes.

4. A HEURISTIC ALGORITHM FOR THE SSPP

To get feasible solutions to the Ship Stowage Planning Problem, we devised a heuristic algorithm, based on two sequential phases: *Construction* and *Improvement*.

The first phase generates a starting feasible solution $(\bar{x}, \bar{z})$, while the second one attempts to improve this solution, exploring a suitable neighborhood system.

4.1. The construction phase

Adopting the same notation introduced in the previous Section, a starting solution $(\bar{x}, \bar{z})$ can be found as follows:

1. Let i be the heaviest container in N, breaking ties arbitrarily.
2. Examine the cranes and choose the first crane $k \in K$ whose positions P^k have not been filled yet. If there exists $p \in P^k$ such that $c_p = c_i$, set $\bar{x}_{ip}^k = 1$, $N = N \setminus \{i\}$, $P^k = P^k \setminus \{p\}$ and update $\bar{z}$, if necessary.
3. If $P^k = \varnothing \ \forall k \in K$ STOP, otherwise GOTO 1.

This simple routine assigns containers to the first suitable position of the first available crane. Since we process containers in decreasing order of their weights and since we stop when in all available positions a container has been stowed, it yields a feasible solution.

4.2. The Improvement phase.

To improve the starting solution $(\bar{x}, \bar{z})$ we use a local search algorithm, which explores a neighborhood $I(\bar{x}, \bar{z})$ of this solution. It is obtained performing swaps of container pairs either within the same crane sequence or between different sequences. We define these moves *swap* and *change*, respectively.

For each sequence P^k, the swap move takes two containers i and j, of the same type and class, currently assigned to positions p and q. If swapping positions of these containers does not violate weight constraints (7):

- $w_i x_{iq}^k - \phi_{qr} \sum_{l\in N} w_l x_{lr}^k \ge 0 \quad \forall r \in P^k$
- $w_j x_{jp}^k - \phi_{pr} \sum_{l\in N} w_l x_{lr}^k \ge 0 \quad \forall r \in P^k$

then assigning container i to position q and container j to position p leads to a new feasible solution.

The *change* rule, similar to the previous one, operates swaps of containers i and j belonging to the sequences P^k and P^h, respectively. In this case one has to check if

- $w_i x_{iq}^h - \phi_{qr} \sum_{l \in N} w_l x_{lr}^h \geq 0 \quad \forall r \in P^h$
- $w_j x_{jp}^k - \phi_{pr} \sum_{l \in N} w_l x_{lr}^k \geq 0 \quad \forall r \in P^k$

are satisfied.

5. PRELIMINARY RESULTS

To verify the correctness of the heuristic algorithm, we have generated three set of small test problems, whose characteristics are listed in Table 1.

Table 1 : Characteristics of the test instances

Set	$n_P = n_C$	Classes	m
I1	20	4	1
I2	20	4	2
I3	64	4	2

The transportation times of containers are uniformly distributed in the range $[1, 2]$ minutes. Assuming an average speed of straddle carriers equal to $4m/s$, the above choice corresponds to consider distances between containers' positions and the quay in the range $[240m, 480m]$, which turns out to be realistic in a *DTS* terminal. As regards the configuration of the yard stacks, we have considered almost three containers on each stack. The number of stacks together with the distribution of their heights in the yard are summarized, for each test set, in Table 2.

Table 2 : Yard stacks heights

Set	Total	1	2	3
I1	8	1	2	5
I2	8	1	2	5
I3	24	0	8	16

The heuristic algorithm has been coded in Java and run on a 2*GHz* PC equipped with 2*GB* of RAM. Results of the runs has been compared with the value of the optimal solution computed by a commercial solver (LINGO 10). This first computational experience has shown that the proposed heuristic is attractive, since the larger gap with respect to the optimal solution is about 3.5% in the worst case.

Undoubtedly, to face with instances of realistic size, the search mechanism of our algorithm must be improved. At the present a Tabu Search algorithm, embedded in a Lagrangean Relaxation scheme, is under investigation, in order to obtain tight lower bounds and good feasible solutions.

6. CONCLUSIONS

In this paper we have discussed the Ship Stowage Planning Problem as it arises at a container terminal. Starting from the analysis of a case-study, the Gioia Tauro Container Terminal in the southern Italy, we have devised a Linear Integer Model for the problem.

In our opinion, the main contribution of this work is related to the particular and seemingly conflicting features of the model: it is very realistic, but surprisingly simple, compared to similar models in the literature. This nice characteristic is obtained by a rational use of the information available to the terminal planners.

The design of an efficient solution algorithm is the object of the current work.

ACKNOWLEDGMENTS

This work was partially supported by Ministero dell'Università e della Ricerca and by Regione Calabria under the Research Project: *PROMIS-Distretto Tecnologico della Logistica e della Trasformazione*.

REFERENCES

Álvarez J., 2006. A heuristic for vessel planning in a reach stacker terminal. *Journal of Maritime Research*, 3, 3–16.

Ambrosino D., Sciomachen A. Tanfani E., 2004. Stowing a containership: the master bay plan problem. *Transportation Research - A*, 38, 81–99.

Ambrosino D., Sciomachen A., Tanfani E., 2006. A decomposition heuristics for the container ship stowage problem. *Journal of Heuristics,* 12, 211–233.

Avriel M., Penn M., Shpirer N., Witteboon S., 1998. Stowage planning for containerships to reduce the number of shifts. *Annals of Operations Research,* 76,55–71.

Dubrovsky O., Levitin G., Penn M., 2002. A genetic algorithm with a compact solution encoding for the container ship stowage problem. *Journal of Heuristics,*8, 585–599.

Kim K.H., Kang J.S., Ryu K.R., 2004. A beam search algorithm for the load sequencing of outbound containers in port container terminals. *OR Spectrum*, 26, 93-116.

Stahlbock R., Voss S., 2008. Operations research at container terminals: a literature update. *Operations Research Spectrum*, 30:1–52.

Steenken D., Voss S., Stahlbock R., 2004. Container terminal operation and operations research - a classification and literature review. *Operations Research Spectrum*, 26, 3–49.

Wilson I., Roach P., 1999. Principles of combinatorial optimization applied to container-ship stowage planning. *Journal of Heuristics*, 5, 403–418.

Wilson I., Roach P., 2000. Container stowage planning: A methodology for generating computerised solutions. *Journal of the Operational Research Society*, 51, 1248–1255.

Wilson I., Roach P.;Ware J., 2001. Container stowage pre-planning: using search to generate solutions, a case study. *Knowledge-Based Systems*, 14, 137–145.

AUTHORS BIOGRAPHY

M. Flavia Monaco is professor of Operations Research at the Engineering Faculty of the Università della Calabria. She is affiliate to the Dipartimento di Elettronica, Informatica e Sistemistica, where she supervises the research team on container terminal logistics.

Marcello Sammarra took his PhD in Operations Research in 2006 and currently is affiliate researcher at the Dipartimento di Elettronica, Informatica e Sistemistica of the Università della Calabria. He has been involved in several research projects on container terminal logistics.