Computer Applications in the Automation of Shipyard Operation
and Ship Design, VII – C. Baraúna Vieira et al. (Eds.)
Elsevier Science Publishers B.V. (North-Holland)

Stowage container planning: a model for getting an optimal solution.

R. C. Botter and M. A. Brinati

Department of Naval Architecture and Ocean Engineering, University of Sao Paulo, Caixa Postal 61548, CEP 01000, Sao Paulo, S.P., Brasil.

Abstract

In this paper a mathematical programming model for the container stowage problem is shown; the binary decision variables determine, for each port, the container unloading and loading sequence. In fact, the solution indicates successively which container will be handled, and from or to which cell in the ship.

Unless for some constraint linearizations (related to ship safety parameters), the proposed model finds, from the theoretical point of view, an optimal global solution for the stowage problem. Nevertheless, this combinatorial problem is NP-HARD and cannot be solved for commercial ship sizes in reasonable processing time using the available computer software and hardware.

The basic features of this model were used for the development of an implicit enumeration procedure for solving the container stowage problem. In spite of the computational complexity of this approach, some heuristic rules are proposed to explore the combinatorial tree in an intelligent way and produce good, if not optimal, solutions for the problem in a reasonable processing time.

1. INTRODUCTION

The technological advance in ship and loading equipment design and construction, with the advent of full cellular container ships, enabled a dramatic reduction of ship in-port time. Nevertheless, in order to take full advantage of this opportunity, the container stowage planning along the ports of the ship route must be carried out efficiently.

The stowage plan consists in the elaboration of the container unloading and loading sequence in each route port, so that no ship stability and stress constraints (such as the transverse metacentric height, trim, cutting force or bending moment) are violated, and container restows aboard the ship are avoided. In each port the unloading sequence must be such that only will be handled, if possible, containers whose final destination is that port. The container loading sequence must specify the position that each container will occupy on the ship, obeying the ventilation and refrigeration requirements and the dangerous cargo conditions of each container.

It must be emphasized that the stowage plan must also contemplate other aspects, the most important being the reduction of ballast required by the vessel and the reduction of the longitudinal crane movement.

The purpose of this paper is to show the development of a appropriate computational tools to help shipping companies in the elaboration of containership stowage plan. In section 2 a brief literature survey of the containership stowage problem is presented. In section 3 it is described an exact mathematical model for the elaboration of the stowage plan of a ship which will sail along a n-port route, and whose main objective is the minimization of the number of restows along that route. In this model, the interdependence of container handling decisions in different ports of the route is well chacterized. Due to the computational difficulties for solving this complete model, two alternative procedures for solving the container stowage problem are presented in sections 4 and 5. Computational experiments with these procedures are mentioned in section 6.

2. LITERATURE SURVEY

The search for an efficient procedure for container ship loading and unloading has drawn the attention of shipping companies and academic researchers since the seventies. The main aim of such a procedure is to minimize the number of overstows subject to a given set of constraints.

The methods used for solving the stowage planning problem may be grouped into the following main classes: probabilistic simulation methods, heuristics procedures, decision support systems, mathematical programming approaches and expert systems. None of them leads to an optimal solution for the problem.

The first group includes the classical Monte Carlo simulation method, which chooses, among several stowage plans generated for the route, the one which presents the smallest number of restows. The works of Nehrling [1] and Shields [2] belong to this group.

In the second class one finds the works of Webster [3], Beliech [4] and Thieu [5], which incorporate heuristic rules derived from the planners experience and are intended for the computerized generation of the stowage plans, without the interaction with the human being.

The methods of first and second groups do not seem to be efficient because they often generate stowage plans that result in a high number of restows along the ports of the route.

The methods of the third group attempt to provide the planner with capable tools so that, at any step of the stowage plan elaboration the ship safety parameters can be calculated. The works of Sha [6], Barauna Vieira [7] and Saginaw [8] belong to this class. This kind of procedure seems to be more efficient, when the planner who uses it as a decision-making support has a solid background in the subject and is able to explore efficiently a stowage plan proposal, obtaining in general responses with a low restows throughout the route.

Among the works belonging to the fourth group, one finds those of Scott [9] and Aslidis [10] and [11], whose models search the optimum solution of the stowage problem by means of a proper mathematical modelling; however, they rely on too many simplification hypotheses, which made them unsuitable for practical applications.

In the fifth class one finds the works of Dillingham [12], [13] and [14], that apply both the theory and softwares of artificial intelligence for the solution of the container stowage problem. The whole potential of this tool has not been explored yet but it is expected that it can generate efficient procedures in the near future.

In a global way, all the works found in the literature do not analyze the loading process in successive ports as interdependent decisions, but study it rather in a disassembled way. The lack of considering this interdependency is likely to be the major factor cause of container restows.

3. THE COMPLETE MATHEMATICAL MODEL FOR THE STOWAGE PROBLEM

In this section a mathematical programming model for the stowage problem is shown; the binary decision variables determine, for each port, the container unloading and loading sequence. In fact, the solution indicates successively which container will be handled, and from or to which cell in the ship.

Some hypotheses have been assumed:

- the ship attends n ports of a route, being empty as the loading starts in port 1, and at the end in port n, when all the containers aboard are unloaded;
- in each intermediate port of the route unloading and loading may occur, but the latter does not begin until the former has finished;
- the ship may carry containers of 20 and 40 feet of length, but they cannot be mixed in a same column;
- the ballast conditions of the vessel at the arrival and departure from each port of the route are set by the user;
- the container supply and demand along the ports of the route are known.

3.1. DEFINITIONS AND DECISION VARIABLES

The complete container loading and unloading process in the several ports of the route may be viewed as a succession of individual stages. Thus, for each possible container handling, either loading or unloading, it is defined a stage k. For suitability of mathematical modelling, all possible unloading and loading stages will be considered, i. e., the model assumes that at each port all the containers aboard the ship may be unloaded.

Let, then:

$I = \{1, \ldots NCONT\}$ be the set of containers supplied along the ports of the route;

I_r be the set of containers supplied at port r of the route;

R_r be the set containers supplied at all the ports of the route before port r and demanded at all the ports after port r;

D_r be the set of containers whose destination is port r;
$I(j)$ be the set of containers that can be placed in cell j of the ship;
O_i be the origin port of container i;
T_i be the destination port of container i;
$J = \{1, \ldots NCEL\}$ be the set of cells in the containership;
$J(i)$ be the set of cells where the container i can be placed;
$JCOL_z$ be the set of cells in the z column of containership, where z = 1 to $NCOL$ and $NCOL$ is the number of columns of the containership;
JSE_m be the set of cells between the bow and section m of containership, where m = 1 to $MSECTION$;
$K = \{1, \ldots NESTA\}$ be the set of the possible stages along the ports of the route;
$KC(i)$ be the set of all possible loading stages of the container i between the ports O_i and T_i;
$KD(i)$ be the set of all possible unloading stages of the container i between ports O_i and T_i;
$KDR(i)$ be the set of all possible unloading stages of the container i between ports O_i and T_i - 1. This set contains the stages for which the unloading of container i causes a restow;
$KPC(r)$ be the set of all possible loading stages in the port r;
$KPD(r)$ be the set of all possible unloading stages in the port r;
$KPCA(s, v)$ be is the sum of sets KPC(r) from port s to port v;
$KPDA(s, v)$ be is the sum of sets KPD(r) from port s to port v.

Let the decision variables for the stowage problem be:

$$X_{ijk} = \begin{cases} 1, \text{if the container i is loaded into cell j at the } k^{\underline{th}} \text{ stage;} \\ 0, \text{otherwise;} \end{cases}$$

for $i \in I$, $j \in J(i)$ and $k \in KC(i)$.

$$Y_{ijk} = \begin{cases} 1, \text{if the container i is removed from cell j in the } k^{\underline{th}} \text{ stage;} \\ 0, \text{otherwise;} \end{cases}$$

for $i \in I$, $j \in J(i)$ and $k \in KD(i)$.

3.2. OBJECTIVE FUNCTION

The objective function of the mathematical model for the stowage problem, besides considering the number of container restows along the route, takes into account also another important factor, which is the longitudinal crane movement along the quay during the container loading and unloading operations.

The general expression of the objective function is:

$$f = \lambda_1 * f_1 + \lambda_2 * f_2 \tag{1}$$

where λ_1 and λ_2 represent respectively the unit cost of a restow and the longitudinal movement of the crane.

The first term of the objective function, f_1, which counts the number of restows can be expressed as below:

$$f_1 = \sum_i \sum_j \sum_k Y_{ijk} \tag{2}$$

where in the summations above $i \in I$, $j \in J(i)$ and $k \in KDR(i)$.

The second term of the objective function, f_2, which refers to the longitudinal crane movement along the quay during the loading and unloading operations, is provided below:

$$f_2 = \sum_i \sum_j \sum_{i'} \sum_{j'} \sum_k X_{ijk} * X_{i'j'k+1} * d_{jj'} + \sum_i \sum_j \sum_{i'} \sum_{j'} \sum_{k'} Y_{ijk'} * Y_{i'j'k'+1} * d_{jj'} \tag{3}$$

where in the summations above:
$i \in I$; $j \in J(i)$; $i' \in I$; $j' \in J(i')$;
$k \in \{KC(i) \cap KC(i')\}$; $k' \in \{KD(i) \cap KD(i')\}$
and $d_{jj'}$ is the longitudinal distance between the pair of cells j and j'.

The function f_2 above is non-linear, but it can be linearized, for instance, according to what has been suggested by Taha [15], increasing however the number of decision variables and constraints.

3.3. CONSTRAINTS

The model considers the stability constraints, structural stress and other feasibility conditions presented below:

a-) Loading of the container i in the port p of origin

$$\sum_j \sum_k X_{ijk} = 1 \, , \quad i \in I_p \tag{4}$$

where in the summations above $j \in J(i)$ and $k \in KPC(p)$.

b-) Loading of the container i in any intermediate port p between the origin and destination ports

$$\sum_j \sum_k X_{ijk} - \sum_j \sum_{k'} Y_{ijk'} = 0 \, , \quad i \in R_p \tag{5}$$

where in the summations above:
$j \in J(i)$;
$k \in KPC(p)$ and $k' \in KPD(p)$.

c-) Unloading of the container i in the destination port p

$$\sum_k X_{ijk} - \sum_{k'} Y_{ijk'} = 0 \, , \quad i \in I \text{ and } j \in J(i) \tag{6}$$

where in the summations above:
$k \in KPCA(O_i, T_i - 1)$;
$k' \in KPDA(O_i + 1, T_i)$.

d-) Unloading of the container i in any intermediate port p between the origin and destination ports

$$\sum_k X_{ijk} - \sum_{k'} Y_{ijk'} \geq 0 \,, \quad i \in I \; and \; j \in J(i) \tag{7}$$

where in the summations above:
$k \in KPCA(O_i, T_i - 1)$;
$k' \in KPDA(O_i + 1, T_i - 1)$.

e-) Occupation of cells in the ship

These constraints guarantee that a cell can only be filled or emptied once at most in each port. For the loading phase at port p, the expression is:

$$\sum_{r=1}^{p} \sum_i \sum_k X_{ijk} - \sum_{r=1}^{p} \sum_{i'} \sum_{k'} Y_{i'jk'} \leq 1 \,, \; j \in J \tag{8}$$

where in the summations above:
$i \in \{I_r \cap (R_p \cup I_p)\}$ and $i \in I(j)$;
$i' \in \{I_r \cap (R_p \cup D_p)\}$ and $i' \in I(j)$;
$k \in KPCA(r, p)$ and $k' \in KPDA(r + 1, p)$.

Expression (8) applied to port 1 does not have the second group of summations, since in port 1 the ship is empty and only loading operations occur. For the unloading phase at port p, the expression for this constraint is similar to equation 8.

f-) Container handling at stage k

This constraint guarantees that in each stage k at most one cargo handling will occur.

$$\sum_i \sum_j X_{ijk} \leq 1 \;\;, \; k \in KCP(p) \tag{9}$$

where in the summations above:
$i \in \{R_p \cup I_p\}$;
$j \in J(i)$.

For port p, in any unloading stage, the expression for this constraint is similar to equation 9.

g-) Impenetrability

These constraints represent the following conditions:

- a container only can be loaded into a cell over an occupied cell or on the bottom or over a hatch cover. If loaded into the hold, the cover hatch must have been previously removed;
- a container can only be unloaded if the pile above it or the hatch cover (if there is one) has already been removed.

For every loading stage k and for every cell j of the ship moored in whichever port p, one has:

$$\sum_i X_{ijk} \leq \sum_i \sum_{k':k'<k} X_{ij_{sub}k'} + \sum_{r=1}^{p-1} \sum_{i'} \sum_{k''} X_{i'j_{sub}k''} - \sum_{r=1}^{p} \sum_{i'} \sum_{k'''} Y_{i'j_{sub}k'''}\ , \quad k \in KCP(p)\ and\ j \in J \tag{10}$$

where in the above summations:

$i \in \{R_p \cup I_p\}$ and $i \in I(j)$;
$i' \in \{I_r \cap R_p\}$; $k' \in KCP(p)$;
$k'' \in KPCA(r, p-1)$; $k''' \in KPDA(r, p)$

and j_{sub} is the cell below j. Equation 10 does not apply to a cell j on the bottom or over the hatch cover.

The constraints that represent the other conditions mentioned are similar to those presented in equation (10).

h-) Weight constraint for containership columns

The equation of this constraint that appears only in the loading phases of each port p is:

$$\sum_i \sum_j \sum_k X_{ijk} * p_i + \sum_{r=1}^{p-1} \sum_{i'} \sum_j \sum_{k'} X_{i'jk'} * p_{i'} - \sum_{r=1}^{p} \sum_{i'} \sum_j \sum_{k''} Y_{i'jk''} * p_{i'} \leq\ PEMAX_z\ ,\ z\ =\ 1\ to\ NCOL \tag{11}$$

where in the above summations:

$i \in \{R_p \cup I_p\}$ and $i \in I(j)$;
$j \in JCOL_z$; $k \in KCP(p)$;
$i' \in \{I_r \cap R_p\}$; $k' \in KPCA(r, p-1)$;
$k'' \in KPDA(r, p)$

and $PEMAX_z$ is the maximum permissible weight for column z of the ship and p_i is the weight of container i.

i-) Transverse metacentric height (GM) constraint

For a stage k during a loading phase in port p:

$$\sum_i \sum_j \sum_k X_{ijk} * p_i * z_j + \sum_{r=1}^{p-1} \sum_{i'} \sum_j \sum_{k'} X_{i'jk'} * p_{i'} * z_j - \sum_{r=1}^{p} \sum_{i'} \sum_j \sum_{k''} Y_{i'jk''} * p_{i'} * z_j + MZo\ \leq\ \Delta * (KB + BM - GM_{min} - DG)\ ,\ k \in KCP(p) \tag{12}$$

where in the above summations:

$i \in \{R_p \cup I_p\}$; $j \in J(i)$;
$i' \in \{I_r \cap R_p\}$; $k' \in KPCA(r, p-1)$;
$k'' \in KPDA(r, p)$

and KB and BM are respectively the vertical coordinate of the center of buoyancy in relation to the base line of the ship and the transverse metacentric radius (both depend on the displacement change after each weight loading into and unloading from the ship); GM_{min} is a known condition given by the operation handbook of the ship; DG is the component related to the free surface effect of the tanks (during the container loading and unloading process, one may assume that there will be no alteration of the tank conditions and therefore DG is known); Δ is the ship displacement after the cargo addition; z_j is the vertical coordinate of the center of cell j in relation to the base line of the ship (it is assumed that the vertical center of gravity of the container lies in the middle of its height) and MZo is the vertical moment resulting from the composition of all the weights items except the cargo.

The GM constraint for the unloading phases is similar to equation (12). Notice the non-linearity of the term $\Delta * (KB + BM)$, where Δ depends on the sum of container weights and KB and BM depend on Δ.

The proposed linearization is the following:

- in the beginning of a loading or unloading phase, the weights of all the containers that will be handled are known and thus an average weight per container can be obtained;
- at each stage, one adds to or subtracts from the previous value of Δ the average weight of the containers;
- the values of KB and BM will also be calculated from the hydrostatics curves, based on the displacement value calculated as mentioned above.

j-) Heel angle constraint

Assuming a maximum allowed heel angle A, this constraint for a stage k of a loading phase may be written in the following way:

$$-B \leq \sum_i \sum_j \sum_k X_{ijk} * p_i * y_j + \sum_{r=1}^{p-1} \sum_{i'} \sum_j \sum_{k'} X_{i'jk'} * p_{i'} * y_j - \sum_{r=1}^{p} \sum_{i'} \sum_j \sum_{k''} Y_{i'jk''} * p_{i'} * y_j + MYo \leq B \ , \ k \in KCP(p) \tag{13}$$

where in the above summations:

$i \in \{R_p \cup I_p\}$; $j \in J(i)$;

$i' \in \{I_r \cap R_p\}$; $k' \in KPCA(r, p-1)$;

$k'' \in KPDA(r, p)$

and $B = | \, tg(A) \, | * \Delta * GM$, y_j is the transverse coordinate of the center of cell which is loaded in stage k and MYo is the transverse moment resulting from the composition of all the weights items except the cargo.

Once more, there is a non-linearity in the term $\Delta * GM$, which depends on the weights of the containers that are being loaded or unloaded. A linearization similar to that proposed in item i can be done.

The heel constraint equation for a stage k of an unloading phase is similar to equation (13).

k-) Trim constraint

Assuming a maximum trim $TRIM_{max}$ for the ship, the equation for trim constraint in the stage k of a loading phase is:

$$-C \leq \sum_i \sum_j \sum_k X_{ijk} * p_i * (x_j - LCF) + \sum_{r=1}^{p-1} \sum_{i'} \sum_j \sum_{k'} X_{i'jk'} * p_{i'} * (x_j - LCF) - + \sum_{r=1}^{p} \sum_{i'} \sum_j \sum_{k''} Y_{i'jk''} * p_{i'} * (x_j - LCF) + MXo \leq C \;,\; k \in KCP(p) \quad (14)$$

where in the above summations:

$i \in \{R_p \cup I_p\}$; $j \in J(i)$;
$i' \in \{I_r \cap R_p\}$; $k' \in KPCA(r, p-1)$;
$k'' \in KPDA(r,p)$

and $C = MTC* \mid TRIM_{max} \mid$, x_j is the longitudinal coordinate of the center of cell j which is loaded in stage k, LCF is the longitudinal coordinate of the center of buoyancy of the vessel, MTC is the moment to change the trim of one unit and MXo is the longitudinal moment resulting from the composition of all the weights items except the cargo.

The non-linearity of equation 14 can be handled in similar way to those suggested for constraints 12 and 13.

The trim constraint equation for a stage k of an unloding phase is similar to equation (14).

l-) Cutting force and bending moment

Assuming a maximum permissible cutting force $FMAX_m$ for ship section m, the corresponding constraint for the stage k of a loading phase is written in the following way:

$$-FMAX_m \leq \sum_i \sum_j \sum_k X_{ijk} * p_i + \sum_{r=1}^{p-1} \sum_{i'} \sum_j \sum_{k'} X_{i'jk'} * p_{i'} - \sum_{r=1}^{p} \sum_{i'} \sum_j \sum_{k''} Y_{i'jk''} * p_{i'} + FRESU_m \leq; FMAX_m \;,\; k \in KCP(p) \quad (15)$$
$$and\, m = 1\, to\, MSECTION$$

where in the above summations:

$i \in \{R_p \cup I_p\}$; $i \in I(j)$;
$j \in JSE_m$; $i' \in \{I_r \cap R_p\}$;
$k' \in KPCA(r, p-1)$; $k'' \in KPDA(r,p)$

and $FRESU_m$ is the cutting force in the section m due to buoyancy and all the weights except the cargo.

Assuming a maximum permissible bending moment $MFMAX_m$ for ship section m, the corresponding constraint for the stage k of a loading phase is written in the following way:

$$-MFMAX_m \leq \sum_i \sum_j \sum_k X_{ijk} * p_i * d_{jm} + \sum_{r=1}^{p-1} \sum_{i'} \sum_j \sum_{k'} X_{i'jk'} * p_{i'} * d_{jm} - \sum_{r=1}^{p} \sum_{i'} \sum_j \sum_{k''} Y_{i'jk''} * p_{i'} * d_{jm} + MFRESU_m \leq MFMAX_m \;,\; k \in KCP(p) \quad (16)$$
$$and\, m = 1\, to\, MSECTION$$

where in the above summations:

$i \in \{R_p \cup I_p\}$; $i \in I(j)$;
$j \in JSE_m$; $i' \in \{I_r \cap R_p\}$;
$k' \in KPCA(r,p-1)$; $k'' \in KPDA(r,p)$
and $MFRESU_m$ is the bending moment in the section m due to buoyancy and all the weights except the cargo, d_{jm} is the distance between the cell j which is loaded in stage k and section m.

The cutting force and bending moment constraint equations for a stage k of an unloding phase are similar to equation (15) and (16), respectively.

3.4. MODEL SIZE AND COMPUTATIONAL COMPLEXITY

The container stowage problem, as represented by the mathematical model previously shown, is a combinatorial problem whose size depends on ship size (given by the number of twenty feet equivalent unities) and the container supply and demand at each port of the route. Anyhow, even for the smallest cases, the container stowage problem, from the point of view of combinatorial optimization, is a large scale problem. In fact, for a commercial size container ship of 1000 T.E.U's, calling in 4 ports, the mathematical model will have nearly 10^9 decision variables and approximately 10^6 constraints, assuming that the ship will sail fully loaded. Obviously, the solution of this problem cannot be obtained even using the fastest computer machine and the best integer linear programming computer code. The set of problem constraints does not have a particular structure which allows the utilization of specialized techniques for the large scale problem, such as lagrangean relaxation or algorithms for solving graph-realization problems.

Two other alternative methods for solving the container stowage problem were developed and are presented below. Both of them consider essentially the same characteristics of the complete model but, attempting to obtain a solution in a shorter processing time, they do not look necessarily for the optimality.

4. DECOMPOSITION OF THE COMPLETE MODEL

In order to overcome the computational difficulties associated to the exact solution of the complete model, the container stowage problem is decomposed in two subproblems namely:

a-) an assignment problem;
b-) a sequencing problem.

The solution of the first one gives a picture of the container cell occupation at the end of unloading and loading phases at each port of the route. That is, for a loading phase, the solution shows the cell to which each container is assigned, without keeping track of the loading sequence, and for the unloading phase, the solution shows which containers are removed from the respective cell, also without specifying the order of the unloading operations.

The decision variables X_{ijk} are similar to those of the complete model, with the difference that there are only two stages for each port, except for the first and last ones, where there is only one stage. For this problem, the objective is to minimize the number of container restows; the set of constraints contains almost all the types of constraints presented for the complete model, but it is much smaller in size. The main difference is that for each k one may have a lot of X_{ijk} equal to 1.

The second subproblem corresponds to determining the optimal loading or unloading sequence between two successive cell occupation pictures obtained in the solution of the assignment problem. In fact, there are as many sequence problems as the number of phases in the assignment problem.

For this problem, it is known a priori that the cell j will receive the container i(j) during the loading phase at the port p. The question is to determine in which stage k the container i(j) will be placed in cell j. In other words, the decision variables are:

$$X_{jk} = \begin{cases} 1, \text{if the cell j is occupied at the } k^{th} \text{ stage of loading phase;} \\ 0, \text{otherwise;} \end{cases}$$

Similarly for the unloading phases,

$$Y_{jk} = \begin{cases} 1, \text{if the cell j is emptied at the } k^{th} \text{ stage of unloading phase;} \\ 0, \text{otherwise;} \end{cases}$$

Notice that, unlike in the complete model, the number of effective stages in this case is known a priori. The objective function to be minimized is the total longitudinal crane movement. The constraints considered for each sequencing problem of a loading phase, are those or similar to those of type a, f, g, i, j, k and l shown in section 3 for the complete model. For an unloading phase the constraint type a is replaced by the constraint type c shown in section 3 for the complete model.

Another step taken toward reduction reducing the model size was the grouping of containers into classes, each of them having containers of same type, origin, destination and within a given weight range. Therefore, a value X_{ijk} in the solution of the assignment problem means that in the stage k, a container of class i is stowed into cell j.

5. IMPLICIT ENUMERATION ALGORITHM

The second alternative method developed for replacing the complete mathematical model in the solution of the container stowage problem aiming to reduce the processing time, is a general implicit enumeration algorithm. A given path in the enumeration tree corresponds to the assignment of values 1 to a subset of the decision variables X_{ijk} and Y_{ijk} of the complete model. The first edge (v_0, v_1) of the path corresponds to the first loading operation in the first port of the route; the following edges represent the successive loading and unloading operations along the ports of the route. The value 1 of variable X_{ijk} (or Y_{ijk}) over the k^{th} edge of the path means that the container i is stowed (removed) into (from) cell j in the stage k.

At each vertex of the path, the constraints of the problem are checked; if some of them is violated, the vertex is abandoned. In fact, the branching selection at each vertex helps itself to eliminate infeasible successors. A path can be interrupted at an intermediate and feasible vertex if a lower bound for the objective function considering all possible branching choices from the vertex on is greater than the objective function for a known feasible path until the last stage.

The objective function in this case is a weighted sum of the number of restows, longitudinal crane movement and amount of ballast used. Notice that this last term could not be introduced in the mathematical model. The use of ballast is considered whenever one reaches a vertex where either GM or trim constraint is violated.

Since the processing time for examining all possible paths using an exact implicit enumeration algorithm is extremely long, some heuristic rules were introduced with the aim of examining first the paths of the enumeration tree which seem to be the best ones. Doing so, it is expected that the search can be interrupted within an acceptable computation time and producing a good (eventually optimal) feasible solution for the container stowage problem. The heuristic rules incorporated to the algorithm are based on practical experience of shipping companies and port operators; some of them are shown below:

- assign a lot of containers with the same destination to a same bay;
- if containers of two different destinations were required to be stowed in a same bay, the containers for the nearer port should be stowed over the containers for the farther port;
- choose during the loading process a sequence of bays to be loaded which avoid the violation of the trim constraints.

6. COMPUTATIONAL EXPERIMENTS

The two alternative methods - decomposition with container grouping and implicit enumeration were implemented through computer programs for main frame. With the first method only very smalls problems could be solved using the linear programming TEMPO and a Unysis B-7900 computer.

The implicit enumeration algorithm was applied to solve the container stowage problem for 740 T.E.U's containership, calling in four ports with a global supply of 1200 containers, using the same hardware.

7. REFERENCES

1 B. C. Nehrling, Computer ship loading and unloading simulation, University of Michigan, Ann Arbor (1970).

2 J. J. Shields , Containership stowage: a computer-aided preplanning system, Marine Technology, Vol. 21, no 4, (1984).

3 W. C. Webster and P. Van Dyke, Automated procedures for the allocation of containers on shipboard, Computer Aided Ship Design Engineering Summer Conference, Ann Arbor (1970).

4 D. E. Beliech, A proposed method for efficient pre-load planning for containerized cargo ships, Naval Postgraduate School, Monterey, (1974).

5 T. Q. Thieu, A programmed loading procedure for containers and cargo ships, Naval Postgraduate School, Monterey, (1975).

6 O. P. Sha, Computer aided on board management, Computer applications in the automation of Shipyard operation and ship design V, Amsterdam, North Holland, (1985).

7 C. L. Barauna Vieira et alli, Planejamento e controle de movimentacao de conteineres, 12º Congresso Nacional de Transportes Maritimos e Construcao Naval, Rio de Janeiro, (1988).

8 D. J. Saginaw and A. N. Perakis, A decision support system for containership stowage planning. Marine Technology, Vol. 26, No. 1, (1989).

9 D. K. Scott and D. S. Chen, A loading model for a container ship, Matson Navigation Company, Los Angeles, (1978).

10 A. H. Aslidis, Optimal container loading, Massachusetts Institute of Technology, Cambridge, (1984).

11 A. H. Aslidis, Combinatorial algorithms for stacking problems Massachusetts Institute of Technology, Cambridge, (1989).

12 J. Dillingham and A. N. Perakis, Application of artificial intelligence in marine industry. Marad Report, MA-RD-770-87015, (1987).

13 J. Dillingham and A. N. Perakis, Design of an expert system for container stowage planning, Fleet Management Technology Conference, Baltimor, (1987).

14 J. Dillingham and J. Dumbleton, An expert system for container stowage planning. Spring Meeting/Star Symposium, SNAME, New Orleans, (1989).

15 H. A. Taha, Integer programming, Academic Press, New York, (1975).