Including Containers with Dangerous Goods in the Slot Planning Problem

Kiros Gebrearegawi Kebedow
School of Mathematical and Statistical Sciences
Hawassa University
Hawassa, Ethiopia
kirosmaths@gmail.com

Johan Oppen
Molde University College, Specialized University in Logistics
Molde, Norway
johan.oppen@himolde.no

Abstract

Container stowage problems are rich optimization problems with both high economic and environmental impact. These problems are typically decomposed into a master bay planning phase, which distributes containers to bay sections of the vessel, and a slot planning phase, which assigns a specific slot within the bay section to each container. In this paper, we extend existing models for slot planning by considering containers with dangerous goods. An important contribution of this paper is that we provide a model closer to the real-world problems faced by planners, and thus solutions based on this model should be easier to implement in practice. We show that our model can be solved to optimality in reasonable time using standard software like Gurobi or CPLEX.

Keywords
Operations research, Container stowage, Optimization, Logistics

1. Introduction

During the last six decades, the use of containers has increased the efficiency of worldwide trade. According to Steenken et al. (2004) over 60% of the world's deep-sea general cargo is transported in containers, and the routes between some countries are containerized up to 90%. The majority of containers are transported by vessels called *container ships*, which are specifically designed to carry containers. Apart from the few years of financial crisis, the shipping industry has had a continuous growth (Christensen and Pacino, 2017). With the increasing level of world trade and the corresponding increase in size of container ships (up to around 21,000 containers (Wikipedia, 2017)) one of the main areas for cost and efficiency gains in the industry is to have an efficient stowage plan to facilitate fast container loading and unloading. However, because of its combinatorial nature and the various operational constraints related to both the ship structure and container properties, stowage planning is a complex problem. To handle the complexity of the stowage planning problem, we use an approach similar to some of the recently most successful approaches, see, e.g., Pacino et al. (2011), Pacino et al. (2012), Kang and Kim (2002), Wilson and Roach (2000), Wei-ying et al. (2005), which decompose the problem into two phases: the *Master Bay Planning Problem* and the *Slot Planning Problem*. The Master Bay Planning Problem distributes the containers to bay sections of the vessel, the Slot Planning Problem then assigns the containers in each bay section to specific slots. In this paper, we develop a new and more realistic binary programming model for the Slot Planning Problem.

The main contributions of our paper is that we include containers with dangerous goods, so-called *IMO containers* (IMO - International Maritime Organization), in our model, and that we define the decision variables in a more realistic way than what has been done before. In addition, we show that, despite these extensions, the model can be solved to optimality in reasonable time. According to Delgado et al. (2012), one second is a reasonable runtime for slot planning problems.

The remainder of the paper is organized as follows: in Section 2 we provide a description of the Slot Planning Problem. Section 3 gives a brief literature review. We present computational results in Section 4, and Section 5 concludes the paper.

2. Problem Description

A *container vessel* is a cargo ship designed to transport containerized cargo. The vessel's cargo space is divided into compartments called *bays*; each bay is partitioned into on deck and below deck parts using *hatch covers*, which are metallic, flat, waterproof structures, which allow containers to be stored on top of them. The under deck part of the ship is physically divided into several cargo holds by upright walls called *bulkheads*, which are indicated by bold vertical lines in Figure 1. Moreover, both the on deck and below deck parts of a bay is transversally partitioned into stacks that are one container wide, and contain two *Twenty foot Equivalent Unit* (TEU) stacks or a single *Forty foot Equivalent Unit* (FEU) stack. A *location* is a bay section which consists of a set of stacks that are either on or below deck, the stacks are adjacent, and coincide with the hatch covers. Figure 2 shows a typical arrangement of locations in a bay. Stacks have maximum height and weight limits. Vertically, cells in a stack are indexed by tiers. The cells in a stack are divided into two *slots*, a *fore* and an *aft* slot.

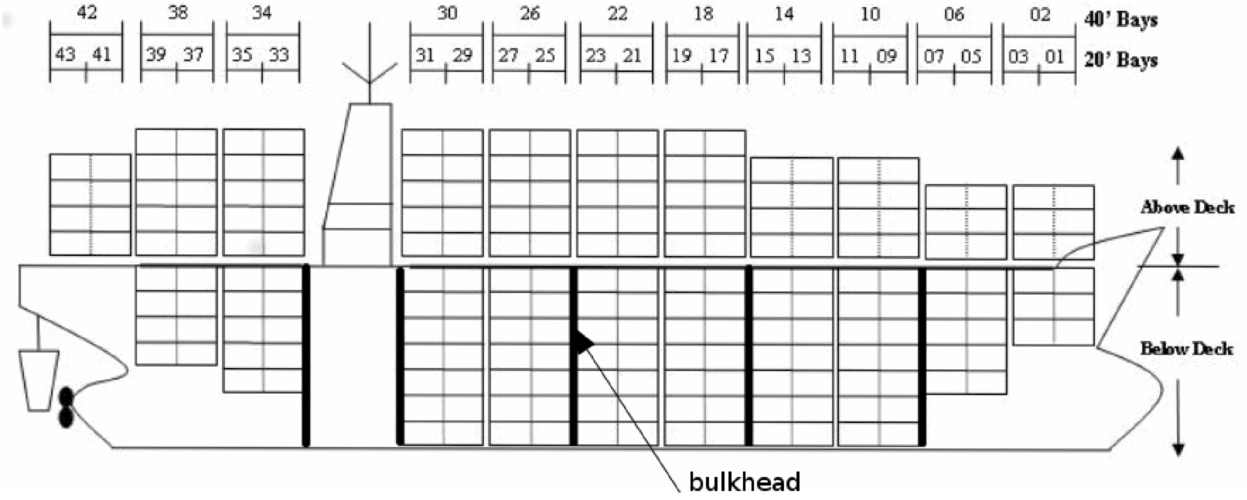
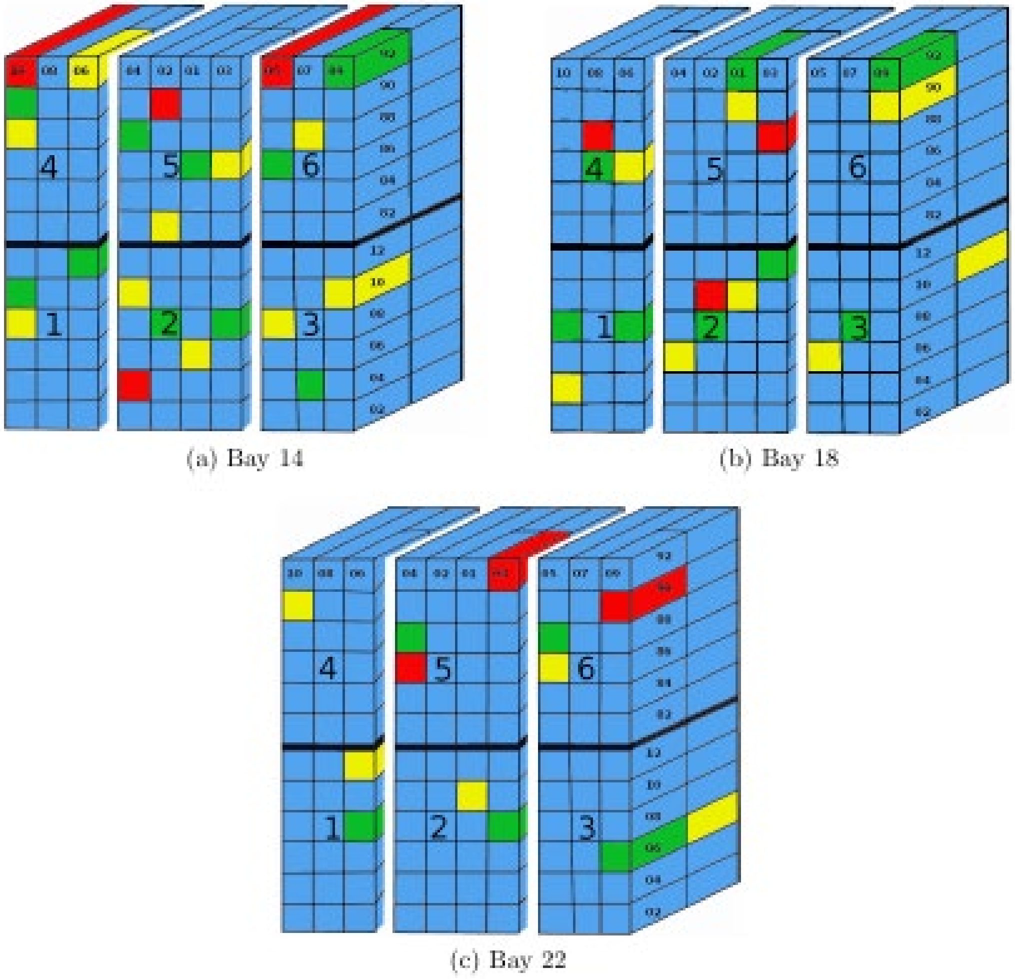


Figure 1. Layout of a container ship, adopted from Fan et al. (2010)

A *container* is a metal box in which goods can be stored. The most commonly used container heights are 0', 8'6", and 9'6", and lengths are 20', 40', and 45'. A 0' high container is called a platform whereas a 9'6" container is called a high-cube container. High-cube 20' containers are rare (Delgado et al., 2012), and we assume that they do not exist when modeling the Slot Planning Problem. Reefer containers are containers that should be kept cool and must be supplied with electricity. Containers carrying dangerous goods are called IMO containers.

Most cells can hold one 40' or 45' container or two 20' containers. 45' containers can only be loaded in cells meant for such containers or on the upper tier (on deck) of an FEU bay, and some cells may be restricted to either 20' or 40' containers (Delgado et al., 2012). Due to the physical layout of the vessel, there may exist odd cells that can hold only one 20' container. *Loaded containers* are containers already on board a vessel when the stowage plan is made. The list of containers to load in the current port is called the *load list*. It contains detailed information about the containers, such as height, length, weight, type (standard, reefer, IMO type), and discharge port. A *release* is a list of loaded containers; it has the same container information as the load list and the exact position where each container is stowed. When a container is unloaded, the containers above it in the same stack must be unloaded first. Moreover, if the container is stowed below a hatch, all containers above this hatch must also be unloaded in order to open the hatch. In stowage planning, a common situation is that, at port *i*, a container with port of destination (POD) *j* after port *i* must be unloaded and reloaded at port *i* in order to access the container below it with POD *j*. This stowage configuration is called *over-stow*. Over-stow requires additional crane movement, which costs both time and money and may lead to an increase in the container ship berthing time. This may lead to increased speed at sea, which consumes more bunker and releases more Co_2



(a) Bay 14

(b) Bay 18

(c) Bay 22

Figure 2. A container stowage configuration for two groups of incompatible IMO containers following principle 2 (marked with yellow and green) and one group following principle 3 (in red) in three adjacent even bays. The setup is based on figure 1, thus there is a bulkhead between bays 14 and 18.

2.1 IMO classes and stowage rules

According to the International Maritime Dangerous Goods Code (IMDG Code), there are nine classes of dangerous goods; some of these are divided into subclasses, currently making up 17 classes. Detailed information can be found in Ambrosino and Sciomachen (2015). Due to their properties, many of these substances are incompatible to each other, and thus a minimum distance has to be kept between them. For this purpose, the IMDG Code provides four segregation rules for IMO containers: (1) Away from, (2) Separated from, (3) Separated by a complete compartment from, and (4) Separated longitudinally by an intervening complete compartment or hold from. Principle (1) only affects the stowing of open top containers, which are not included in this paper, thus only principles 2-4 are considered here. Each class of dangerous goods is incompatible to the other classes following one of the principles referred to above. For example, assume that c_1, c_2, c_3, c_4 are IMO containers of different classes and $C_{c_1}^2 = \{c_2, c_3\}, C_{c_1}^3 = \{c_4\}$ are incompatible containers, meaning that containers in $C_{c_i}^j$ are incompatible with container *i* following principle *j*. Looking at our example, this means that c_1 and c_2 contain goods that need to be separated according to principle 2, while c_1 and c_4 contain goods that need to be separated according to principle 3. IMO containers following principle 3 in under deck locations and IMO containers following principle 4 in both on deck and under deck locations require a complete compartment separation, so they are handled in Master Bay Planning. IMO containers following principle 2 in both on deck and under deck locations and IMO containers following principle 3 in on deck locations are handled in both Master Bay Planning and the Slot Planning Problem. The stowage rules (IMO, 2006) for principle 2 and principle 3 are described as follows.

Two IMO containers following segregation principle 2 cannot be placed in the same stack (row), unless separated by a deck, while they can be stowed horizontally (in both longitudinal and transversal directions) separated by one container space. If there is a bulkhead between them, one container space is not necessary. For instance, given two incompatible IMO containers following principle 2, if one container is stowed in bay 15, row 09, tier 10, the other can be stowed in bay 17, row 09, tier 10, as illustrated in Figure 2. Note that there is a bulkhead between bay 14 and bay 18.

Two IMO containers following segregation principle 3 on deck cannot be placed in the same stack, while they can be stowed longitudinally separated by one container space and transversally separated by two container spaces. For example, given two incompatible IMO containers following principle 3, if one container is stowed in bay 19, row 03, tier 88, the other cannot be stowed in on-deck locations of rows 02, 01, 03, 05, or 07 of bays 17, 19 or 21 as illustrated in Figure 2. The separation rules described above are summarized in Table 1, which is taken from (IMO, 2006).

Table 1. Segregation principles for IMO containers

Principle	Vertical	Horizontal		
			On deck	Under deck
1	One on top of other allowed	Fore and aft	No restriction	No restriction
		Athward ships	No restriction	No restriction
2	Not in the same vertical line unless segregated by deck	Fore and aft	One container space	One container space or bulkhead
		Athward ships	One container space	One container space
3	Not in the same vertical line unless segregated by deck	Fore and aft	One container space	One bulkhead
		Athward ships	One container space	One bulkhead

In this paper, we focus on the Slot Planning Problem (SPP). We solve the SPP for all on deck or all under deck locations of a given bay, in order to avoid incompatibilities between neighboring stacks. In addition, the solution from the previous bay may restrict the placement of IMO containers in the current bay. The SPP modelled and solved in this paper includes the stacking rules for 20', 40'/45' containers, reefer containers, IMO containers, loaded containers, weight and height constraints, and port of destination of containers for both under deck and on deck locations. In practice, 40' containers cannot be stowed on top of 45' containers, and 45' containers cannot be placed in all cells that can take a 40' container, but we have chosen to leave these constraints out of the model, as we believe they only make the notation and model grow without adding much in terms of insights. This means we only consider 20' and 40' containers from now on.

The objective function in our version of the SPP is the same as the objective function in Delgado et al. (2012), and reflects rules of thumb from stowage coordinators in order to produce stowage plans that are robust to changes in forecasted demands. The objective includes maximizing the number of empty stacks, placing containers with the same discharge port in the same stack, minimizing the number of reefer slots used for non-reefer containers, and minimizing over-stowage within stacks. A more detailed description of the objective function can be found in (Delgado et al., 2012). The constraints included in our model are described below.

1. A container must be placed in one cell only and a cell can only hold one 40' container or two 20' containers.
2. A 20' container cannot be placed on top of a 40' container.
3. A container must have support from below (it cannot be placed “on air”).
4. Stacks on deck, which are more than five tiers high, cannot be more than one tier higher than neighboring stacks.
5. For each stack, the total weight and height of the containers must be within the weight and height limits of the stack.

6. A heavy container cannot be stacked on top of a light container (the container weight must be decreasing with stack height).
7. A 20' reefer container must be placed in a reefer slot. A 40' reefer container must be placed in a cell that has at least one reefer slot.
8. All IMO containers must be stacked according the segregation rules given in Table 1.
9. The length constraint of a cell must be satisfied (for example, a cell which can hold only one 20' container must be occupied only by a container of 20' length).
10. All loaded containers must be stowed in their original slots and they cannot be swapped to any other slots.
11. All containers must be stowed in stacks of their respective location.
12. A cell with two slots must have either both slots occupied or both slots empty.

2.2 Mathematical Model

We have developed a linear binary programming model for the SPP, which is an extension of the work of Delgado et al. (2012) and Parreño et al. 2016). The notation and the formulation of the model covers about six pages, so we have chosen not to include it in this paper, and we direct interested readers to any of the authors for details.

3. Literature review

The number of papers dealing with stowage planning has grown significantly within the last few decades. The contributions can be divided into single phase and multi-phase approaches. Single phase approaches represent the stowage planning problem in a single optimization model, whereas multi-phase approaches decompose the problem hierarchically into two or more phases.

As discussed before, the decomposition approaches (Wilson and Roach (2000), Wilson et al. (2001), Kang and Kim (2002), Wei-ying et al. (2005), Pacino et al. (2011), Pacino et al. (2012), Ambrosino et al. (2015a), Ambrosino et al. (2015b)) are currently the most successful to solve stowage planning problems. In decomposition approaches, a stowage planning problem is typically divided into a Master Bay Planning Problem and a Slot Planning Problem. Optimization models developed for Master Bay Planning problems contain high-level constraints, mainly regarding the stability of the vessel, whereas optimization models for Slot Planning Problems contain low-level constraints, mainly concerning the stacking rules. For instance, Pacino et al. (2015) solved the stowage planning problem by decomposing the problem hierarchically into two phases: the Multi Port Master Bay Planning Problem and the Slot Planning Problem. In the MP-MBPP, an integer programming model that assigns groups of different container types (standard light, standard heavy, reefer light and reefer heavy) to locations of the vessel was presented. Then a constraint based local search procedure for stowing the individual containers into slots was introduced. A matheuristic for the cargo mix problem with block stowage in the master planning phase, based on a partial variable fixing procedure, was presented in (Christensen and Pacino, 2017).

Within the group of single phase approaches, binary integer programming models have been developed by Ambrosino et al. (2004) and Li et al. (2008). The model in (Ambrosino et al., 2004) considers 20' and 40' standard containers with three weight classes. Over-stowage is modeled as constraints rather than as part of the objective, and containers with special requirements, such as reefer and IMO containers, are not included. The model considers only the current port, and the objective is to minimize berthing time. To solve the problem the authors presented a heuristic approach that enables them to relax some constraints from the model and give some pre-stowage rules. Li et al. (2008) proposed a binary integer programming model for the multi-port stowage planning problem, aimed to determine the exact location of each container. Similar to (Ambrosino et al., 2004), only standard containers of length 20' and 40' were considered. This work, however, did consider over-stowage as part of the objective, but did not consider weight limitations for individual containers. The authors tested the model with different instances of a small vessel of 800 TEUs capacity.
The single phase approaches have also been solved using constraint programming (Ambrosino and Sciomachen, (1998), Delgado et al. (2012)) and different types of heuristics (Avriel et al. (1998), Ding and Chou (2015), Dubrovsky et al. (2002), Pacino and Jensen (2012)).

To the best of the authors' knowledge, the only three papers which propose a Slot Planning Problem similar to our definition are Delgado et al. (2012), Pacino and Jensen (2012) and Parreño et al. (2016). The model presented in Delgado et al. (2012) does not take into consideration IMO containers, neither does the constraint based local search approach in Pacino and Jensen (2012). Parreño et al. (2016) introduced a binary programming model and a GRASP algorithm for solving the SPP, including standard containers, high cube containers, reefer and IMO containers. This paper seems to be the only one where IMO containers are included, but the IMO segregation rules are only applied within the stacks of the locations under consideration. However, IMO segregation rules also affect IMO containers in nearby locations. Due to this, we believe their model might lead to infeasible solutions when implemented for a real world case.

4. Computational results

We have tested our model on 304 instances, 174 contain under deck locations and 130 contain on deck locations. An overview of the instance characteristics can be found in Table 2. We are aware of the instances used in Delgado et al. (2012), but, because we define a location to be the entire on deck or under deck part of a bay, these cannot be used in our model directly. We also seem to interpret and handle the IMO rules differently, and we therefore generate our own instances from solutions for the Multi Port Master Bay Planning Problem. The MPMBP instances are randomly generated, but they have been made deliberately to correspond closely to real world stowage problems.

The purpose of the computational experiments described here is twofold: we want to ensure that our model is "correct" in the way that it produces solutions according to the problem description in Section 2, and we want to find out if problem instances of realistic size can be solved to optimality in reasonable time. Delgado et al. (2012) recommend one second as a reasonable runtime to solve the SPP. Our SPP instances correspond to two SPPs in Delgado et al. (2012), thus two seconds would be a reasonable solution time. Table 2 gives an overview of the instances used, with column headers as follows. ID identifies a set of instances, #Inst is the number of instances in the given set. The columns under Length tell if at least one instance has the given length or not. Under/On tells if the instances contain under deck or on deck locations. Reefer indicate whether at least one container in at least one instance of the given set is a reefer container. Principle 2 and 3 indicate whether at least one instance of the given set contain containers following the respective principle. The # Containers to be loaded columns give the maximum, minimum and average number of containers to be loaded in each instance, and finally # Ports gives the maximum number of ports of destination.

Table 2. Overview of instances.

		Length				Principle		# Containers to be loaded			
ID	# Inst	20'	40'	Under/On	Reefer	2	3	Max	Min	Aver	# Ports
1	47	✓	✓	Under	✓	✓		78	21	43	3
2	50	✓	✓	On		✓	✓	70	19	42.22	3
3	46	✓	✓	Under	✓			83	19	40.7	3
4	49	✓	✓	On				85	13	48.29	3
5	50	✓	✓	Under	✓			86	20	48.74	3
6	11	✓	✓	Under	✓	✓		90	26	61.27	3
7	11	✓	✓	On		✓	✓	94	34	58.45	3
8	20	✓	✓	Under	✓	✓		84	44	45.7	3
9	20	✓	✓	On		✓	✓	65	6	42.8	3

Table 3. SPP results.

	Runtime			# Inst solved	
ID	Max	Min	Aver	≤ 1 sec	> 1 sec
1	2.0	0.01	0.79	35	12
2	2.3	0.03	0.52	47	3
3	2.0	0.01	0.59	42	4
4	2.5	0.05	0.52	42	7
5	2.5	0.06	0.90	38	12
6	2.2	0.15	0.85	8	3
7	0.4	0.03	0.17	11	0
8	3.0	0.03	0.54	17	2
9	1.2	0.05	0.48	19	1

All experiments have been run on a Linux machine with Intel Core i7-5600U, CPU 2.60GHz x4 and 16 GB of memory. The model was implemented in Pyomo and solved with Gurobi 7.0, which is a state-of-the-art linear solver. Since the model is linear, all solutions are globally optimal. The results given in Table 3 show that most of the instances are solved to optimality within one second. There are a few instances in almost all sets which have a runtime between two and three seconds. This means we are very close to being able to solve realistically sized instances in reasonable time. If all SPP instances for a container vessel are solved sequentially in one operation, a runtime of between two and three seconds for some instances will not be a problem, as the average runtime is well below one second.

5. Conclusions

In this paper, we have extended known models for the Slot Planning Problem to include proper handling of IMO containers according to segregation rules for such containers. The problem has been rigorously described, and is formulated as a pure binary programming problem. Computational experiments show that the model can be solved to optimality in reasonable time using standard commercial software.

References

Ambrosino, D., and Sciomachen, A., Using a Bin Packing Approach for Stowing Hazardous Containers into Containerships. In: *Optimization and Its Applications*. Springer, pp 1-17, 2015.

Ambrosino, D., Sciomachen, A., and Tanfani, E., Stowing a containership: the master bay plan problem, *Transportation Research Part A: Policy and Practice,* vol. 38, no. 2, pp 81-99, 2004.

Ambrosino D., Paolucci, M., and Sciomachen A., Experimental evaluation of mixed integer programming models for the multi-port master bay plan problem, *Flexible Services and Manufacturing Journal,* vol. 27, 263—284, 2015a.

Ambrosino D., Paolucci M., and Sciomachen A., Computational evaluation of a MIP model for multi-port stowage planning problems, *Soft Computi*ng, vol. 21, no. 7, pp1753-1763, 2015b.

Ambrosino, D., and Sciomachen, A., A constraint satisfaction approach for master bay plans, *Transactions on the Built Environment*, vol. 36, pp 175--184. 1998.

Avriel, M., Penn, M., Shpirer, N., and Witteboon, S., Stowage planning for container ships to reduce the number of shifts, *Annals of Operations Research,* vol. 76, pp. 55-71, 1998.

Christensen, J., and Pacino, D., A matheuristic for the Cargo Mix Problem with Block Stowage, *Transportation Research Part E*, vol. 97, pp. 151-171, 2017.

Delgado, A., Jensen, R. M., Janstrup, K., Rose, T. H., and Andersen, K. H., A Constraint Programming model for fast optimal stowage of container vessel bays, *European Journal of Operational Research,* vol. 220, pp. 251-261, 2012.

Ding, D., and Chou, M. C., Stowage planning for container ships: a heuristic algorithm to reduce the number of shifts, *European Journal of Operational Research,* vol. 246, no. 1, pp. 242-249, 2015.

Dubrovsky, O., Levitin, G., and Penn, M., A genetic algorithm with a compact solution encoding for the container ship stowage problem, *Journal of Heuristics,* vol. 8, no. 6, pp. 585-599, 2002.

Fan, L., Low, M. Y. H., Ying, H. S., Jing, H. W., Min, Z., and Aye, W. C., Stowage planning of large container-ship with tradeoff between crane workload balance and ship stability, *Proceedings of the International Multi-Conference of Engineers and Computers Scientists*, vol. 3, pp. 1-7, 2010.

Kang, J. G., and Kim, Y. D., Stowage planning in maritime container transportation, *Journal of the Operational Research Society,* vol. 53, no. 4, pp. 415-426, 2002.

Li, F., Tian, C., Cao, R., and Ding, W. An Integer Linear Programming for Container Stowage Problem. In: Bubak, M., van Albada, G. D., Dongarra, J., and Sloot, P. M. A., (eds) *Computational science - ICCS 2008. Lecture notes in computer science*, vol. 5101, Springer, Berlin, Heidelberg, 2008.

Pacino, D., Delgado, A., Jensen, R. M., and Bebbington, T., Fast generation of near-optimal plans for eco-efficient stowage of large container vessels, *Proceedings of the International Conference on Computational Logistics,* pp. 286-301, 2011.

Pacino, D., Delgado, A., Jensen, R. M., and Bebbington, T., An accurate model for seaworthy container vessel stowage planning with ballast tanks, *Proceedings of the International Conference on Computational Logistics,* pp. 17-32, 2012.

Pacino, D., and Jensen, R. M., Constraint-based local search for container stowage slot planning, *Proceedings of the International Multi-conference of Engineers and Computer Scientists,* pp. 1467-1472, 2012.

Parreño, F., Pacino, D., and Alvarez-Valdes, R., A GRASP algorithm for the container stowage slot planning problem, *Transportation Research Part E*, vol. 94, pp. 141-157, 2016.

Steenken, D., Voss, S., and Stahlbock, R., Containerterminal operation and operations research – a classification and literature review, *OR Spectrum,* vol. 26, no 1, pp. 3-49, 2004.

Wilson, I. D., and Roach, P. A., Container stowage planning: a methodology for generating computerised solutions, *Journal of the Operations Research Society,* vol. 51, no. 11, pp. 1248-1255, 2000.

Wilson, I. D., Roach, P. A., and Ware, J. A., Container stowage pre-planning: using search to generate solutions, a case study, *Knowledge-Based Systems,* vol. 14, no. 3, pp. 137-145, 2001.

Wei-ying, Z., Yan, L., and Zhou-shang, J., Model and algorithm for container ship stowage planning based on bin packing problem, *Journal of Marine Science and Application,* vol. 4, no. 3, pp. 30-36, 2005.

Wikipedia, https://en.wikipedia.org/wiki/List_of_largest_container_ships, accessed September 29, 2017.

IMO IMDG.1: *International Maritime Dangerous Goods code*, vol. 1, International Maritime Organization 2006.

Biographies

Kiros Gebrearegawi Kebedow is a PhD student and lecturer at Hawassa University, Hawassa, Ethiopia. He has an MSc in applied mathematics from Norwegian University of Technology and Science.

Johan Oppen is a research scientist in logistics at Møreforsking Molde and Associate Professor in logistics at Molde University College, Specialized University in Logistics, in Molde, Norway. He has and MSc and a PhD in logistics from Molde University College, and has ten years of experience in teaching and research in quantitative logistics.