Journal of the Operational Research Society (2000) 51, 1248–1255 0160-5682/00 $15.00

www.stockton-press.co.uk/jors

Container stowage planning: a methodology for generating computerised solutions

ID Wilson and PA Roach*

University of Glamorgan, UK

The container stowage problem concerns the suitable placement of containers in a container-ship on a multi-port journey; it requires consideration of the consequences each placement has on decisions at subsequent ports. A methodology for the automatic generation of computerised solutions to the container stowage problem is shown; objective functions that provide a basis for evaluating solutions are given in addition to the underlying structures and relationships that embody this problem. The methodology progressively refines the placement of containers within the cargo-space of a container ship until each container is specifically allocated to a stowage location. The methodology embodies a two stage process to computerised planning, that of a generalised placement strategy and a specialised placement procedure. Heuristic rules are built into objective functions for each stage that enable the combinatorial tree to be explored in an intelligent way, resulting in good, if not optimal, solutions for the problem in a reasonable processing time.

Keywords: artificial intelligence; planning; search; sea transport; container stowage

Introduction

Since the 1970s, *containerisation* (the packing of cargo into large, dedicated boxes, of different dimensions, enabling multiple units of cargo to be handled simultaneously) has facilitated transportation. Shipping companies compete around the world to provide profitable container transportation services. In order to increase the benefits of economy of scale, the size of *container ships* has increased. The increase in capacity has been typically from relatively small 350 Twenty Foot Equivalent Units (TEUs) to ships with capacities of more than 4500 TEUs.[1] The standardisation of containers has permitted the introduction of *inter-modal* transportation systems, that is, containerised cargo can be transported by rail, truck or sea due to its standard frame and dimensions, having enabled the introduction of carriers dedicated to this purpose.

Container ships are vessels possessing a structure that facilitates the handling of containerised cargo. These vessels travel on 'round-robin' routes. At each port of destination (POD) along the vessel's journey, containers may be unloaded and additional containers destined for subsequent ports may be loaded. Determining a viable configuration of containers that facilitates this process, in a cost-effective way, constitutes the container stowage problem. The work of determining a stowage configuration for a container ship, on leaving a port, is performed by human stowage planners. These planners work under strict time constraints, and are limited in the number of configurations that they can consider. The large container ships of today can require thousands of container *movements* (the loading, unloading or re-positioning of each container) at each port-of-call to complete the discharge and load process. Figure 1 shows such a ship from above and in transverse, longitudinal (along the length of the ship) and latitudinal (along the width of the ship) section; annotations mark positions of 'hatches' and 'bays' that are groupings of physical locations for container stowage. It is important that the process of loading and discharging container ships be carried out with a minimum of disruption. However, given this large number of container movements, reaching optimum efficiency is very difficult.

Container-ship efficiency is largely determined by the arrangement of containers both within the container-terminal and on the container ship. Determining the arrangement of these containers is an error-prone process relying largely on the intuitive skills of human planners. The planner must determine the optimum placement of containers so that all *constraints* (restrictions placed upon where and how containers can be stowed) are satisfied and *material handling costs* (the costs associated with loading, unloading and transporting cargo) are minimised. One of the most important problems associated with this optimisation process is the *re-handle*. A re-handle is a movement of a container which is only required in order to permit access to another, or to improve a stowage configuration to take into account expected loads at subsequent ports.

**Correspondence: Dr PA Roach, Division of Mathematics and Computing, University of Glamorgan, Pontypridd, Mid Glamorgan, CF37 1DL, UK. email: proach@glam.ac.uk*

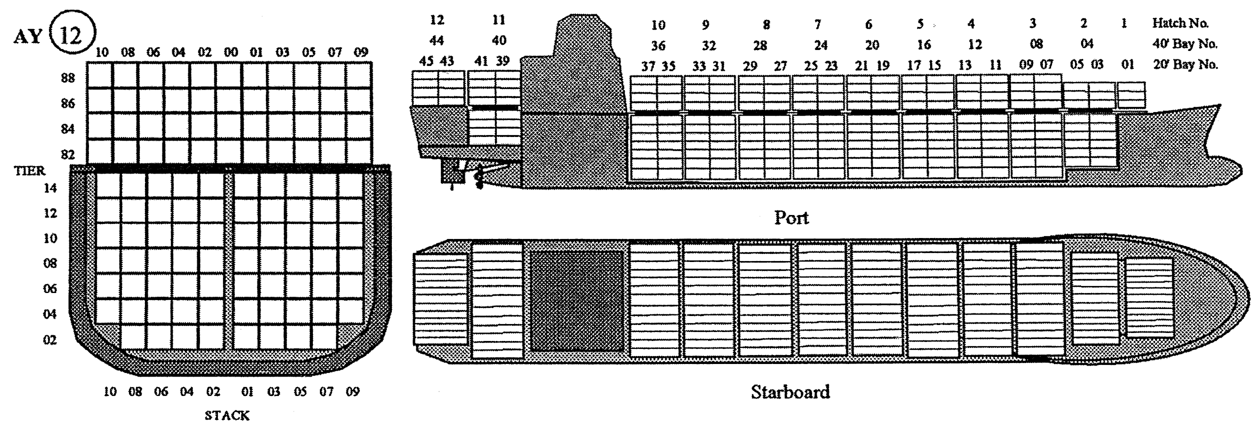

Figure 1 Stowage arrangement for a container ship.

These re-handles are considered the result of poor stowage planning. Minimising the number of re-handles reduces operating costs and helps maximise efficiency.

The complexity of stowage planning is increased by its multi-port nature (illustrated in Figure 2 showing a round trip including Europe and south-east Asia), that is, a plan for a stowage configuration at one port must take into account the consequences at subsequent ports. Given the 'round-robin' nature of the ship's voyage, planners will generally consider a fixed number of subsequent ports when producing stowage plans.

In full, the basis of the deep-sea container-ship stowage problem is the determination of a stowage configuration for a container ship, on leaving a port, so that no ship stability and stress constraints are violated and container re-handles are minimised. It must be noted that it is also necessary for the planner to contemplate other aspects of stowage efficiency; the most important of these is the reduction of ballast required by the vessel and efficient use of cranes when loading and unloading. The purpose of this paper is to show the development of an appropriate computerised methodology for generating stowage plans. The work presented was sponsored by Maritime Computer & Technical Services Ltd. Cardiff, Wales, UK.

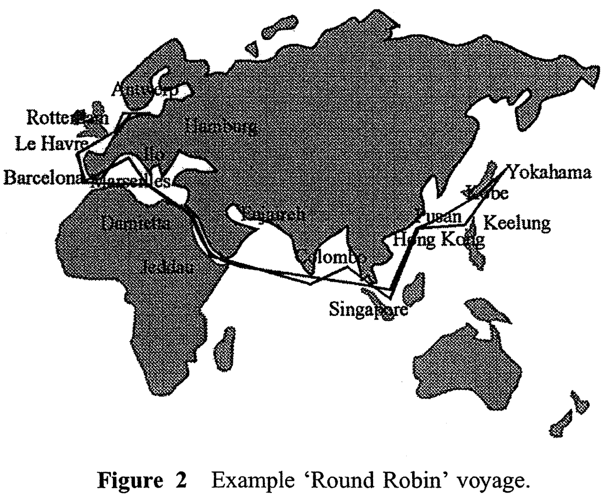

Figure 2 Example 'Round Robin' voyage.

Literature survey

The search for an efficient procedure for container stowage planning has drawn the attention of shipping companies and academic researchers since the 1970s. The methods used for producing solutions for the stowage planning problem have been grouped into the following five main classes: (1) simulation based upon probability; (2) heuristic driven; (3) mathematical modelling; (4) rule-based expert system; and (5) decision support systems.[2] None of these approaches have provided a solution to the complete stowage planning problem. A brief review of relatively recent research into automating stowage planning follows.

The first class includes the work completed by Shields,[1] which selects from a small number of heuristically generated stowage plans, which are evaluated by simulation of the voyage across a number of legs. A limited number of solutions are generated by varying the order (using a Monte Carlo method) in which loading heuristics are applied. The second class of automated planning processes incorporates human planners' experience encoded in the form of heuristics. This class includes the work completed by Martin[3] in automating stowage planning at container-terminals. These heuristics can produce a complete, but rarely near-optimum, solution to the container-terminal stowage problem without the interaction of a user. The third class includes work carried out by Botter[2] and Cho[4] exploring the application of mathematical models and linear programming to the problem. Those practising this method of solving the stowage problem have incorporated too many simplification hypotheses, which have made their approaches unsuitable for practical applications. The fourth class explores the potential of applying the theory of artificial intelligence to cargo stowage problems. This class includes the work of Dillingham,[5] Perakis,[6] Wilson[7–9] and Sato.[10] The work included within the fifth, and last class is entirely separate to the rest. No effort is made here to automate the generation of stowage solutions. Instead, sets of tools are made available to the users that *assist* in the generation of stowage solutions, the works of Saginaw,[11] Lang[12] and Sansen[13] belong to this class.

The partnership of stowage planning tools and human expertise has, to date, provided the best commercial improvements.

Problem size and complexity

The container stowage problem is a combinatorial problem the size of which depends upon ship capacity (given by the number of TEU units) and the container supply and demand at each POD. Combinatorial optimisation is made more complicated by the need to consider stowage across a number of ports. Even for the smallest cases, container stowage planning is a large-scale problem.

Determining the optimum allocation of specific containers to slots over even a few ports is computationally explosive and is not solvable in a commercially realistic length of time. An alternative method for solving the container stowage problem, developed by the authors, is presented in this paper. All characteristics of the problem are considered, but optimality is not necessarily sought.

Problem decomposition

In this section, considerations relating to the stowage planning model are described, and an overview of the model's underlying data-structures and planning processes is provided, along with a detailed discussion about stowage objectives and their corresponding mathematical formulation within the planning methodology.

Constraints relating to intact stability[14] are well documented in existing literature.[2] Therefore, the following discussion of the planning methodology deals only with the underlying heuristics used to generate stowage solutions and their subsequent evaluation. The following sections describe the system in relation to a specific voyage, for which the following should be noted:

- At each POD, unloading and loading occurred, but the latter did not begin until the former had finished;
- Ballast conditions were set by the user;
- Two cranes were available for loading and unloading at each POD.

Overview of the planning methodology

In order that the computational difficulties associated with producing an exact solution for the stowage problem be overcome, the authors propose that the process be decomposed into two sub-processes,[7,8] namely:

1. A strategic planning process; generalised containers are assigned to a blocked cargo-space in which slots corresponding to hatch-lids are grouped together (illustrated in Figure 3 and Figure 4).
2. A tactical planning process; specific containers are assigned to specific slots within blocks determined during the strategic planning phase (illustrated in Figure 5).

The strategic planning process provides a picture of the generalised cargo stowage distribution at the end of the unloading and loading processes at each POD. This approach models human planners' use of documents called the *General Arrangement* and the *Outline Plan*[15] to plan stowage, and reduces the combinatorial size of the problem whilst retaining the inherent characteristics of the problem.[7,8] The cargo space is blocked into locations that share both the same longitudinal position (indicated in Figure 3) and the same relationship to hatch-lids. The hatch-lids are the removable separators of above-deck and below-deck cargo, and are usually composed of a number of sections that interlock latitudinally. Above deck cargo can be placed across two sections of the lid (indicated in Figure 4). This creates blocks of cargo locations that have a partnership relationship with these sections of hatch-lids. This in

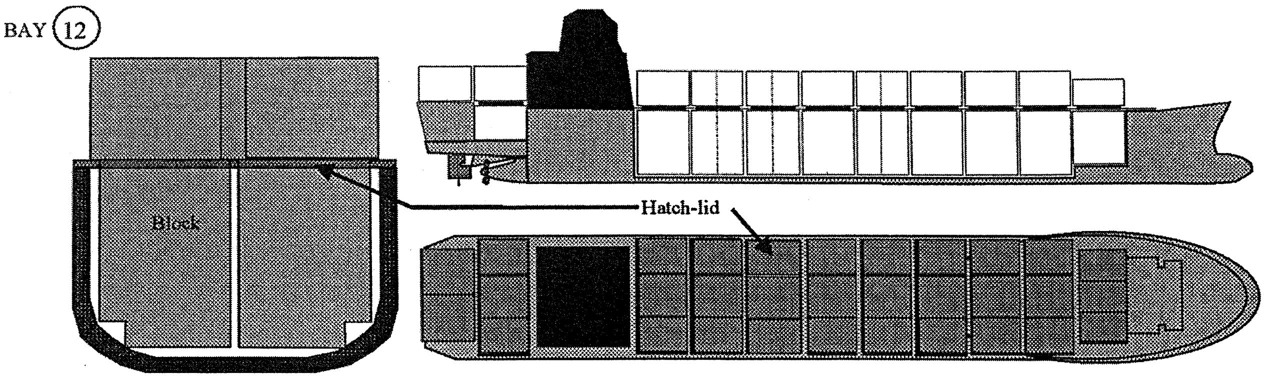

Figure 3 Blocked container ship abstraction.

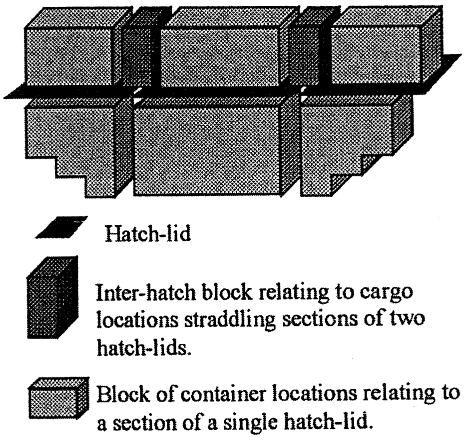

Figure 4 Example of cargo-space blocking relating to a single hatch.

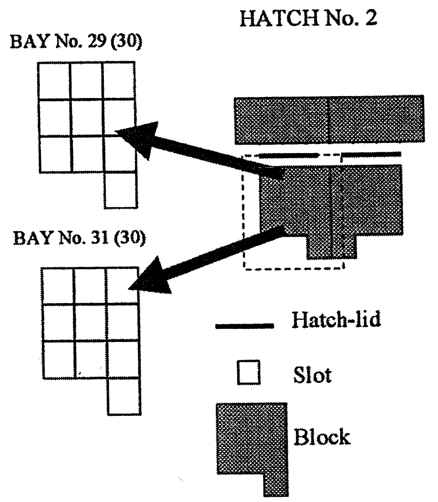

Figure 5 Relationshp between blocks and slots.

turn has consequences for which lids and containers must be removed by cranes to allow access to other containers and locations. Blocking the cargo-space of the container-ship enables the number of options for specifying container placements available at any stage of the planning process to be reduced from, perhaps, thousands of possibilities to within a hundred. The second, tactical planning, phase determines the exact slot occupied by each container at the current port-of-call. Notice that the problem is now reduced to allocating specific containers within a *part* of the container-ship (a block).[7,8] This avoids the combinatorial difficulties associated with attempting to make specific placements within the entire cargo space. Each block is composed of a number of locations in the same hatch (latitudinal grouping), shown in Figure 5. This blocked-stowage procedure models the human planner's conceptual approach and their use of documents called *Bay Plans*.[14]

Strategic planning phase

The following sections describe the general guidelines followed when making long-term stowage decisions, the underlying representation for the blocked cargo-space and the formulation of an objective function that measures how well a stowage configuration meets these objectives.

Strategic stowage objectives. The objectives of the strategic planning phase are to:

- Minimise the number of cargo spaces occupied by each destination;
- Maximise the number of cranes in operation at each POD;
- Minimise the number of hatch-lids moved;
- Minimise the number of re-handles;
- Minimise the number of cargo blocks occupied by containers.

Underlying model and definitions. The objective function used to evaluate solutions to the strategic planning problem requires a number of definitions that model the underlying structure of the problem:

- $DH_{ij} = 1$ if a container exists with destination i within hatch j else 0;
- $XC_i =$ the highest number of containers with destination i stowed within any of the hatches;
- $YC_i =$ the total number of the containers with destination i stowed minus XC_i;
- $DHH_{ijk} = 1$ if there exists a container with destination i within hatch j and another within adjacent hatch k;
- $DB_{ij} = 1$ if a container with destination i exists within stowage block j else 0;
- $DBR_{ijkl} = 1$ if a container with destination k exists within a block i and a container with a destination l exists within a block j where block i is above block j and destination k is further than destination l, else 0;
- $DL_{ij} = 1$ if a container with destination i exists under hatch-lid j else 0;
- VR_{ij} =The remaining capacity of under-lid stowage block i where containers are stowed in over-lid stowage block j and block i is below block j;
- $cr_i =$ the number of cranes at destination i;
- $nd =$ the number of ship's destinations;
- $nh =$ the number of ship's hatches;
- $nc =$ the number of containers;
- $nb =$ the number of stowage blocks;
- $nr =$ the number of upper stowage blocks;
- $nl =$ the number of hatch-lids.

The strategic objective function. The objective function used to evaluate solutions to the longitudinal stowage problem examines a stowage pattern in nine ways. The general expression of the objective function is:

$$f = ((f_1 \times w_1) + (f_2 \times w_2) + \cdots + (f_9 \times w_9))$$

Where f_i and w_i represent, respectively, an abstracted measure of one factor of the attractiveness of a solution and the weight, or importance, of that particular measure. A low value of f indicates a good solution.

The first term of the objective function, f_1, counts the number of hatches occupied by containers of each POD. Minimising the number of hatches having containers destined for different PODs facilitates good block stowage

$$f_1 = \sum_{i=1}^{nd} \sum_{j=1}^{nh} DH_{ij}$$

The second term of the objective function, f_2, counts how many hatches are occupied by containers of each POD and then compares this with how many cranes there are at that POD (in this case 2). The objective here is ensuring that the number of cranes at a given POD is a factor of the number of hatches occupied by that POD

$$f_2 = \sum_{i=1}^{nd} \left(\sum_{j=1}^{nh} DH_{ij} \right) cr_i$$

The third term of the objective function, f_3, provides a measure of how well the containers are spread between hatches and, hence, how efficiently the cranes will be able to operate. Ideally, containers should be spread to allow all cranes to be used simultaneously throughout the unloading process

$$f_3 = \sum_{i=1}^{nd} ABS(XC_i - YC_i)$$

The fourth term of the objective function, f_4, counts the number of PODs of cargo that exist within each hatch. Minimising the number of containers with different POD stowed in each hatch will lead to better block stowage

$$f_4 = \sum_{i=1}^{nh} \sum_{j=1}^{nd} DH_{ji}$$

The fifth term of the objective function, f_5, penalises stowage patterns in which containers of a particular POD are stowed inside two adjacent hatches (preventing the two cranes from working simultaneously)

$$f_5 = \sum_{i=1}^{nd} \sum_{j=1}^{nh} \sum_{k=1}^{nh} DHH_{ijk}$$

The sixth term of the objective function, f_6, counts the number of blocks occupied by containers of each POD. Minimising the mixing of destinations within blocks used leads to better overall stowage

$$f_6 = \sum_{i=1}^{nd} \sum_{j=1}^{nb} DB_{ij}$$

The seventh form of the objective function, f_7, counts how many containers are stowed on hatch-lids, beneath which are containers destined for an earlier POD. This particular type of stowage is thereby penalised by the evaluation function

$$f_7 = \sum_{i=1}^{nb} \sum_{j=1}^{nb} \sum_{k=1}^{nd} \sum_{l=1}^{nd} DBR_{ijkl}$$

The eighth term of the objective function, f_8, provides a measure of how well the containers are spread beneath hatch-lids and, hence, how efficiently the cranes will be able to operate

$$f_8 = \sum_{i=1}^{nd} \sum_{j=1}^{nl} DL_{ij}$$

The ninth term of the objective function, f_9, counts how many empty spaces exist below a hatch-lid that supports containers. Such occurrences are indications of poor stowage, as these spaces are unavailable without first removing the hatch-lid and any containers stowed there on

$$f_9 = \sum_{i=1}^{nr} \sum_{j=1}^{nb} VR_{ij}$$

Implementation using branch and bound search. The branch and bound approach to search is a very useful method for solving discrete optimisation, combinatorial optimisation and integer problems in general.[16] In particular, it is well suited to the blocked stowage problem. For the blocked stowage problem, the branch and bound algorithm and related sub-procedures are specialised as follows.

(1) *Initialisation*. The initial state is made up of the discharged cargo-space, an ordered list comprised of all containers to be loaded at the current port of call and an evaluation of the fitness of the stowage configuration. The cargo-space is composed of a list of areas to fill that correspond to blocks within the ship (as in Figure 4). The list of containers to be loaded has containers with the fewest available legal stowage locations first. Within the groups of different types of containers, those groups of containers with the furthest POD are placed first in sequence. The fitness of the solution reflects an abstract measure of the cost, base upon simulation of the unloading process at discharge ports.
(2) *Branching*. New solutions are generated that reflect every possible placement of the first container in the load-list within the cargo-space associated with this partial-solution. All invalid solutions are then removed from the list of new states. If after expanding a partial solution a feasible solution for the stowage problem is found, then it is set aside.
(3) *The search strategy.* The candidates produced during the branching process are ordered according to the least fitness value determined by the objective function and the least number of containers remaining within its

associated load-list. This new list is placed at the front of the existing list of partial-solutions. This strategy reflects a depth first approach to the search process.

(4) *Pruning.* When one candidate sub-problem has the same, or worse, fitness value as another but has more containers to load then it can be deleted from the pool of partial solutions.

(5) *Choice of new sub-problem.* The partial-solution with the best fitness value is selected as the new current candidate problem and the algorithm continues in a similar manner until n solutions are found and d destinations are processed. Upon delivery of n candidates the search process for the current port of call is terminated, the problem is reinitialised, and the process repeated again for each of the n solutions at the next POD. This process simulates a planning procedure at a given number of destination ports (illustrated in Figure 6). Once this process has been repeated for each destination, the best solution is the one with the least summation of the fitness values accumulated at each port.

Tactical planning phase

The best, generalised, long-term solution that was determined during the strategic planning phase is refined in the tactical planning phase. The following sections describe the general guidelines followed when making short-term stowage decisions, the underlying representation for the cellular cargo-space and the formulation of an objective function that measures how well a stowage configuration meets these objectives.

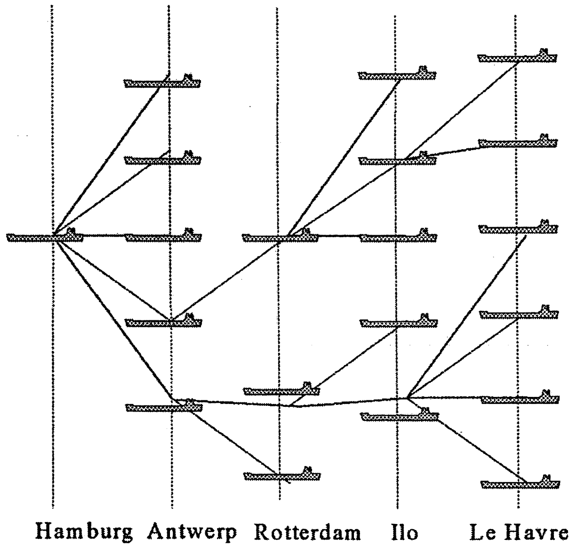

Figure 6 Tree showing some possible partial solution paths.

Tactical stowage objectives. A large number of generalised and specialised stowage heuristics exist that direct the placement of containers.[7] For the model under consideration, the following are considered salient:

- Re-handles are to be minimised;
- Container weight is to be graded upwards in the cargo space, heaviest to lightest;
- Stacks (vertical collections of containers) with mixed POD are to be minimised.

Underlying model and definitions. The objective function used to evaluate solutions to the tactical planning problem requires a number of definitions that model the problem's underlying structure, specifically:

- C: $\{c_1, \ldots, c_{nc}\}$ is the set of all containers;
- nc is the number of containers;
- D_i is the destination port of container i;
- DR_i is the set of re-handles related to container i, for example the containers stowed above one to be discharged;
- DW_i is the set of containers in the same stack stowed above container i and having a greater weight;
- DS_i is the set of containers stacked with container i and having a different POD.

The tactical objective function. The general expression for the objective function for the problem of container assignment within a block is:

$$f = (f_{10} \times w_{10}) + (f_{11} \times w_{11}) + (f_{12} \times w_{12})$$

where w_i is the weighting associated with function f_i. A low value of f indicates a good solution.

The first term of the objective function, f_{10}, counts the number of re-handles.

$$f_{10} = \sum_{i=1}^{nc} \sum_{j=1}^{nc} \begin{pmatrix} 1 & \text{if } j \in DR_i \\ 0 & \text{else} \end{pmatrix}$$

The second term of the objective function, f_{11}, counts the number of containers with a different POD stowed in the same stack.

$$f_{11} = \sum_{i=1}^{nc} \sum_{j=1}^{nc} \begin{pmatrix} 1 & \text{if } j \in DS_i \\ 0 & \text{else} \end{pmatrix}$$

The third term of the objective function, f_{12}, counts the number of containers with a greater weight stowed above each other in the same stack.

$$f_{12} = \sum_{i=1}^{nc} \sum_{j=1}^{nc} \begin{pmatrix} 1 & \text{if } j \in DW_i \\ 0 & \text{else} \end{pmatrix}$$

Optimisation using Tabu search. Tabu search can be viewed as an iterative technique which explores a set of problem solutions by repeatedly making moves from one solution s to another solution s' located in the neighbourhood $N(s)$.[17] For the container stowage problem, s is the stowage configuration for the entire container-ship and $N(s)$ is the set of all configurations obtained by making moves within a single hatch, with each hatch being optimised separately. Here, a move is the swapping of the contents a of location p with the contents b of location q (where the contents of one location can be 'empty'). Note that it is the specific grouping of a, b, p and q that define the move; the swapping of different contents in the same locations would be regarded as an entirely different move. For each solution s, $M(s)$ is the set of legal moves m that can be applied to s to obtain a new solution $s' = s \oplus m$, giving $N(s) = \{s' | \exists m \in M(s) \text{ with } s' = s \oplus m\}$.[17] These moves are performed with the aim of reaching a near optimal solution by the evaluation of some objective function $f(s)$ to be minimised. To prevent the search process from returning a local optimum f, a guidance procedure is incorporated that accepts a move from s to s' even when $f(s') > f(s)$. Should no improving move be found in a given number of iterations then the original, best, local solution is returned as the global solution. This in itself could lead to cycling causing the process to return repeatedly to the same local solution without moving towards a global solution. Here, cycling is prevented through use of a recency list of Tabu moves that cannot be repeated or reversed for a number of iterations k. For the problem of container assignment within a block, the neighbourhood $N(s)$ was determined by the blocked stowage procedure. An initial random value for s would suffice, but the application of a packing heuristic that generates a sensible value for s ultimately improves the efficiency of the search algorithm.[7] Given this, the form of the procedure used to optimise the arrangement of containers is as follows:

$$s^* := f(s), \qquad k := 1, j := 1$$

while $(j < \max(j))$ and $(k < \max(k))$ and $(f(s^*) \neq 0)$
    $j :\neq j + 1$
    $M^* \subseteq N(s, k)$ (all legal, non-tabu, states)
    Choose the best s' in M^*
    $s := s'$
    If $f(s') < f(s^*)$ then $s^* := s'$, $k := 1$ else $k := k + 1$.
End of while

The form of Tabu search described is a very simple version. The pre-planning that occurs during the strategic phase prunes the search space significantly, resulting in sub-problems where optimal solutions can be found easily. Consideration given to aspiration levels, intermediate or long term memory, and other features described in the literature[17] proved unnecessary.

Computational experiments

The described methodology was implemented using commercial data.[19,20] In brief, a nine-hatch container-ship was modelled, with an on-deck capacity of 352 TEU and an under-deck capacity of 336 TEU giving a total TEU capacity of 688. The loading and unloading strategies for four ports were considered with a total of 696 containers being loaded and 312 containers being unloaded for 1008 movements. Of the containers handled:

- 54% were 20 feet in length;
- 44% were 40 feet in length;
- 2% were of other lengths;
- 20% required refrigeration (so-called *reefers*);
- 14% required special segregation (due to the hazardous or tainting nature of their contents);[18]
- 66% were a general type.

Ship specific constraints upon where different lengths and types of containers can be stowed were:

- All on-deck bays can have any length of container stowed there;
- Under-deck hatches 1 and 8 can have 40 feet and 20 feet containers stowed there;
- Under-deck hatches 2 and 7 can only have 20 feet containers stowed there;
- Under-deck hatches 3, 4, 5 and 6 can only have 40 feet containers stowed there;
- Under-deck hatches 1 and 8 are specially treated so that tainting cargo can be stowed there;
- Under-deck hatches 3, 4, 5 and 6 can have reefers stowed there.

Results were obtained on a 166 MHz Pentium with 40 megabytes of memory using Allegro Lisp to encode the blocking and GFA (a PC-based 3GL with a high degree of functionality and graphic display features) to encode the specific placement algorithm. The software used the weights given in Table 1 for each of the fitness functions. A

Table 1 Fitness function weightings

Function (f_n)	*Weight* (w_n)
f_1	2
f_2	4
f_3	3
f_4	1
f_5	10
f_6	1
f_7	4
f_8	2
f_9	3
f_{10}	3
f_{11}	1
f_{12}	2

generalised solution to the described problem was obtained in approximately 90 minutes whereas specialised solutions for all blocks were produced in under an hour. This contrasts with, typically, a period of several days required by human planners. Solutions were evaluated using the stowage objectives embodied in the objective function, as detailed in this paper. Further detail on the analysis of solutions has been reported elsewhere.[7,9] The authors take the view that the production of commercially viable solutions demonstrates the effectiveness of the methodology described in this paper. The stowage plans generated were found by experts at P&O Containers Ltd., London, to be commercially viable, and comparable with those generated by experienced human planners.

Conclusion

Providing an optimal solution to the container to slot allocation problem is considered to be NP-Hard and cannot be solved for commercial ship sizes in a reasonable amount of processing time using available computer software and hardware.[2] However, modelling how human planners solve this problem has resulted in the development of a heuristic driven computerised methodology that provides workable solutions to the problem.

References

1 Shields JJ (1984). Container-ship stowage: a computer-aided preplanning system. *Marine Technol* **21**: 370–383.
2 Botter RC and Brinati MA (1992). Stowage container planning: a model for getting an optimal solution. *IFIP Trans B (Applications in Technology)* **B-5**: 217–229.
3 Martin GL, Randhawa SU and McDowell ED (1988). Computerised container-ship load planning: a methodology and evaluation. *Comput Ind Engng* **14**: 429–440.
4 Cho DW (1984). *Development of a Methodology for Containership Load Planning*. PhD Thesis, Oregon State University.
5 Dillingham JT and Perakis AN (1986). Application of artificial intelligence in the marine industry. *Presented at the Fleet Management Technology Conference*. Boston, USA. (Available from author).
6 Perakis AN and Dillingham JT (1987). The application of artificial intelligence techniques in marine operations. *Presented at the 1987 Ship Operations, Management and Economics International Symposium USMMA*. Kings Point, New York. (Available from author).
7 Wilson ID (1997). *The Application of Artificial Intelligence Techniques to the Deep-Sea Container-Ship Cargo Stowage Problem*. PhD Thesis, University of Glamorgan.
8 Wilson ID and Roach PA (1997). The deep-sea container-ship stowage problem: modelling and automating the human planning process. In: Pearson DW (ed) *Proceedings of the Second International ICSC Symposium on Intelligent Industrial Automation*. ICSC: Canada/Switzerland, pp 129–135.
9 Wilson ID and Roach PA (1999). Principles of combinatorial optimisation applied to containership stowage planning. *J Heuristics* **5**: 403–418.
10 Sato K, Itoh H and Awashima Y (1992). Expert system for oil tanker loading/unloading operation planning. In: Barauna Vieira C, Martins P and Kuo C (eds). *Computer Applications in the Automation of Shipyard Operation and Ship Design, VII*: Elsevier Science Publishers: B.V. (North Holland).
11 Saginaw DJ and Perakis AN (1989). A decision support system for container-ship stowage planning. *Marine Technol* **26**: 47–61.
12 Lang GJP (1985). Some computer aids in the loading of deep sea container vessels—a personal experience. *Presented at the Computer Applications in Operation and Management of Ships and Cargoes*. London. (Available from author).
13 Sansen H (1989). Ship-Planner, a conception support tool for the bay plans of container ships. Systemia, Domaine de ST Hilaire, Pichaury 13290 AIX LES MILLES, FRANCE.
14 Goldberg LL (ed). (1980). *Principles of Naval Architecture*. J.P. Comstock: New York.
15 Roach DK and Thomas BJ (1994). *Portworker Development Programme Unit C.2.2. Container-Ship Stowage Plans. International Labour Organisation*: ISBN 92-2-109271-2.
16 Cao B (1992). Transportation problem with nonlinear side constraints: a branch and bound approach. *ZOR—Methods and Models of Opns Res* **36**: 185–197.
17 Glover F (1993). A user's guide to tabu search. *Annals Opns Res* **41**: 3–28.
18 Stowage and Segregation Guide to IMDG-Code. Published by U.O. Storck Verlag (Stahltwrek 7, D-2000 Hamburg 50, Germany).
19 Private communication. Technical information provided by Maritime Computer & Technical Services Ltd., Cathays, Cardiff, South Glamorgan, Wales, UK.
20 Private communication. Technical information provided by P&O Containers Limited, Beagle House, Braham Street, London, E1 8EP, UK.

Received February 1998;
accepted June 2000 after two revisions