AN AIMMS-BASED DECISION-MAKING MODEL FOR OPTIMIZING THE INTELLIGENT STOWAGE OF EXPORT CONTAINERS IN A SINGLE BAY

JIAN JIN
Scientific Research Academy, Shanghai Maritime University, China
and
Jiujiang Port Co. Ltd of Shanghai International Port Group
Jiangxi Province, China

WEIJIAN MI*
Scientific Research Academy, Shanghai Maritime University, China
and
Container Supply Chain Tech. Engineering Research Center
Shanghai Maritime University, China

ABSTRACT. Stowage operations in container terminals are an important part of a port’s operational system, as the quality of stowage operations will directly affect the efficiency of port loading and discharge operations, and the scheduling of container shipping liners. The intelligent stowage of containers in container ships was studied in this work. A multi-objective integer programming model was constructed with the minimization of container rehandling, yard crane movements, and the sum of weight differences between stacked container pairs as its objective functions, to address the need for intelligent optimization of single bay export container stowage on a ship’s deck. This model also satisfies the stability requirements of preliminary stowage plans drawn by shipping companies, and the operational requirements of container terminals. Linear computational methods were then constructed to transform non-linear constraints into linear ones for better AIMMS solution. Through numerous case analyses and systematic tests, it was shown that our system is able to rapidly solve for stowage planning optimization problems with complex preliminary stowage data, thus proving the applicability and effectiveness of this model. In particular, the application of this model will simultaneously address the safety of ship voyages, the transportation quality of shipping containers and other forms of cargo, and the cost efficiency of ship operations. In addition, this model will also contribute to the optimization of loading and discharge processes in container terminals. Therefore, our model has immense practical value for improving port productivity, as it will contribute to the organization of port operations in a rational, orderly and effective manner.

1. Introduction. The variability of ship stowage and uncertainties in the smartization of ship scheduling and ship stowage makes it necessary for ports to create rational and effective export container stowage plans for any situation that may arise during their operations. The studies on new technologies have begun to apply

2010 Mathematics Subject Classification. 68U35.
Key words and phrases. Container terminals, stowage plans, single bay stowage, AIMMS decision-making models.
* Corresponding author: Weijian Mi.
to improve the handling efficiency in traditional terminals [8, 16]. And the other studies around the world on export container ships tend to focus on vessel stability, cargo unloading and rehandling rates, and decision-making factors in preliminary stowage plans, but rarely discuss specific slot allocation plans, the dispatch of containers from container yards, and ship loading sequences. In practice, the formulation of plans based on the stowage plans provided by the shipping company will ensure that the stability and center of gravity requirements of a shipping vessel will be met, and the container relocation problem for target discharging ports will also be taken into consideration in these preliminary stowage plans. Hence, the level of effort placed on the consideration of these decision-making factors may be reduced.

The intelligent stowage problem for container terminals is widely regarded as a fully NP problem. In particular, the optimization of intelligent stowage in single bays may be treated either as a single-layered model, or a single layer in the development of an overall stowage plan, during deconstruction and analysis. Avriel et al. [3] introduced a 0–1 integer programming (IP) model and heuristic algorithms for optimizing stowage in a single ship bay. In this optimization, it was assumed that all containers have the same features and belong to the same set of containers, and the minimization of container rehandling rates was set as the objective function. Wilson [22] combined stowage planning and the scheduling of ship loading operations, and considered the balance between quay crane and yard crane operations. Wilson et al. [23,25] proposed a theoretical model based on multiple technical constraints to improve decision support systems for intelligent stowage planning. In this particular work, the stowage plan was divided into two stages: the first stage is a strategic planning stage, in which the export containers were classified according to their characteristics, e.g., size, type, and discharge ports, and the container sets were then assigned to specific slots in the bay using the branch and bound method, from which a preliminary stowage plan was formulated. The second stage is the tactical planning stage, in which each container is assigned to specific slots according to a determined stowage sequence; the Tabu search heuristic was then used to formulate a detailed stowage plan. Kim et al. [13] suggested that container stacks on the deck should be arranged in a rational manner to speed up quay crane operations, while container stacks in the yard should be arranged in a rational manner to facilitate orderly yard truck operations, and the beam search algorithm may be used to formulate stowage plans. Abrosino et al. [1] used the minimization of total ship loading time as an objective function, and considered practical constraints such as the containers’ type and weight class, to create stowage plans that suited different vessel structures. Furthermore, export containers with the same discharge port were stowed in the same bays to avoid inefficient container rehandling during ship discharging. Imai et al. [11] suggested that stowage plans and ship loading plans are primarily determined by two criteria: the first is ship stability, while the second is the minimum number of container rehandles. Neither criterion is prioritized over the other, so this may be treated as a multi-objective integer programming problem. Sciomachen and Tanfani [19] constructed the 3D-BPP model to minimize total loading time, and maximize quay crane usage rates. However, the remaining deterministic constraints, e.g. yard crane operations, were not considered in this model. Lee and Lee [4,14,21] presented a heuristic for retrieving containers to be loaded onto a ship in a specified order from some given yard, with the objective functions being the minimization of container movements and yard crane operations.
times; the container movements were reduced by repeatedly formulating and solving a binary integer program. Ren et al. [17,20] proposed a greedy algorithm-based tree search to perform the assignment of container sets to container slots, which specifically optimizes the ship-loading priority of the container sets. Junqueira et al. [12] proposed solutions for the container loading problem based on mixed integer linear programming models that consider the horizontal and vertical stability of the cargo. Delgado et al. [9] decomposed the stowage problem into two phases: the first phase is the master planning phase where container sets are distributed to bay sections, while the second phase is the slot planning phase in which the containers of each bay section are assigned to slots. In this work, they also gave a clear definition of the Container Stowage Problem for Below Deck Locations (CSPBDL), and constructed the Constraint Programming (CP) and Integer Programming (IP) models. Kang et al. [10] proposed a hybrid genetic algorithm for the three-dimensional bin packing problem to maximize the number of containers loaded onto a ship. Salido et al. [18] proposed a decision support system for the Container Stacking Problem (CSP), the Berth Allocation Problem (BAP) and the Quay Crane Assignment Problem (QCAP). Winter [24] proposed a multi-layer search based on heuristic algorithms for the three-dimensional container stowage problem. Chen and Lu [7] proposed a two-staged stowage model based on mixed integer programming and a hybrid sequence stacking algorithm. Botter [5] proposed an optimal model for the stowage of export containers based on linear programming; however, the assumptions that were used to simplify their model led to discrepancies between reality and the model's results. Avriel et al. [2] proposed a 0-1 integer programming model based on heuristic algorithms, and demonstrated the reliability and computational efficiency of this model through numerous example calculations on actual data. However, this heuristic method is only able to solve simplistic stowage problems, and thus lacks viability.

In summary, the studies described above mainly focus in the optimization of ship stability, the usage rates of facilities and resources, and the total ship loading time. However, other impact factors such as the number of container rehandles, the movement of yard cranes and the application of new technologies are either neglected or simply passed over briefly [6,15]. Furthermore, some of these studies place too much focus on meeting the requirement for ship stability, and used stability, trim, and hull strength as constraints. However, these constraints are unnecessary for port operations, and the preliminary stowage plans of a ship already ensure the compliance of stability constraints. Furthermore, containers of the same set (i.e., export containers on the same shipping voyage, dimensions, discharge port, weight class, container type, and cargo type) are already stowed together in some bay of each ship. Therefore, the excessive addition of stability-related constraints in export container stowage models will expand the scale of a model's computational requirements and reduce computational efficiency.

2. Description of the container stowage problem for single bays. A port’s stowage plans refer to ship stowing decisions made from the port’s perspective, and thus correspond to actual stowage; this is referred to as practical stowage in this text. The primary principle that practical stowage adheres to is compliance with a port’s operational requirements, which includes: (1) compliance with the container retrieval rules of the yard; (2) compliance with the requirements of the operational plans of each ship; (3) the guaranteeing of rational and orderly
Figure 1. A schematic diagram of the decision making processes in stowage planning

machinery movements; this includes the minimization of yard machinery movements, and the minimization of the movement distances by the ships' machinery during their multilinear operations. To address the needs of ship discharging operations while satisfying the ship stability requirements of preliminary stowage plans, the port and the ships were combined into a single system to improve the operational efficiency of the port's production systems, and satisfy the transportation and scheduling needs of the ships.

As shown in Figure 1, the decision making processes in stowage planning may be divided into two phases: the first phase is the master bay planning (MBP) phase, while the second phase is the slot planning (SP) phase. [8]

The problem being studied in this work is the single bay container stowage problem, i.e., the aforementioned SP decision-making problem. This refers to the assignment of one or more sets of containers into a single predetermined bay section via an optimized stowing sequence, so that the containers to be stowed have a one-to-one correspondence with the ship's slots. In this context, a “container set” refers to export containers on the same shipping voyage, with the same dimensions, discharge port, weight class, container type and cargo type.

As mentioned previously, the practical stowage of a port is based on the constraints of the preliminary stowage plan, and the output of a preliminary stowage plan is the input for the constraints of practical stowage. Due to the previously established stability constraints in the preliminary stowage plans drawn by shipping companies (stability, trim, hull strength and so on), constraints of this variety have no bearing on the practical stowage plans of a port. These constraints will instead, expand the computational scale of the model and reduce computational efficiency. Hence, stability issues were not given excessive consideration in our model. Furthermore, the true focus of this research is the on-deck shipping slots. The factors that need to be considered during bay assignment of under-deck shipping slots are completely different from those for on-deck shipping slots, e.g., the number of hatch cover movements, etc. In summary, the model proposed in this work is suitable for the stowage of export containers on a single-bay deck. On-deck loading operations must adhere to the following principles:

(1) Compliance with the permissible load weights defined in the preliminary stowage plan

The weight of each container to be loaded into a slot cannot be lower than the minimum permissible weight, nor can it exceed the maximum permissible weight, in accordance with the weight capacities of each slot defined in the preliminary stowage plan. Furthermore, the difference between the weights of each pair of
stacked containers (one stacked above the other) must remain within the tolerable limits of the preliminary stowage plan. Otherwise, the strength and structure of the ship may be compromised, thus threatening the safety of the voyage.

(2) Compliance with the “light on heavy” principle

The purpose of stowing light cargo above heavy cargo is to satisfy a ship’s stability requirements. When a ship is tilted by external forces (e.g., wind and waves), this maximizes the ship’s ability to automatically restore itself to its original, balanced position. Modern container ships are often heavily loaded with containers on their decks, which increases the height of the ship’s center of gravity and reduces its stability to an extent. Therefore, this factor must be adequately accounted for during stowage, and heavier cargo should be stowed in lower slots.

(3) Compliance with the permissible maximum stacking heights of each container stack

According to the permissible single-stack stacking height defined in the preliminary stowage plan, containers may not be stacked limitlessly, and container stacks must not exceed the permissible height. This is to ensure compliance with the hull’s structural requirements, for ensuring the stability of the ship and the safety of the containers and cargo during shipping.

(4) The restriction of container rehandles in yard stacks

A stowage plan should maximize the coordination between the top-to-bottom retrieval of containers at a port and the bottom-to-top loading of a ship, to restrict the number of container rehandles in yard stacks. This will optimize the resource allocation of ship loading operations and minimize unnecessary movement times, therefore increasing the scheduling efficiency of yard crane – container truck – quay crane loading operations. A ship may otherwise spend too much time in a port, resulting in shipping schedule delays. Therefore, in the single bay stowage process described below, we will calculate the number of container rehandles in yard stacks during the loading of export containers on the deck, and minimize this number.

3. The construction of a model for optimizing the intelligent stowage of export containers in a single bay.

3.1. Symbol definitions. (1) Dimensions

\( I \): The set of containers to be assigned, \( i, k \in I \)

\( J \): The set of slots to be assigned, \( j, l \in J \)

\( S \): The set of stowage sequences, \( s \in S \)

\( R \): The set of slot row numbers, \( r \in R \)

(2) Parameters

\( A_{ir} \): A 0 – 1 array, which represents the subordinate relationships between slot \( j \) and row \( r \). If slot \( j \) is in row \( r \), then \( A_{ir} = 1 \);

\( YP_{ik} \): A 0 – 1 array, which represents the above-below relationship between containers \( i \) and \( k \) in the same vertical stack in the yard. If \( i \) is above \( k \), then \( YP_{ik} = 1 \).

\( VP_{il} \): A 0 – 1 array, which represents the above-below relationships between slots \( j \) and \( l \). If \( l \) is above \( j \), then \( VP_{il} = 1 \).

\( W_{j}/M_{j} \): The minimum/maximum permissible weights of a slot \( j \).

\( D_{jl} \): The tolerance of a stack containing slots \( j \) and \( l \) for differences in weight between stacked container pairs

\( GP_r \): The permissible number of 20 ft GP containers that may be stacked in a single stack on the deck
1106 JIAN JIN AND WEIJIAN MI

$YB_i$: The yard-bay of container $i$ (location)

$Q_s$: The one-dimensional array of sequence number $s$

$F_j$: The preliminarily assigned container weight at slot $j$

$G_i$: The gross weight of container $i$

(3) Decision-making variables

$X_{ij,s}$: A 0 – 1 3D array, which represents the one-to-one correspondence between container $i$, slot $j$ and stowage sequence $s$. If container $i$ is dispatched to slot $j$ via stowage sequence $s$, $X_{ij,s} = 1$; otherwise, $X_{ij,s} = 0$.

$F_{1ik}$: An auxiliary decision-making variable, and a 0 – 1 array, which represents the order of containers $i$ and $k$ in the stowage sequence. If $i$ is dispatched before $k$, then $F_{1ik} = 0$; if $k$ is dispatched before $i$, then $F_{1ik} = 1$.

$F_{2s}$: An auxiliary decision-making variable, and a 0 – 1 array, which indicates whether the two adjacent containers in the stowage sequence are stored in the same yard-bay. If they are not in the same bay, then $F_{2s} = 1$; if they are in the same bay, then $F_{2s} = 0$.

3.2. Constraints. (1) The uniqueness constraint

A container $i$ may only be stowed in a single slot $j$, using a single stowage sequence $s$. This constraint may be expressed as:

$$\sum_{j \in J} \sum_{s \in S} X_{ij,s} = 1$$

$$\sum_{i \in I} \sum_{s \in S} X_{ij,s} = 1$$

$$\sum_{i \in I} \sum_{j \in J} X_{ij,s} = 1$$

(1)

(2) The permissible stacking limits of each container stack

![Figure 2. The permissible stacking limit of each stack](image)
The number of containers, $GP$, in each container stack on the deck must comply with the limits set in the preliminary stowage plan. As shown in the figure below, the permissible number of stacked containers in the 16th stack is 5, while the permissible number for stacks 06, 08, 10, 12 and 14 are 6; any further stacking will exceed this limit. This constraint may be expressed as:

$$\sum_{ijs} X_{ijs} \cdot A_{jr} \leq GP_r$$

(2)

(3) The permissible weight limits of each slot

The gross weight of the containers must remain within the range of permissible weights for each slot, as defined in the preliminary stowage plan, so that the weights of the export containers have some level of variability within the defined weight limits. This will satisfy the stability requirements of the preliminary stowage plan, balance the distribution of slot weights, and make sure that the lighter containers are stacked on top of the heavier containers. As shown in the figure below, the lowest permissible weight on the 92nd layer is 10t, while the maximum permissible weight is 12, so only the container with a weight of 11 can be stored here, not the container with a weight of 15. This constraint may be expressed as:

$$W_j \leq \sum_{is} X_{ijs} \cdot G_i \leq M_j$$

(3)

(4) Constraints in the difference in weight between stacked container pairs

Lighter export containers must be stacked above heavier containers, as shown in the figure below. For each stack, the difference in weight between a pair of stacked containers cannot be too large, and must remain within the tolerance of each stack. This constraint may be expressed as:

$$\left(\sum_{is} X_{ijs} \cdot G_i - \sum_{is} X_{ils} \cdot G_i\right) \cdot VP_{jl} \leq D_{jl}$$

(4)

(5) The “bottom-to-top” constraint

Containers that are meant to be stowed in lower slots must be dispatched prior to containers in upper slots, to prevent the stowage of containers meant for upper slots.
3.3. **Objective functions.** (1) Minimization of container rehandles

If the first containers to be dispatched are placed below containers that will be dispatched later in the container yard, the latter must be temporarily moved to in lower slots. As shown in the figure below, if the container no. 18 (its stowage sequence number) is the last container to be dispatched, the containers above this container cannot be dispatched (i.e., containers with stowage sequence number 20, 22, 25 and so on), lest they are stored in a lower than intended slot. This constraint may be expressed as:

\[
\sum_{i_s} X_{ij_s} * Q_s \leq \sum_{l} \left[ \sum_{i_s} (X_{ij_s} * Q_s) * VP_{jl} \right]
\]  

(5)
Figure 6. A schematic of container rehandling in yard stacks

an empty yard slot to enable the retrieval of the former – this action is denoted a “container rehandle”. As shown in the figure above, the numbers in the box represent the containers’ dispatch sequence; in this yard stack, container no. 18 needs to be loaded first, but since it sits below containers no. 20 and no. 22, they need to be moved to an empty container rehandling slot on the side for the retrieval of container no. 18. This objective may be expressed as:

$$Obj_1 = \sum_{ik} YP_{ik} \ast \left[ If \left( \sum_{js} X_{ijs} \ast Q_s - \sum_{js} X_{kjs} \ast Q_s \right) > 0, 1, 0 \right]$$  \hspace{1cm} (6)$$

(2) Minimization of yard crane movements

The figure below illustrates two different stowage plans for some bay, and the containers to be stowed are being stacked in yard-bays M and N. A yard crane movement from bay M to bay N, or the reverse, is treated as one movement. When

Figure 7. The effects of stowage plans on the number of yard crane movements
dispatching containers for ship loading, Stowage Plan I will require repeated movements by the yard crane between the two yard-bays, with this process being: N-M-N-M-N (5 movements). The corresponding container retrieval process in Stowage Plan II is only N-M, so all of the containers in bay N are dispatched, before the yard crane moves to bay M to continue dispatching containers, which results in only a single yard crane movement. Hence, Stowage Plan II is superior to Stowage Plan I.

The second objective of the stowage plan is to minimize the number of yard crane movements, and this objective may be expressed as:

$\text{Obj}_2 = \sum_s \left[ If \left( \sum_{ij} X_{ijs} \cdot YB_i - \sum_{ij} X_{ij(s-1)} \cdot YB_i \right) \neq 0, 1, 0 \right]$  \hspace{1cm} (7)

(3) Minimization of the sum of weight differences between stacked container pairs

$\text{Obj}_3 = \sum_j \left( \sum_{is} X_{ijs} \cdot G_i - F_j \right)$ \hspace{1cm} (8)

(4) Overall objective function

$T_{\text{obj}} = \alpha \cdot \text{Obj}_1 + \beta \cdot \text{Obj}_2 + \gamma \cdot \text{Obj}_3$ \hspace{1cm} (9)

In this equation, $\alpha, \beta$, and $\gamma$ are weight coefficients, such that $\alpha + \beta + \gamma = 1$.

3.4. Linear transformations. In the following text, auxiliary decision-making variables and auxiliary constraints will be introduced in $\text{Obj}_1$ and $\text{Obj}_2$, and forced linear transformations will be performed.

For $\text{Obj}_1$,

$\text{Obj}_1 = \sum_{ik} YP_{ik} \cdot \left[ If \left( \sum_{js} X_{ijs} \cdot Q_s - \sum_{js} X_{kjs} \cdot Q_s \right) > 0, 1, 0 \right]$  

(1) We introduce the intermediary variable, $M_{1ik}$ and define this as:

$M_{1ik} = \sum_{js} X_{ijs} \cdot Q_s - \sum_{js} X_{kjs} \cdot Q_s$

$M_{1ik}$ refers to the relative values of the sequence numbers of containers $i$ and $k$.

(2) Next, we introduce the auxiliary decision-making variable, $F_{1ik}$, which is a binary:

$F_{1ik} = \left[ If \left( \sum_{js} X_{ijs} \cdot Q_s - \sum_{js} X_{kjs} \cdot Q_s \right) > 0, 1, 0 \right]$  

or

$F_{1ik} = \begin{cases} 1, & \sum_{js} X_{ijs} \cdot Q_s - \sum_{js} X_{kjs} \cdot Q_s > 0 \\ 0, & \sum_{js} X_{ijs} \cdot Q_s - \sum_{js} X_{kjs} \cdot Q_s < 0 \end{cases}$

If the sequence number of $i$ is larger than the sequence number of $k$, $i$ is dispatched after $k$, and $F_{1ik} = 1$. Otherwise, $F_{1ik} = 0$. 

(3) The auxiliary constraint, \( CF1 \) is then introduced, and this constraint may be expressed as:

\[
\left( \sum_{js} X_{ijs} \cdot Q_s - \sum_{js} X_{kjs} \cdot Q_s \right) / 1000 \leq F_{1ik}
\]  (10)

or:

\[
M_{1ik} / 1000 \leq F_{1ik}
\]

The left term in this inequality represents a one-dimensional array for \( i \) and \( k \). Since the sequence number values of \( i \) and \( k \) are much smaller than 1000, the division of this difference by 1000 results in a one-dimensional array with element values that vary between 0 – 1.

Therefore, \( \text{Obj1} \) may be expressed as:

\[
\text{Obj1} = \sum_{ik} YP_{ik} \cdot F_{1ik}
\]  (11)

Similarly, \( \text{Obj2} \) may be expressed as

\[
\text{Obj2} = \sum_s \left[ \text{If} \left( \sum_{ij} X_{ijs} \cdot YB_i - \sum_{ij} X_{ij(s-1)} \cdot YB_i \right) \neq 0, 1, 0 \right]
\]

(1) Here, we introduce the intermediary variable, \( M_{2s} \), which is defined as

\[
M_{2s} = \sum_{ij} X_{ijs} \cdot YB_i - \sum_{ij} X_{ij(s-1)} \cdot YB_i
\]

\( M_{2s} \) refers to the relationship between the yard-bays corresponding to the containers with dispatch sequence numbers \( s \) and \( (s-1) \).

(2) Next, we introduce the auxiliary decision-making variable \( F_{2s} \), which is a binary:

\[
F_{2s} = \left[ \text{If} \left( \sum_{ij} X_{ijs} \cdot YB_i - \sum_{ij} X_{ij(s-1)} \cdot YB_i \right) \neq 0, 1, 0 \right]
\]

that is:

\[
F_{2s} = \begin{cases} 1 & \text{If} \left( \sum_{ij} X_{ijs} \cdot YB_i - \sum_{ij} X_{ij(s-1)} \cdot YB_i \right) \neq 0 \\ 0 & \text{If} \left( \sum_{ij} X_{ijs} \cdot YB_i - \sum_{ij} X_{ij(s-1)} \cdot YB_i \right) = 0 \end{cases}
\]

If the yard-bays corresponding to the containers with dispatch sequence numbers \( s \) and \( (s-1) \) are different, then \( F_{2s} = 1 \); if they are the same, then \( F_{2s} = 0 \).

(3) The auxiliary constraint, \( CF2 \), is then introduced, and this constraint may be expressed as:

\[
\left( \sum_{ij} X_{ijs} \cdot YB_i - \sum_{ij} X_{ij(s-1)} \cdot YB_i \right) / 1000 \leq F_{2s}
\]  (12)

that is:

\[
M_{2s} / 1000 \leq F_{2s}
\]
Therefore, \( Obj2 \) may be expressed as:

\[
Obj1 = \sum_s F2_s
\]  

(13)

3.5. **Summarization of the model.** In summary, the stowage model for export containers in a single bay may be treated as a multi-objective integer programming (MIP) model. This model may then be summarized as follows:

Minimize:

\[
\begin{align*}
    T_{\text{obj}} &= \alpha \sum_{ik} YP_{ik} \left[ \text{If} \left( \sum_{js} X_{ij} * Q_s - \sum_{js} X_{kj} * Q_s \right) > 0, 1, 0 \right] \\
    &+ \beta \sum_s \left[ \text{If} \left( \sum_{ij} X_{ij} * YB_i - \sum_{ij} X_{ij(s-1)} * YB_i \right) \neq 0, 1, 0 \right] \\
    &+ \gamma \sum_j \left( \sum_{is} X_{ij} * G_i - F_j \right)
\end{align*}
\]  

(14)

Subject to:

\[
\begin{align*}
    \sum_{j \in J} \sum_{s \in S} X_{ij} &= 1 \quad \forall i \\
    \sum_{i \in I} \sum_{s \in S} X_{ij} &= 1 \quad \forall j \\
    \sum_{i \in I} \sum_{j \in J} X_{ij} &= 1 \quad \forall s \\
    \sum_{j \in J} X_{ij} * A_{jr} &\leq GP_r \quad \forall r \\
    W_j &\leq \sum_{is} X_{ij} * G_i \leq M_j \quad \forall j \\
    (\sum_{is} X_{ij} * G_i - \sum_{is} X_{il} * G_i) * V P_{jl} &\leq D_{jl} \quad \forall j, l \\
    \sum_{is} X_{ij} * Q_s &\leq \sum_{i} \left( \sum_{is} (X_{ij} * Q_s) * V P_{jl} \right) \quad \forall j \\
    (\sum_{js} X_{ij} * Q_s - \sum_{js} X_{kj} * Q_s) / 1000 &\leq F1_{ik} \quad \forall i, k \\
    (\sum_{ij} X_{ij} * YB_i - \sum_{ij} X_{ij(s-1)} * YB_i) / 1000 &\leq F2_s \quad \forall s
\end{align*}
\]  

(15)

4. **Solving the AIMMS decision-making model and the analysis of example calculations.** The test bed for performing the example calculations of this model uses the Windows 7 64 bit operating system. The hardware is comprised of an Intel Core i5-4200U (1.6GHz/L3 3M) CPU running at 1.6 GHz, and 4GB of DDR3L 1600 RAM.

The change in computational time and memory usage with respect to the number of variable nodes were plotted as line charts to facilitate comparisons in computational efficiency, as shown in Figure 8. The analysis of the example calculations and systematic tests thus demonstrate that the AIMMS decision-making system performs very well, as it is able to maintain relatively short computational times even with large numbers of variable nodes, and only requires a small amount of
Table 1. Computational efficiency tests

| Test Number | Containers to be Stowed | Num. of Ship Slots | Num. of Sequences | Num. of Variable Nodes | Solution Time(s) | Memory Usage(M) |
|-------------|-------------------------|--------------------|-------------------|------------------------|------------------|-----------------|
| 1           | 5                       | 5                  | 5                 | 125                    | 0.1              | 0.9             |
| 2           | 10                      | 10                 | 10                | 1000                   | 0.2              | 1               |
| 3           | 15                      | 15                 | 15                | 3375                   | 0.3              | 1.1             |
| 4           | 20                      | 20                 | 20                | 8000                   | 0.5              | 1.1             |
| 5           | 25                      | 25                 | 25                | 15625                  | 0.6              | 1.2             |
| 6           | 30                      | 30                 | 30                | 27000                  | 0.9              | 1.4             |
| 7           | 35                      | 35                 | 35                | 42875                  | 1.2              | 1.7             |
| 8           | 40                      | 40                 | 40                | 64000                  | 1.4              | 2               |
| 9           | 45                      | 45                 | 45                | 91125                  | 1.9              | 2.4             |
| 10          | 50                      | 50                 | 50                | 125000                 | 2.5              | 2.8             |

Figure 8. Line chart for comparing computational efficiency

memory. Hence, this system may be used to solve actual stowage problems for the loading of single-bay decks.

5. Conclusions. The container stowage problem is a multi-objective combinatorial optimization problem with complicated constraints, and “combinatorial explosion” tends to render conventional optimization methods unsuited for this problem. To solve this problem, new optimization methods need to be investigated, as they will have broad practical implications and applicative value for the modernization of container terminal management and container ship transportation. The container ship bay stowage problem in automated terminals was studied in this work, and we have constructed a multi-objective mathematical optimization model for the intelligent stowage of export containers in a single bay based on the AIMMS decision-making support system. The findings of this study will provide highly viable, effective and high-quality solutions for actual stowage planning operations.
Acknowledgment. This research was supported by the National Natural Science Foundation of China (No.61540045), the Science and Technology Commission of Shanghai Municipality (No.15YF1404900, No.14170501500), Ministry of Education of the PR China (No.20133121110005), Shanghai Municipal Education Commission (No. 14ZZ140), Shanghai Maritime University (No.2014ycx040).

REFERENCES

[1] U. d. G. V. V. . G. . I. Ambrosino Daniela (Dipto. di Econ. e Metodi Quantit., A. Sciomachen and E. Tanfani, Stowing a containership: The master bay plan problem, Transportation Research Part A: Policy and Practice, 25 (2004), 81–99.
[2] M. Avriel and M. Penn, Exact and approximate solutions of the container ship stowage problem, Computers & Industrial Engineering, 25 (1993), 271–274.
[3] M. Avriel, M. Penn, N. Shipirer and S. Witteboon, Stowage planning for container ships to reduce the number of shifts, Annals of Operations Research, 55–71.
[4] M. Bonamy, B. Lévêque and A. Pinlou, Planar graphs with $\geq 7$ and no triangle adjacent to a $c$-4 are minimally edge and total choosable, Computer Science, 17 (2016), 131–145.
[5] R. C. Botter and M. A. Brinati, Stowage container planning: A model for getting an optimal solution, 1992, 217–229.
[6] J. Y. Cai, V. T. Khodyreva, E. A. and A. N. Khuziakhmetov, Higher education curricula designing on the basis of the regional labour market demands, Eurasia Journal of Mathematics Science & Technology Education, 13 (2017), 2805–2819.
[7] L. Chen and Z. Lu, The storage location assignment problem for outbound containers in a maritime terminal, International Journal of Production Economics, 135 (2012), 73–80.
[8] Y. Chen, Z. Liu, B. Liu and X. Fu, Research on a fast human-detection algorithm for unmanned surveillance area in bulk ports, 2014, 1–17.
[9] E. Elsborg, T. T. Hildebrandt and D. Sangiorgi, A constraint programming model for fast optimal stowage of container vessel bays, Trustworthy Global Computing, 220 (2012), 251–261.
[10] H. S. Hwang and G. S. Cho, A hybrid geneticalgorithm with a new packing strategy for the three-dimensional bin packing problem, Applied Mathematics and Computation, 219 (2012), 1287–1299.
[11] A. Imai, E. Nishimura, S. Papadimitriou and M. Liu, Multi-objective simultaneous stowage and load planning for a container ship with container rehandle in yard stacks, European Journal of Operational Research, 171 (2006), 373–389.
[12] L. Junqueira, R. Morabito and D. S. Yamashita, Three-dimensional container loading models with cargo stability and load bearing constraints, Computers and Operations Research, 39 (2012), 74–85.
[13] K. H. Kim, J. S. Kang and K. R. Ryu, A beam search algorithm for the load sequencing of outbound containers in port container terminals, Or Spectrum, 26 (2004), 93–116.
[14] Y. Lee and N. Y. Hsu, A heuristic for retrieving containers from a yard, Computers & Operations Research, 37 (2010), 1139–1147.
[15] C. Mi, Y. Shen, W. Mi and Y. Huang, Ship identification algorithm based on 3d point cloud for automated ship loaders, Journal of Coastal Research, 73 (2015), 28–34.
[16] C. Mi, Z. Zhang, Y. Huang and Y. Shen, A fast automated vision system for container corner casting recognition, Journal of Marine Science & Technology, 24 (2016), 54–60.
[17] J. Ren, Y. Tian and T. Sawaragi, A tree search method for the container loading problem with shipment priority, European Journal of Operational Research, 526–535.
[18] M. A. Salido, M. Rodriguez-Molins and F. Barber, A decision support system for managing combinatorial problems in container terminals, Knowledge-Based Systems, 29 (2012), 63–74.
[19] A. Sciomachen, M. Acciaro and M. Liu, Department of economics and quantitative methods (diem), university of genova, via vivaldi 5,16126 genova, italy; a 3d-bpp approach for optimising stowage plans and terminal productivity, European Journal of Operational Research, 183 (2007), 1433–1446.
[20] M. Shaabani and L. Pourmemar, On the definition of a probabilistic normed group, Journal of Interdisciplinary Mathematics, 20 (2007), 383–395.
[21] L. B. Si and H. Y. Qiao, Evaluation of technological innovation efficiency in equipment manufacturing industry based on input orientation–panel data analysis based on data envelopment model, *Journal of Discrete Mathematical Sciences & Cryptography*, 20 (2017), 1381–1386.

[22] I. D. Wilson and P. A. Roach, Principles of combinatorial optimization applied to container-ship stowage planning, *Journal of Heuristics*, 5 (1999), 403–418.

[23] I. D. Wilson and P. A. Roach, Department of mathematics, university of glamorgan, pontypridd, mid glamorgan cf37 1dl, united kingdom; container stowage pre-planning: Using search to generate solutions, *Knowledge-Based Systems*, 14 (2001), 137–145.

[24] T. Winter, Online and real-time dispatching problems. *Ph.D. Thesis*, Technical University of Braunschweig, Germany, 1999.

[25] S. Q. Wu, Models for evaluating the technological innovation capability of small and micro enterprises with hesitant fuzzy information, *Journal of Intelligent & Fuzzy Systems: Applications in Engineering and Technology*, 32 (2016), 249–256.

Received June 2017; revised November 2017.

E-mail address: 1317021038@qq.com
E-mail address: mwj@shmtu.edu.cn