The Stowage of Containers for Inland Shipping: A System for Maximizing Containers Allocation and Meeting Stability Requirements

S. Fazi
Faculty of Economics and Business, University of Groningen, Groningen, The Netherlands (s.fazi@rug.nl)

Abstract – In many countries worldwide, inland shipping is a reliable and viable option to transport maritime containers inland. This modality comes with a set of operational challenges, among which the stowage of containers, i.e. the positioning of a set of containers onboard, is a complex and delicate task. In this paper, we develop a DSS to support the generation of feasible stowage plans, with the goal of maximizing the amount of containers onboard, while guaranteeing stability and complying with a set of rules. In this regard, we propose the case of inland shipping in the Netherlands. The stowage problem is modelled with a mixed integer linear programming mathematical formulation. Numerical experiments validate the model and give insights on the complexity of the problem for real-world size instances.

Keywords – Inland shipping, Stowage, Containers, MILP, DSS

I. INTRODUCTION

In Northern Europe, inland shipping has a leading role in the transport of containers from seaports to the hinterland. The reason is behind the presence of many canals and good infrastructures that allow a far reach and an easier generation of economies of scale. In this regard, several authorities are promoting this modality, with clear modal split goals for the next two decades [1].

Operationally, the transport by barge comes with a set of challenges, which require both expertise and decision support systems. Among these, the stowage of a set of containers on a barge is among the most delicate and complex challenges. The stowage is defined as the decision on the exact position of containers within a boat in order to guarantee stability during the navigation and comply with legislation and physical constraints. Compared to large vessels, potentially carrying thousands of containers, barge stowage is rather delicate, as the wrong position of even a single container may upheaval stability.

Due to limited ship sizes, from 20 TEU to ~400 TEU, stowage plans are currently performed by hand. Typically, sailors try several configurations by trial and error on an available stowage simulation controlling software, if available, which gives green light as soon as the tried configuration reaches an acceptable stability. The captain also follows basic guidelines from experience. Obviously, this procedure, besides being time consuming, may also bring to container rejections in case the stowage turns out to be too complex. These drawbacks are more evident when: the sailor has little experience in stowage plans, the sailor has never sailed on a specific boat, the amount of containers is large or close to maximum capacity.

Given this problem, our goal is to develop a decision support system (DSS) to support the stowage plan of barges. By means of a case study, of a Dutch barge transport provider, we provide a description of the system and main stability guidelines. The core of the DSS is based on a mixed integer linear programming mathematical formulation, considering stability, legislation and typical practical constraints. By means of numerical experiments we validate the DSS and give insights on complexity for real-world size instances.

In terms of stowage, to our knowledge, no available literature is specific on barge transport, but rather focuses on large container vessels. In general, scientific papers tackle specific aspects of the stowage problem and provide several assumptions, due to the complexity of the problem and size of real-world problems for large vessels. In this regard, we find two main lines of research. In the first branch, papers do not face stowage in terms of stability or weight, but focus on the minimization of shifts required (i.e. overstowage) by cranes to load and unload containers, given a certain rotation for the boat to follow. See for example, [2] and [3]. In the second branch, accurate stowage plans are developed, with formulations accounting for the exact position of containers and in some cases, stability guidelines. For example, [4] develop a mathematical model that includes stability guidelines and aim to minimize the time for loading containers. Finally, the model considers a given sequence of docks where only unloading operations are performed, for 20 and 40 foot containers. The model is solved heuristically and results are validated with a set of real world instances.

Our contribution is to present and analyze the stowage of containers in inland shipping; and to develop a novel model. Our work extends the work of [4], by including: an extended model with pickup and delivery routing, loading and unloading of containers; all types of containers; actual stability formulas in accordance to legislation and physics of the boat.

This paper is structured as follows. In Section II, we provide a formal problem description. In Section III, we develop the mathematical formulation. In Section IV, we
show preliminary results. Finally, in Section V, we draw our conclusions.

II. PROBLEM DESCRIPTION

We tackle the problem of transporting a set of containers from an inland terminal to the sea port, and a set from the sea port to the inland terminal by barge, in the same round trip. The details of this system follow.

A. Transport

We consider a set of export containers (i.e. delivery), available at the inland terminal, that need to be moved to the sea port to a specific dock. We also consider a set of import container (i.e. pickup), that need to be picked up at certain docks and moved to the inland terminal. The barge follows a specific route, including docks where pickup and delivery operations occur. In every route, capacity limits should be met. We refer to [5] for a description of this routing problem.

In this paper, we assume that a feasible route is given, together with a list of containers. We focus on the task of the captain who has to find a right stowage plan that guarantees stability for each leg. His/her task is to limit container rejections, thereby maximize the amount of transported containers.

B. Stability

As soon as any load is placed on board, the initial stability of the barge is jeopardized. The position of a container on a barge is usually considered according to its location in terms of rows, bays and tiers. In the barge, a bay, referred to with “x”, expresses the longitudinal position, the row the latitudinal position, referred to with “y”, and the tier represents the vertical position on the boat, referred to with “z”. For convenience, and given the fact that containers are standardized units, it is typical to see the boat divided in imaginary slots. Some slots could be dedicated either to 20 or 40 foot, or special containers, such as 45 footers, containers with dangerous cargo, or reefer containers, for the proximity of electric plugs.

There are three main factors to be calculated to assess stability: the center of gravity, the angle of list, and the trim. These items concern respectively forces that act on the barge, and they are typical measurements for every ship. The first two items concern the transverse (or latitudinal) stability, whereas the latter the longitudinal stability.

Looking at a boat from the stern in a perfect upright position, by tracing an imaginary line up from the keel (K), perpendicular to the surface of the earth, we encounter the center of buoyancy, the center of gravity, and the metacenter. The center of buoyancy (B) is the point where the force of buoyancy pushes the barge upwards to keep it afloat. The metacenter (M) is a property of the ship and is an abstract reference point where the ship starts oscillating as soon as there is a displacement. The center of gravity (G) is an imaginary point where the whole mass of the ship is concentrated for the gravitational force to act and pull the boat down.

The first requirement for a stable barge is that G is below M. As soon as weight is placed on the boat, G and M tend to shift respectively upwards and downwards. To prevent G to raise too much, the heavier cargo should be placed at the bottom of the boat, to keep a low center of gravity. In the Netherlands, by law, G should additionally be below an imaginary point G\text{max}, which is a precautionary measure that considers also external factors, such as wind. In order to compute G and thereby KG, i.e. the distance between K and G, we can use the following formula:

\[ KG = KG_0 + \sum_i w_i (z_i - z_0) \left/ \left( W_b + W_c \right) \right. \]

where \( KG_0 \) is the KG of a barge, without containers and including half of the maximum water, oil and fuel supply. \( z_i - z_0 \) is the height of the center of gravity of a placed container, computed from the floor of the boat; if the container is placed for example in the second tier \( z_i \) includes the height of the container placed below. \( W_b \) is the weight of the empty boat, including half of the maximum water, oil and fuel supply. Whereas \( W_c \) is the weight of the loaded containers, thereby is a variable. From the formula, it is clear that the KG will tend to be larger in case the heavier containers are placed in intermediate or high tiers. For a given load \( W_c \), \( KG < KG\text{max} \), where \( KG\text{max} \) is a value available in the documentation of the boat, and depends on weights, amount of layers, and type of containers. For example, a boat loaded with 3 layers of 40 foot standard containers has a higher \( KG\text{max} \) of another with 3 layers of 40 foot high-cube containers with same weight, since high-cube containers have a higher center of gravity.

A second requirement for stability concerns the angle of list, which is the angle to which the boat heels to either port or starboard. The angle of list should be minimum, which entails that there should not be excess of loads in one of the sides of the boat. The formula related to the angle of list is as follows:

\[
\tan(\sigma) = \frac{\sum_i w_i (y_i - y_0)}{\sqrt{w_b + w_c} GM} \]

where \( y_i \) is the y-coordinate of the placed \( i^{th} \) container, \( y_0 \) is the central position, \( GM \) is the distance between G and M, \( GM = KG - KG' \). \( GM \) is a value that depends on the loaded weight and is drawn during construction phase of the boat. Hence, it must be reported
Finally, the trim concerns longitudinal stability and is the difference between the forward (bow side) and aft (stern side) drafts, where draft is the vertical distance between the keel and the waterline. When the boat is inclined towards the aft, it is said that has a trim of 

\[ \delta \] 
defined on a network of a container on a barge, in the transport legs of a given container on top of 40 foot and the weights of containers to avoid overstowage, which is the re-handle of containers or has its origin (if pickup) at a dock 

\[ V \]. With concern to bays, each slot may be occupied by the containers, and 

\[ X, Y \] and the remaining to the docks (subset \( D \)). We formulate the problem with the following compact MILP formulation:

\[ \text{Max } \sum_{xyz} \sum_i \delta_{xyz,i} + \sum_{xyz} \sum_k \delta_{xyz,k0} \quad (1) \]

\[ \text{Subject to:} \]

\[ \sum_{i} \delta_{xy,ik} B_{ij} \leq 0 \quad \forall \ i \in E, j \in D \quad (2) \]

\[ \sum_{i} \delta_{xy,ik} B_{ij} \leq 0 \quad \forall \ i \in E, j \in D \quad (3) \]

\[ \sum_{i} \delta_{xy,ik} (1 - B_{ij}) = \sum_{i} \delta_{xy,jk} (1 - B_{ij}) \quad \forall \ i \in C, j \in D, xyz \quad (4) \]

\[ \delta_{xy,ik} B_{ij} + \delta_{xy,jk} (1 - B_{ij}) \leq 1 \quad \forall \ i \in E, s \in C, j \in D, xyz \quad (5) \]

\[ \sum_{i} \sum_{xy} \delta_{yx,jk} \leq f_{jk} M \quad \forall (j, k) \in A \quad (6) \]

\[ \sum_{i} \delta_{xy,jk} \leq 1 \quad \forall (j, k) \in A, i \in C \quad (7) \]

\[ \sum_{i} \delta_{xy,jk} \leq 1 \quad \forall (j, k) \in A, xyz \quad (8) \]

\[ \sum_{i} \delta_{yx,jk} \leq 0 \quad \forall (j, k) \in A, x \in V \quad (9) \]

\[ \sum_{i} \delta_{yx,jk} \leq 0 \quad \forall (j, k) \in A, x \in O \quad (10) \]

\[ \sum_{i} \delta_{xy,jk} + 0.5 \sum_{i} \delta_{x+1yz,jk} + 0.5 \sum_{i} \delta_{x-1yz,jk} \leq 1 \quad \forall (j, k) \in A, x \in V, yz \quad (11) \]

\[ \sum_{i} \delta_{xy,jk} + \sum_{i} \delta_{x+1yz,jk} + \sum_{i} \delta_{x-1yz,jk} - \sum_{i} \delta_{yx,jk} \geq 0 \quad \forall (j, k) \in A, x \in V, y, z > 1 \quad (12) \]

\[ h_{jk} \geq \delta_{xy,jk} H_{i} - (1 - \delta_{xy,jk}) M + h_{jk} - (1 - \delta_{xyz-1,jk}) M \quad \forall (j, k) \in A, i, s \in C, xyz \quad (13) \]

\[ h_{jk} \leq \delta_{xy,jk} H_{i} + (1 - \delta_{xy,jk}) M + h_{jk} + (1 - \delta_{xyz-1,jk}) M \quad \forall (j, k) \in A, i, s \in C, xyz \quad (14) \]

\[ K_{G_{\text{max}}} > KG (\delta) \quad (15) \]

\[ \sigma (\delta) \leq \tan (11) \quad (16) \]

\[ t(\delta) > 0 \quad (17) \]

\[ t(\delta) \leq 0.15 \quad (18) \]

The objective function (1), maximizes the amount of containers stowed on board for all transport legs. With inequalities (2), we impose that export containers are not on board from their discharging node. With (3), import
containers should not appear before their pickup node. With constraints (4) we impose that the container, if on board, remains in the same position before and after an intermediate dock. With (5) we impose that no overstowage is possible. With inequalities (6), we impose that the stowage is empty in leg \((j, k)\), if this is not in the rotation. Inequalities (7) and (8) are typical assignment constraints, imposing that a container can only be assigned to one slot and that a slot can only be assigned to one container. Constraints (9) and (10) make sure that a 20 foot containers is placed in an odd bay, whereas a 40 foot in an even one; likewise, additional constraints could be added for other special containers, such as reefers, open top, and 45 foot containers. With (11), a physical slot may be occupied by either 1 or 2 20 foot containers, or by 1 40 foot container. Inequalities (12) make sure that a 40 foot containers is supported in the tier below by a heavier 40 foot container or by 2 20 footers, whose total weight is larger; these also implicitly state that a container must not float. Likewise, (13) is the condition for 20 foot containers. Constraints (14) and (15) compute the height of each container from the keel. This is needed for the formula of the KG, reported in section II-B. Inequalities (16), (17), (18) and (19) are the stability constraints. For brevity, we report them as a function of variable \(\delta\). Given the non-linearity of the original stability equations, made of the product between binary and linear variables, these formulas are linearized with a system of inequalities.

The formulation is an extension of the master bay problem introduced by [4]. The model adds specifically the hard constraint for no overstowage; pickup and delivery features; exact computations of stability indexes. In the numerical section, we provide a set of experiments to test the computational limits of a solver.

IV. NUMERICAL ANALYSIS ON REAL WORLD DATA

In this section, we provide numerical tests on a set of 18 real-world instances, provided by a barge transport provider. We focus on a specific barge, whose documentation is available. The boat has a maximum capacity of 104 TEU.

The instances are drawn from an available data set, where import and export containers are tagged with a voyage trip. We draw the instances by taking containers from a specific voyage and if the amount does not fulfill the capacity, additional containers are included in the instance.

The MILP formulation is implemented in CPLEX 12.6, using the concert technology and solved by a branch and cut algorithm. All experiments have been run on an Intel(R)Core(TM)2 DUO machine with 2.93 GhZ and 4.00 GB RAM.

A. A simple instance

To give a better understanding of our problem, we propose here a simple example, consisting of 20 containers, 10 import and 10 export, and a route consisting of 4 docks, numbered from 1 to 4, and the inland terminal with index 0. The route starts from 0, follows the index order and finishes at 0. In Table 1, the details of the containers.

<table>
<thead>
<tr>
<th>Destination / Origin</th>
<th>Containers Export</th>
<th>Weight (tons)</th>
<th>20TEU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0,1,4,5,8,12,19</td>
<td>12,19</td>
<td>4,16,6,16,23,8</td>
</tr>
<tr>
<td>2</td>
<td>2,3,10,11,13,15</td>
<td>10,11,13,15</td>
<td>6,16,23,23,9,3</td>
</tr>
<tr>
<td>3</td>
<td>6,7,14,16</td>
<td>14,16</td>
<td>13,16,29,23</td>
</tr>
<tr>
<td>4</td>
<td>9,17,18</td>
<td>17,18</td>
<td>13,9,8</td>
</tr>
</tbody>
</table>

The computation, successful in this case, will give as output the stowage for each transport leg. In Figure 1, we report the given solution for legs (2,3) and (3,4). At dock 3, export containers 14 and 16 are dropped, whereas 6 and 7 are picked up. The barge is still carrying export containers 17 and 18, that will be discharged at dock 4. Moreover, import containers 0,1,2,3,4,5,8 were picked up at docks 1 and 2, thereby they will stay on board with 6 and 7 till the end.

The MILP formulation is implemented in CPLEX 12.6, using the concert technology and solved by a branch and cut algorithm. All experiments have been run on an Intel(R)Core(TM)2 DUO machine with 2.93 GhZ and 4.00 GB RAM.

B. Experiments

We now present experiments on 18 instances. Due to complexity only a few have an amount of containers for import/export reaching maximum capacity of the boat. We try experiments with and without stability constraints. The results are summarized in Table II and Table III.

Fig. 1. Example of a stowage plan
TABLE II
Results with full formulation. OoM stands for “Out of Memory”, “BI” is the best integer found (Lower Bound of the maximization problem). “BN” is the best non integer solution (Upper Bound). Time is expressed in seconds.

<table>
<thead>
<tr>
<th>Instance #</th>
<th>C Docks</th>
<th>BI</th>
<th>BN</th>
<th>GAP</th>
<th>Time</th>
<th>OoM</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3 5 0</td>
<td>30</td>
<td>30</td>
<td>0</td>
<td>675</td>
<td>yes</td>
</tr>
<tr>
<td>2</td>
<td>3 4 31</td>
<td>31</td>
<td>31</td>
<td>0</td>
<td>875</td>
<td>no</td>
</tr>
<tr>
<td>3</td>
<td>32 32</td>
<td>32</td>
<td>32</td>
<td>0</td>
<td>232</td>
<td>no</td>
</tr>
<tr>
<td>4</td>
<td>33 33 0</td>
<td>33</td>
<td>33</td>
<td>0</td>
<td>104</td>
<td>no</td>
</tr>
<tr>
<td>5</td>
<td>35 35 0</td>
<td>35</td>
<td>35</td>
<td>0</td>
<td>81</td>
<td>no</td>
</tr>
<tr>
<td>6</td>
<td>36 36 0</td>
<td>36</td>
<td>36</td>
<td>0</td>
<td>31</td>
<td>no</td>
</tr>
<tr>
<td>7</td>
<td>36 36 0</td>
<td>36</td>
<td>36</td>
<td>0</td>
<td>326</td>
<td>yes</td>
</tr>
<tr>
<td>8</td>
<td>37 37 0</td>
<td>37</td>
<td>37</td>
<td>0</td>
<td>1525</td>
<td>yes</td>
</tr>
<tr>
<td>9</td>
<td>3 4 31</td>
<td>31</td>
<td>31</td>
<td>0</td>
<td>875</td>
<td>no</td>
</tr>
<tr>
<td>10</td>
<td>38 38 0</td>
<td>38</td>
<td>38</td>
<td>0</td>
<td>175</td>
<td>no</td>
</tr>
<tr>
<td>11</td>
<td>40 40 0</td>
<td>40</td>
<td>40</td>
<td>0</td>
<td>224</td>
<td>yes</td>
</tr>
<tr>
<td>12</td>
<td>45 45 0</td>
<td>45</td>
<td>45</td>
<td>0</td>
<td>2162</td>
<td>yes</td>
</tr>
<tr>
<td>13</td>
<td>55 55 0</td>
<td>55</td>
<td>55</td>
<td>0</td>
<td>1525</td>
<td>yes</td>
</tr>
<tr>
<td>14</td>
<td>56 56 0</td>
<td>56</td>
<td>56</td>
<td>0</td>
<td>175</td>
<td>no</td>
</tr>
<tr>
<td>15</td>
<td>57 57 0</td>
<td>57</td>
<td>57</td>
<td>0</td>
<td>224</td>
<td>yes</td>
</tr>
<tr>
<td>16</td>
<td>65 65 0</td>
<td>65</td>
<td>65</td>
<td>0</td>
<td>2162</td>
<td>yes</td>
</tr>
<tr>
<td>17</td>
<td>86 86 0</td>
<td>86</td>
<td>86</td>
<td>0</td>
<td>1525</td>
<td>yes</td>
</tr>
<tr>
<td>18</td>
<td>93 93 0</td>
<td>93</td>
<td>93</td>
<td>0</td>
<td>224</td>
<td>yes</td>
</tr>
</tbody>
</table>

FULL FORMULATION

The experiments show that in case of full formulation CPLEX struggles to find optimal solutions. Instance #12 is the only instance, among those generating Out of Memory status, for which CPLEX could get close to the UB. As the size of the instances increases, the complexity obviously increases; however, also the amount of docks to be visited play a role. See for example Instance #1, with “only” 30 containers, but 5 docks to be visited. For larger instances, CPLEX cannot even start the computation due to the large size of the problem.

In case of partial formulation, CPLEX can handle the problem more easily. Only Instance #17 finished without a solution. The reason behind this is due to the massive amount of additional constraints and variables, needed to keep track of the relative height of each placed container from the keel, and the linearization of the stability formulas.

V. DISCUSSION

The fact that the relaxed formulation has such a large impact on the performances of CPLEX suggests several alternatives for future research. First, limiting the definition of additional variables and constraints, by using simpler constructions. For the trim, for example, an idea could be to impose that the total weight placed fore is slightly larger than the weight aft. For the KG, it might be enough to guarantee that containers are supported by heavier ones, since in this case the center of gravity is kept low.

Alternatively, heuristics and metaheuristics could be developed. For heuristics, an idea could be to divide the boat into slots dedicated to specific docks, and solve a sub problem for each slot. In order to reach stability, these slots may be interchanged.

REFERENCES