The storage space allocation problem in a dry bulk terminal: a heuristic solution

Assia Ait Ouhaman*,**. Khalid Benjelloun*,**. Jean Pierre Kenné***. Najib Najid****

*Mohammadia School of Engineers, Mohammed V University, Rabat, Morocco (e-mail: aitouahman.assia@gmail.com).
**Innovation Lab for Operation, Mohammed VI Polytechnic University, Benguerir, Morocco, (e-mail: assia.aitouahman, khalid.benjelloun}@um6p.ma)
*** Mechanical Engineering Department, École de Technologie Supérieure, University of Quebec, Montréal, Canada, (e-mail: Jean-Pierre.Kenne@etsmtl.ca))
**** Mechanical and Productive Engineering Department, University of Nantes, Nantes, France, (e-mail: najib.najid@univ-nantes.fr)

Abstract: The bulk port operations are given a growing attention for their important role in the global supply chain in different industries (mining, energy ...). To guarantee their competitiveness, the efficient management of port logistics, including yard side management is crucial. In this paper, we consider a real-world storage space allocation problem at an export bulk terminal. We formulate the problem as a mixed integer linear program and we propose a heuristic method to solve large scale data sets. Both the model and the heuristic can help the yard planner to test different scenarios and provide better stock yard plan, which is a first step toward improving the management of port operation in the bulk terminal under study.

Keywords: Stockyard management, storage space allocation, bulk port, port operations, optimization.

1. INTRODUCTION

Bulk ports are a critical part of the supply chain in the mining industry as they are the link between the production and the delivery of the product to the end customer. A good management of logistic operations at the port level is needed in order to allow a continuous flow from production to the loading terminal. The logistic operations must be carried out as efficiently as possible to avoid ship delays and demurrage penalties, as well as to prevent congestion, which hinder capacity development and supply diversification. The reader is referred to Leite et al. (2019) for more insights into the different operations carried out in mining companies’ supply chain including the port management and the optimization methods applied in this field.

Port operations can be classified into three categories: seaside operations, yard operations and landside operations. The berth and the quay areas are considered seaside, while the storage yard is considered as part of the yard side. The transport area is at the intersection of the seaside and landside areas. The logistic operation in terminal is a complicated system that can be divided into several interrelated subsystems. In Ding et al. (2012) four subsystems are proposed: berth subsystem, loading and unloading subsystem, storage subsystem and horizontal transportation subsystem. We can also provide a hierarchical decomposition of the management system of bulk terminal according to the three categories classification (Fig. 1) that represent different decisions to be made in day-to-day port operations.

In this work, we focus on the storage yard management in bulk port, which has been studied previously by some researcher. Although the interest in container terminals received more attention (Carla et al., 2014), a growing attention is turning toward the bulk ports in recent year. In the following we present some previous work that tackled the problem of stockyard management in bulk ports environment.

Fig. 1. A hierarchical decomposition for port operations

From the earliest work, we can find the work of Abdekhoadae et al. (2004) where they presented an integrated approach for the stockyard management with an emphasis of the railway network scheduling in a coal export terminal. Ago et al. (2007) studied a steel-making plant proposing a model for the integrated storage allocation and routing problem. They provided a formulation of the problem using a mixed integer programing approach and proposed a Lagrangian decomposition to solve it and compared it with the hierarchical planning method.

Many authors focused on approximate solution approaches. Boland et al. (2011) studied the combined problem of stockpile allocation with ship scheduling and developed a model with a greedy solution. They improved it using enumeration and integer programming in a later work (Boland...
et al. 2012). Hanoun et al. (2013) proposed a heuristic for planning stockpiles and scheduling resources to minimize delays in production taking into consideration the coal age in the stockyard. Babu et al., (2015) proposed two heuristic-based greedy construct algorithms to improve port terminal operation in a coal terminal. The objective was to minimize the delay of ships while maximizing the throughput capacity. Pratap et al., (2016) provided a block-based evolutionary algorithm to optimize operation of rake loading and stockyard management considering the order of berthing of vessels. Another approach was proposed by Burdett et al., (2019) where the yard management in a coal export terminal was compared to flexible job shop scheduling problem and solved with a meta-heuristic algorithm.

More recent work attempt to solve this problem with exact solution methods. For example Menezes et al., (2017) proposed a branch and price algorithm to solve the integrated problem of planning and scheduling phase in an iron export terminal. Tang et al., (2016) presented a formulation of storage yard management with ship scheduling in an iron ore import terminal. They used the Bender decomposition approach to provide an iterative procedure to solve the proposed MILP.

We can note from the state of art above that each problem studied differ from the others as each terminal has its own rules of management which require specific constraints. In this paper we address a dry bulk material handling facility for an export terminal where we model the storage yard allocation problem while capturing its specific rules. The problem of storage space allocation consists of two main decision: where to store incoming materials from production plant and when to start this operation.

The remainder of this paper is as follow: section 2 describes the yard side management problem in the bulk terminal under study. Section 3 gives a Mixed Integer Linear Program (MILP) formulation of the problem. Section 4 presents the computational results of the MILP. We present the heuristic approach in section 5 and we finally conclude in section 6.

2. PROBLEM DEFINITION

The stockyard management consists of receiving multiple types of materials from different production sites, storing them in the storage yard and delivering these materials to the arriving ships in the terminal through conveyors (Fig.2). The storage of material is done in six identical hangars that have the same capacity and that could contain different types of products. Each hangar has one stacker (the engine used to store product coming from plant) and one reclaimer (the engine used to reclaim the materials from the storage area toward the ship loaders during the loading phase of the handling process).

For each planning period, we have a set of products coming from the production sites which have to be stored in a precise subarea of a hangar. The production plan is based on the commercial plan that expresses the needs from each product in order to satisfy client demands. It is usually provided a week earlier and it contains, for each operation, the product grade, the expected production time and the quantity. The yard planner is in charge of finding the best storage configuration according to the yard layout while taking into consideration the state of each hangar. It is a critical phase, as the planner needs to provide a plan that guarantees the continuity of the material flow preventing any production stoppage.

![Fig. 2. Bulk product flow](image1)

The stacking (storing) operation consist of storing the material using a stacker equipped with a moving tripper car that can travel along the top of the storing hangar and can reach all its subareas. Fig.3 gives more details of the composition a storage hangar.

![Fig. 3. The composition of two storage hangars](image2)
are retrieved first. For these reasons we limit our problem to the material flow from plant to storage hangar and we characterize each subarea by an availability time that reflects the end of the retrieval operation. The scheduling of retrieval operations is not considered in this formulation.

The formulation of the problem is based on the discretization of each hangar into equal subareas. An illustrative example is shown in Fig. 4. Three types of subareas are considered:

- Subareas that are occupied and will remain untouched during the planning period (marked in red) and therefore could not be assigned to any product.
- Subareas that are being retrieved during the planning period and are expected to be free to receive new product starting from a certain time.
- All remaining subareas are considered available at the beginning of the planning period and have an availability time equal to zero.

![Fig. 4. Illustration of the discretization approach](image)

In the next section, we will present a MILP formulation based on the following assumptions:

- The material stored are not retrieved completely during the planning horizon.
- Each hangar can store only one product at a time.
- The problem is always feasible as we consider that there will be enough space for storing all upcoming material during the planning period.

### 3. PROBLEM FORMULATION

Our formulation of the stockyard allocation problem is inspired from Tang et al., (2016). They considered that each subarea can receive a maximum amount of a given product, which is a fixed value considered as an input independently from the total quantity of the product to be stored. But in fact, the maximum quantity that a subarea can contain depends on the total quantity of product to be stored. Therefore, we chose to add a preprocessing module that defines the required number of subareas for each product to be stored, following equation (1) that gives the relation between stockpile length \( l \) and the total quantity of product \( Q \) to be stored \( Q_{tp} \). This equation considers the characteristics presented in table 1 and the stockpile shape in Fig. 5. The value for the height of the stockpile corresponds to the maximum height that can the reclaim reach.

\[
l = \frac{\rho Q t_p}{2 w h} + \frac{2 h}{\tan(\alpha)} - \frac{2 R}{h^2} = Y \sum_i^p \star S_i = X_{endphs} \frac{Y_{phs} + X_{endphs}}{Z_{ph} + X_{endphs}} (1)
\]

From the total length required to store a product we calculate the required number of subareas for each product dividing it by the discretization step \( d_c \) chosen.

![Fig. 5. A stockpile shape](image)

### 3.1 Notation:

**Parameters sets:**

- \( P \) Set of products to be stored
- \( H \) Set of hangars.
- \( S \) Set of Subareas.

**Parameters:**

- \( C_{phs} \) Penalty cost of storing product \( p \) in hangar \( h \) at subarea \( s \).
- \( C_{tp} \) Penalty cost of tardiness product \( p \).
- \( T_{ap} \) Arrival time of product \( p \).
- \( T_{sp} \) Time for product \( p \) to be stored.
- \( T_{spq} \) Setup time between product \( p \) and product \( q \).
- \( Num_{subp} \) The required number of subareas of product \( p \) to be stored.
- \( T_{avail} \) Availability time of subarea \( s \) in hangar \( h \).
- \( S_{ocphs} \) \{0, if subarea \( s \) in hangar \( h \) is occupied. 1, otherwise.

**Decision variables:**

- \( T_{startp} \) Starting time of the storing of product \( p \).
- \( T_{tardp} \) Tardiness time of the storing of product \( p \).
- \( Seq_{pq} \) \{1, if product \( p \) is processed before product \( q \). 0, otherwise.
- \( Y_{phs} \) \{1, if product \( p \) is assigned to the hangar \( h \) in subarea \( s \). 0, otherwise.
- \( X_{endphs} \) \{1, if subarea \( s \) is the last subarea assigned to product \( p \) in hangar \( h \). 0, otherwise.
- \( Z_{ph} \) \{1, if product \( p \) is assigned to hangar \( h \). 0, otherwise.

### 3.3 Objective function:

\[
\text{Minimize} \sum_{p \in P} C_{pph} \sum_{h \in H} \sum_{s \in S} C_{phs} \cdot Y_{phs} + \sum_{p \in P} C_{tp} \cdot T_{tardp} (2)
\]

The first term of the objective function reflects the penalty cost for storing a given product \( p \) to a number of adjacent subareas. The cleaning operations are usually not carried out as

<table>
<thead>
<tr>
<th>Product density ( \rho )</th>
<th>1.0412 ( \text{t/m}^3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repose angle ( \alpha )</td>
<td>30</td>
</tr>
<tr>
<td>Stockpile height ( h )</td>
<td>8 m</td>
</tr>
<tr>
<td>Hangar width ( w )</td>
<td>15 m</td>
</tr>
</tbody>
</table>

Baseline values for the parameters in table 1.
efficiently as required, therefore some products type have higher contamination rate if they are stored in the subareas where certain types of other products were stored previously. For example, if a product $p_1$ is highly contaminated with a product $p_2$, the penalty cost for storing product $p_1$ to subareas previously occupied by product $p_2$ will be very high.

The second term concern the penalty related to the tardiness time of supply satisfaction that should be as small as possible in order to guarantee the continuity of the production flow.

3.4 Constraints:

The tardiness time for each product is the difference between the arrival time of the product and the starting time of the stacking operation for this product:

$$ T_{\text{tard}_p} \geq T_{\text{start}_p} - T_{\text{ap}}, \quad \forall p \in P $$

(3)

For a given hangar, only one product can be stored at a time, as each hangar has only one stacker. The following constraints (4) to (7) ensure the product sequentially:

$$ T_{\text{start}_p} - T_{\text{start}_q} + M * (1 - \text{Seq}_{pq}) \geq T_s + T_{\text{pq}}, \quad \forall p, q \in P, p \neq q $$

(4)

$$ \text{Seq}_{pq} + \text{Seq}_{qp} \leq 1, \quad \forall p, q \in P, p \neq q $$

(5)

$$ \text{Seq}_{pq} + \text{Seq}_{qp} \leq Z_{ph} + Z_{qh} - 1, \forall p, q \in P, p \neq q $$

(6)

$$ \text{Seq}_{pq} + \text{Seq}_{qp} \leq 1 + Z_{ph} - Z_{qh}, \quad \forall p, q \in P, p \neq q $$

(7)

The storing operation of a product cannot begin in a given subarea before its availability time and the product’s arrival time:

$$ T_{\text{start}_p} \geq T_{\text{av}_p} * Y_{\text{phs}}, \quad \forall p \in P, \forall h \in H, \forall s \in S $$

(8)

$$ T_{\text{start}_p} \geq T_{\text{ap}}, \quad \forall p \in P $$

(9)

The number of subareas affected to the product to be stored should be equal to the required number of subareas:

$$ \sum_{s \in S} Y_{\text{phs}} \geq \text{Num}_{\text{sub}_p} * Z_{ph}, \quad \forall p \in P, \forall h \in H $$

(10)

The occupied subareas cannot be affected to any product:

$$ \sum_{p \in P} Y_{\text{phs}} \leq \text{Soc}_{hs} \cdot \forall h \in H, \forall s \in S $$

(11)

Each product should be placed in only one hangar:

$$ \sum_{h \in H} Z_{ph} = 1, \quad \forall p \in P $$

(12)

For different adjacent products, a safety distance should be respected. We add a fictive end subarea for each hangar (to represent the end of the hangar) which will always be equal to 0:

$$ Y_{\text{phs}_{\text{end}}} = 0, \quad \forall p \in P, \forall h \in H $$

(13)

$$ \sum_{p \in P(s)} Y_{\text{phs}(s+1)} \leq 1 - Y_{\text{phs}}, \quad \forall p \in P, \forall h \in H, \forall s \in S $$

(14)

The allocation should not exceed the last subarea that is assigned to product $p$ and each product is allocated to only one hangar:

$$ Y_{\text{phs}} - Y_{\text{phs}(s+1)} \leq X_{\text{phs}}, \forall p \in P, \forall h \in H, \forall s \in S $$

(15)

$$ \sum_{s \in S} X_{\text{phs}} \leq Z_{ph}, \forall p \in P, \forall h \in H $$

(16)

The non-negativity and binary constraints are expressed as follow:

$$ T_{\text{tard}_p}, T_{\text{start}_p} \geq 0, \quad \forall p \in P $$

(17)

$$ Y_{\text{phs}} * Z_{ph}, \text{Seq}_{pq} \in \{0,1\} \quad \forall p \in P, \forall h \in H, \forall s \in S $$

(18)

4. COMPUTATIONAL RESULTS

We run the program in IBM ILOG CPLEX Optimization 12.7.1.0 on an Intel(R) Core (TM) i7-8550U CPU 1.80GHz, 1992 MHz, quadcore processor with 8GB RAM.

We tested different scenarios, depending on the initial state of the storage hangars and we varied the discretization step $d_s$ as well as the number of arriving products. The experimental results are listed in Table 2.

For each data set, the model provides a different allocation. Fig. 6 is an illustration of the allocation resulted from running data set 1, 2 and 3. It shows that the initial state of each hangar affects the way products are assigned as well as the cost associated with the operation of storage. This could help the yard planner to test different configuration and to measure its impact on the yard layout.

Fig. 6. Example of the allocation result

As expected, the computation time of the proposed MILP failed to converge within accepted computational time for large instances (Data set 11 took about 20 hours). Therefore, we propose a heuristic approach, which is presented in the next section.

Table 2. Experimental Results:

<table>
<thead>
<tr>
<th>$d_s$</th>
<th>Data set</th>
<th>Stock occupancy</th>
<th>Product Nbr</th>
<th>Time (s)</th>
<th>Tardiness Cost</th>
<th>Storage cost</th>
<th>Objective function</th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>1</td>
<td>0%</td>
<td>5</td>
<td>2</td>
<td>0</td>
<td>134</td>
<td>134</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>53%</td>
<td>5</td>
<td>2</td>
<td>0</td>
<td>1238</td>
<td>1236</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>52%</td>
<td>5</td>
<td>3</td>
<td>10</td>
<td>1294</td>
<td>1304</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>52%</td>
<td>5</td>
<td>3</td>
<td>10</td>
<td>1294</td>
<td>1304</td>
</tr>
<tr>
<td>8</td>
<td>7</td>
<td>39%</td>
<td>7</td>
<td>15</td>
<td>1470</td>
<td>3215</td>
<td>4685</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
<td>39%</td>
<td>10</td>
<td>22</td>
<td>1480</td>
<td>3215</td>
<td>4695</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>53%</td>
<td>5</td>
<td>18</td>
<td>990</td>
<td>6424</td>
<td>7414</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>40%</td>
<td>7</td>
<td>204</td>
<td>1500</td>
<td>1236</td>
<td>2736</td>
</tr>
<tr>
<td>11</td>
<td>10</td>
<td>20%</td>
<td>10</td>
<td>207</td>
<td>2010</td>
<td>2848</td>
<td>4494</td>
</tr>
<tr>
<td>11</td>
<td>1452</td>
<td>20%</td>
<td>15</td>
<td>71452</td>
<td>63830</td>
<td>7523</td>
<td>71353</td>
</tr>
</tbody>
</table>

5. HEURISTIC APPROACH

The storage space allocation problem can be compared to the bin packing problem according to Boland et al. (2011). Following this approach, we define a set of blocs that constitute the available subareas. Each bloc is characterized by: the number of subareas that it contains, the hangar in which
it’s located and the first and the last subarea as represented in Fig. 7.

![Available bloc](image)

Fig. 7. Example of an available bloc

### 5.1 Additional notation:

- $B$: Set of all available blocs.
- $\text{Num}_{\text{sub}}_b$: Number of subareas in the available bloc $b$.
- $A_b$: Set of feasible allocations for product $p$ in feasible bloc $b$.
- $\text{marg}_{bp}$: The available space left if product $p$ is allocated to bloc $b$.
- $\text{Start}_b$: The first subarea of the available bloc $b$.
- $\text{End}_b$: The last subarea of the available bloc $b$.

### 5.2 Allocation procedure:

For each product, ordered by its arrival time, we form a set of feasible blocs. A feasible bloc is the one that has sufficient numbers of subareas to store the product. From this set, we choose the bloc that has the least left space (so that product that need more space could be stored later) and then we construct the set of feasible allocations.

The steps of the method are presented in the pseudo code of the algorithm below:

1. Order $p \in P$ according to arrival time
2. Select first $p$
3. Construct set of feasible blocs $B^p_f$:
   - For each $b \in B$
     - If $\text{Num}_{\text{sub}}_b \geq \text{Num}_{\text{sub}}_p$
       - $B^p_f \leftarrow b$
       - $\text{marg}_{bp} = \text{Num}_{\text{sub}}_b - \text{Num}_{\text{sub}}_p$
     - Endif
4. Choose blocs with lowest $\text{marg}_b$
5. Update $B^p_f$
6. Construct set of feasible allocations $A_b$ for each $b \in B^p_f$:
   - $A_b = \{(\text{Start}_b + \text{marg}_b - l, \text{End}_b - l) \mid l \in \{0, 1, ..., \text{marg}_b\}\}$
7. Calculate the Objective Function for each feasible allocation.
8. Choose the allocation with the lowest Objective Function.
9. Update the set of available blocs $B$.
10. Repeat for all products.

This algorithm was tested for the large instance (Data set 11) and could find a feasible solution in 1 minute. Although it showed an important gap from the optimal solution, it could be considered as a first step toward improving the planning process in the stock yard.

### 6. DISCUSSION AND CONCLUSIONS

Both the MILP model and the heuristic algorithm developed can be considered as a prototyping for a decision support system to manage the yardside operations in the bulk terminal under study. The proposed heuristic helped solving large scale data in a less computational time than the model. Further tests will be carried out to improve the proposed algorithm. The future work will focus on implementing meta heuristics approach for better quality solution.

### REFERENCES


Menezes, G.C., Mateus, G.R., Ravetti, M.G., 2017. A branch and price algorithm to solve the integrated production
