AN EXACT METHOD FOR BERTH ALLOCATION AT RAW MATERIAL DOCKS

Shaohua Li[1, a], Lixin Tang[b], Jiyin Liu[c]

[a]*Key Laboratory of Process Industry Automation, Ministry of Education, China*
[b]*Department of Systems Engineering, Northeastern University, Shenyang, 110004, China*
[c]*Business School, Loughborough University, Leicestershire, LE11 3TU, UK*

Abstract: This paper studies a short-term berth allocation problem encountered in the Baoshan Iron and Steel complex. A mathematical model is developed for the problem to minimize the total tardiness particularly considering special industrial characteristics. A lower bound derived by performing a Lagrangian relaxation, along with appropriate branching rules, is incorporated into a branch and bound algorithm for the berth allocation problem. Real data collected from the Baoshan Iron and Steel Complex are used to test the performance of the algorithm. Computation result indicates that the optimal berth scheduling can be obtained for the industrial-sized problem within an acceptable running time.

Keywords: Optimization problem; Mathematical models; Linear programming; Bounding method; Algorithms

1. INTRODUCTION

With the development of modern iron and steel industry, the operations at raw material docks are enormously increasing so that they become very congested. Although constructing more berths can avoid or decrease such congestion, it is a cheaper and more feasible method to improve the productivity of the existing berths. So to make an effective berth allocation has become a key factor to save financial and improve efficiency. Li *et al.* (1998) considered the berth allocation problem with a "multiple-job-on-one-processor" pattern and applied a generalized First-Fit Decreasing heuristic to several variations of the problem. Chen and Lee (1999) presented a one-job-on-multiple-machine model which can be used to handle a complicated berth allocation. Imai *et al.* (2003) modified the existing formulation of the berth allocation problem in order to treat calling ships at various service priorities. Compared with the literature in container terminals, studies often give more attentions to strategic and planning problems in the iron and steel industry. Only very scarce studies focus on the berth allocation problem in such an industry. Suzuki *et al.* (1996) described the coastwise transportation planning and administration system of Kawasaki steel in Japan. An expert system technology was applied to develop the subsystem providing transportation schedules by ships. Kao *et al.* (1990) expressed the constraints of the port and the working rules adopted by the ports of China Steel Corporation as knowledge rules and embedded them into the framework of the logic of dock arrangement. Kao and Lee (1996) regarded the medium-term ship scheduling problem as parallel-machine scheduling problem. This is the sole one which has formulated a pure zero-one integer program for the berth allocation problem in the iron and steel industry. However such a model does not match the characteristics of the berth allocation problem in the Baosteel. The material type is a deterministic factor to a berth allocation in the Iron and Steel industry for the great difference in ship types and other facilities, for example, conveyors and storage yards. If Berths are regarded as machines in a general scheduling problem, the difference in berths and unloaders installed on them make it impossible to treat a berth allocation problem in the iron and steel industry as an

[1]Corresponding author. E-mail: qhjytlx@mail.neu.edu.cn

identical parallel machine problem. Compared with container terminals, docks in the iron and steel industry have a smaller size and diversified ships. In order to improve the productivity of berths, more than one berth may process a huge ship and one berth can be shared by several small ships at the same time. These two speical cases may occur in container terminals and a very few researchers have made their studies on one of them (Li, *et al.*, 1998, Chen and Lee, 1999), whereas no literature include both of them. From the above discussion we can see that a berth allocation in the iron and steel industry is much different from that at container terminals and by now we have not found any models and algorithms that clearly formulate and solve this special problem in the iron and steel industry. However, besides the Baosteel, raw material docks are constructed and used in many large-scale iron and Steel Corporations, for example, Wuhan Iron and Steel Corporation in China, China Steel Corporation in Taiwan, and Kawasaki steel plant in Japan. So there is a practical need for researches on this topic. In the next several sections, we present our model and algorithm for a berth allocation problem in the Baosteel.

2. PROBLEM DESCRIPTION AND MODEL

2.1 Problem assumptions

Like other studies in the berth allocation problem, we make the following general assumptions:

(a) All of the ships are ready at the beginning of the planning horizon.

(b) Resources in docks are available at the beginning of the planning horizon.

(c) Every ship must be serviced once and exactly once.

(d) No preemption is allowed.

However, besides the general characteristics, our problem has some special features that are summarized as follows.

(1) Berth allocation has a very close relation with materials loaded on incoming ships.

(2) More than one ship can share the same berth at the same time if the total length of such ships does not exceed the quay length of the berth.

(3) If the length of a ship is greater than the quay length of one berth, it can occupy several berths that belong to the same dock.

(4) The processing time of each ship is dependent on the berth it is processed.

(5) Ships can be allocated to berths with acceptable physical conditions such as water depth and quay length.

The general and special features form constraints of the following model.

2.2 Model

In order to fully express such features, we first make some classification of ships. Let W be the set of all incoming ships during the planning horizon, we can divide the set into subsets below, according to the materials loaded on these ships.

W_1 – The set of ships that can only be processed at the main material dock;

W_2 – The set of ships that can only be processed at the auxiliary dock;

W_3 – The set of ships that can be serviced at either the main material dock or the auxiliary dock;

W_4 – The set of oil ships.

Based on the above definition, we have $W = (1, \ldots, N) = W_1 \cup W_2 \cup W_3 \cup W_4$, where N is the total number of incoming ships.

In our problem, berths do not always service one ship each time. A bulk ship may occupy more than one berth and one berth may be shared by several small ships. This makes the problem difficult to be formulated and solved if we still use a general berth as our processor. Therefore one concept of "berth type" is introduced to the complicated and practical berth allocation problem in the Baosteel. Berth type is defined as follows:

(1) Only the berths that are adjacent with each other can be made up of one berth type.

(2) Unloaders installed on the berths in the same berth type are considered as the resources of this berth type.

(3) One berth type must have the same berths and unloaders.

(4) If a berth cannot be included in any other berth type, it is a dependent berth type.

Let Y be the set of all of berth types in the raw material docks, and then we can divide this set into four subsets according to the situation of raw material docks in the Baosteel.

Y_1 – The set of berth types equipped with single-arm bridge-type ship unloaders at the main material dock;

Y_2 – The set of berth types equipped with double-arm bridge-type unloaders at the main raw material dock;

Y_3 – The set of berth types equipped with single-arm unloaders at the auxiliary dock;

Y_4 – The set of berth types for oil ships at the heavy oil dock.

Thus from the above definition, we have $Y = (1, \ldots, M) = Y_1 \cup Y_2 \cup Y_3 \cup Y_4$, where M is the total number of berth types.

To model the problem, the entire planning horizon and the total length of each berth type are divided into small units so that all the time and length parameters, such as processing times, due dates and ship lengths, are of integer values. The following additional symbols are used for defining the problem parameters and variables.

T – Time set, $T = \{1, \ldots, K\}$, where K is the total number of planning periods in the planning horizon;

p_{ij} – The processing time of ship i on berth type j;

e_{ij} – The percentage of the total length of berth type j that is occupied by ship i;

m_{ij} – The number of unloaders that is used to process ship i on berth type j;

d_i – The due date of ship i;

l_i – The length of ship i (including the horizontal safety length);

wd_j – The water depth in berth type j;

dr_i – The draft depth of ship i (including the vertical safety length);
q_{jk} – The percentage of the total length of berth type j that is available at time k;
cn_{jk} – The available unloader number of berth type j at time k.
Decision variables:

$$x_{ijk} = \begin{cases} 1 & \text{if ship } i \text{ occupies berth type } j \text{ at time } k \\ 0 & \text{otherwise} \end{cases}$$

i =1, , N; j = 1,…, M; k = 1, …, K;

$$y_{ijp} = \begin{cases} 1 & \text{if ship } i \text{ occupies location } p \text{ on berth type } j \\ 0 & \text{otherwise} \end{cases}$$

i =1, , N; j = 1,…, M; p = 1, …, bl_j, where bl_j is the quay length of berth type j;

$$z_{ijq} = \begin{cases} 1 & \text{if ship } i \text{ occupies unloader } q \text{ on berth type } j \\ 0 & \text{otherwise} \end{cases}$$

i =1, , N; j = 1,…, M; q = 1, …, cn_j, where cn_j is the number of unloaders on berth type j;

$$g_{ij} = \begin{cases} 1 & \text{if ship } i \text{ occupies berth type } j \\ 0 & \text{otherwise} \end{cases}$$

i =1, , N; j = 1, …, M.
c_i – The departure time of ship i, i =1, …, N;
s_{ij} – The starting berthing location of ship i when it is serviced on berth type j, i =1, …, N; j = 1, …, M;
u_{ij}– The first unloader allocated to ship i when it is serviced on berth type j, i =1, …, N; j = 1, …, M.

Objective function Minimize the total tardiness can exempt or reduce expensive demurrage charged to docks, the main target of the berth allocation problem, which can be expressed as:

$$\min \sum_{i \in \Omega} \max\{0, c_i - d_i\}$$

Resource constraints

$$\sum_{i \in \Omega} e_{ij} x_{ijk} \le q_{jk} \qquad \forall j \in \Psi,\ \forall k \in T \tag{1}$$

$$\sum_{i \in \Omega} m_{ij} x_{ijk} \le cn_{jk} \qquad \forall j \in \Psi, \forall k \in T \tag{2}$$

$$\sum_{j \in \Psi} (wd_j - dr_i) g_{ij} \ge 0 \quad \forall i \in \Omega \tag{3}$$

Constraints (1) ensure that the assigned berth length on each berth type must be less than its total berth length at any time. Constraints (2) ensure that the assigned number of unloaders on each berth type cannot be more than the total number of unloaders. Constraints (3) state that each ship can occupy one berth type whose water depth is deeper than the draft of the ship.

Continuity constraints

$$\sum_{k=c_i-p_{ij}+1}^{c_i} x_{ijk} = p_{ij} g_{ij} \qquad \forall i \in \Omega, \forall j \in \Psi \tag{4}$$

$$\sum_{p=s_{ij}}^{s_{ij}+l_i-1} y_{ijp} = l_i g_{ij} \qquad \forall i \in \Omega, \forall j \in \Psi \tag{5}$$

$$\sum_{q=u_{ij}}^{u_{ij}+m_{ij}-1} z_{ijq} = m_{ij} g_{ij} \qquad \forall i \in \Omega, \forall j \in \Psi \tag{6}$$

Constraints (4) assure that a ship must be serviced at a berth type during an uninterrupted time period. Constraints (5) indicate that a ship can be serviced at some berth type only when it has been berthed entirely, that is to say, it can not start discharging if only part of this ship occupies the allocated berth type. Constraints (6) state that a ship is processed by a given fixed number of consecutive unloaders, m_{ij}, simultaneously in material unloading operation.

Berth type constraints

$$\sum_{j \in \Psi} x_{ijk} \le 1 \qquad \forall i \in \Omega,\ \forall k \in T \tag{7}$$

$$\sum_{j \in \Psi} g_{ij} = 1 \qquad \forall i \in \Omega \tag{8}$$

$$\sum_{j \in \Psi_1 \cup \Psi_2} g_{ij} = 1 \qquad \forall i \in \Omega_1 \tag{9}$$

$$\sum_{j \in \Psi_3} g_{ij} = 1 \qquad \forall i \in \Omega_2 \tag{10}$$

$$\sum_{j \in \Psi_4} g_{ij} = 1 \qquad \forall i \in \Omega_4 \tag{11}$$

Constraints (7) guarantee that each ship can occupy at most one berth type at any time. Constraints (8) guarantee that one ship can be serviced at one and only one berth type. Constraints (8) and (9) ensure that the ships which should be serviced at the main raw material dock must occupy the berth type at the main raw material dock. Constraints (8) and (10) restrict that the ships which should be serviced at the auxiliary raw material dock must occupy the berth type at the auxiliary material dock. Constraints (8) and (11) guarantee that the ships which should be serviced at the heavy oil dock must occupy the berth type at the heavy oil dock.

Variable *constraints*

$x_{ijk} \in \{0, 1\}$, i =1, …, N; j = 1,…, M; k = 1, …, K (12)
$y_{ijp} \in \{0, 1\}$, i =1, …, N; j = 1,…, M; p = 1, …, bl_j(13)
$z_{ijq} \in \{0, 1\}$, i =1, …, N; j = 1,…, M; q = 1, …, cn_j(14)
$\gamma_{ij} \in \{0, 1\}$, i =1, …, N; j = 1,…, M (15)
c_i, s_{ij}, u_{ij} are integer numbers, i =1, …, N; j = 1,…, M(16)

3. LOWER BOUND BASED ON LAGRANGIAN RELAXATION

In this section we develop a Lower bound mechanism based on Lagrangian relaxation (Fisher, 1981) of resource constraints that will contribute to our later branch and bound algorithm. The Lagrangian relaxation can be both independently used as a near optimal solution algorithm (Luh, *et al*. 1998) and treated as a method to provide lower bounds, which has been successfully used in many problems.

3.1 *Model of the relaxation problem*

Relaxing constraints (1) and (2) with the non-negative Lagrange multipliers u_{jk} and v_{jk}, the relaxed problem is formulated as follows.

(LR)

Minimize Z_L , with $Z_L(c_i,d_{ijk}) \equiv$

$$\sum_{i\in\Omega}\max\{0,c_i-d_i\}+\sum_{j\in\Psi}\sum_{k\in T}u_{jk}(\sum_{i\in\Omega}e_{ij}d_{ijk}-q_{jk}) + \sum_{j\in\Psi}\sum_{k\in T}v_{jk}(\sum_{i\in\Omega}m_{ij}d_{ijk}-cn_{jk}) \tag{17}$$

subject to constraints (3)-(16), and u_{jk}, $v_{jk} \geq 0$, $j \in \Psi$, $k \in T$.

This problem can be decomposed into sub-problems, each for one ship. The sub-problem for ship i, $i \in \Omega$, is given below.

(LR_i)

Minimize $Z_{Li}(c_i,d_{ijk})$ =

$$\max\{0,c_i-d_i\}+\sum_{j\in\Psi}\sum_{k\in T}u_{jk}e_{ij}d_{ijk} + \sum_{j\in\Psi}\sum_{k\in T}v_{jk}m_{ij}d_{ijk} \tag{18}$$

subject to (3)-(16), and u_{jk}, $v_{jk} \geq 0$, $j \in \Psi$, $k \in T$.

3.2 *Solving the sub-problem*

The sub-problem for ship i can be solved optimally as follows:

Step1. For all the combinations of berth type j which can be occupied by ship i and time k, where $j \in \Psi$, $k \in T$, perform the following procedure:

(1) Locate the ship on berth type j at time k;

(2) Evaluate the objective value of the equation (18).

Step2. Find the combination that has the minimal objective value.

Step3. Let the start time of ship i equal to the time in the combination that has the minimal objective value.

3.3 *Finding a feasible solution*

The decomposed problems are easily infeasible because the constraints (1) and (2) are relaxed. To find a feasible solution, we propose a heuristic for the lagrangian relaxation problem. The task of the heuristic is to allocate appropriate resource for each ship including a continuous berth length and consecutive unloaders which belong to the berth type selected for it in section 3.2. We design an occupied resources queue, an available continuous berth length queue and an available consecutive unloader queue for each berth type to describe the dynamic resources allocation process. Each of the three queues is sorted in non-decreasing or increasing order of the due dates, start berth locations and start unloader numbers, respectively. In order to make full use of resources at each berth type, the first appropriate continuous berth length and consecutive unloaders in the corresponding queues are chosen.

3.4 *Update* Lagrangian multipliers

We use a subgradient method to obtain the values of the Lagrangian multipliers $\{u_{jk}, v_{jk}\}$. Let u^r_{jk} and v^r_{jk} be the multipliers at iteration r, and let Z^{UB}, updated by the heuristic algorithm in section 3.3, denote the upper bound on the minimum value of the total tardiness time. After the relaxation problem is solved by the method in section 3.2, the obtained solution Z^{LB} is the lower bound on the optimal objective function. l_r is a step length and initially set to be 2. If five consecutive iterations fail to improve the lower bound, l_r will be havled. Then the multipliers can be determined by the following recursive formulation:

$$u^{r+1}_{jk} = \mathrm{Max}\{0, u^r_{jk} + l_r\frac{(Z^{UB}-Z^{LB})A^r_{jk}}{\sum_{j\in\Psi}\sum_{k\in K}{A^r_{jk}}^2}\} \tag{19}$$

$j = 1,\ldots, M$, $k = 1,\ldots, K$

$$v^{r+1}_{jk} = \mathrm{Max}\{0, v^r_{jk} + l_r\frac{(Z^{UB}-Z^{LB})B^r_{jk}}{\sum_{j\in\Psi}\sum_{k\in K}{B^r_{jk}}^2}\} \tag{20}$$

$j = 1,\ldots, M$, $k = 1,\ldots, K$

Where ${A_{jk}}^r = \sum_{i\in\Omega}e_{ij}d_{ijk} - q_{jk}$, $j = 1,\ldots, M$, $k = 1,\ldots, K$;

${B_{jk}}^r = \sum_{i\in\Omega}m_{ij}d_{ijk} - cn_{jk}$, $j = 1,\ldots, M$, $k = 1,\ldots,$ K.

4. BRANCH AND BOUND ALGORITHM

In this section we give our branch and bound algorithm to obtain an optimal solution for the presented berth allocation problem. Special attention is given to the description of our branching rules that are related to the special characteristics of raw material docks in the Baosteel and the process that are used to prune the search tree.

Before we construct a search tree for the stated problem, some definitions and rules are necessary. The search tree contains two types of nodes, each of which corresponds with a partial berth allocation. If the node is a circle node, ship j is assigned on the same berth type with its father node; however, if the node is a square node, ship j is assigned on a new berth type. The following are the necessary rules for the algorithm to develop a search tree for the considered problem.

Rule 1. At level 1, there are at most N square nodes which represent the current berth type is 1 and no circle nodes can occur in such a level.

Rule 2. At other levels, there are at most $2*(N - l + 1)$ nodes which may be square nodes or circle nodes.

Rule 3. In each branch on the tree, the number of the square nodes must less than or equal to the total number of berth types.

Rule 4. One ship can appear exactly once in each branch.

Rule 5. At level l, if the last ship in the partial sequence of the current active node is serviced at berth type *j*, then only those ships which can occupy berth type *j* or berth type *j*+1 may generate both or either of one circle node and one square node. If there are no such ships, this node is eliminated.

Rule 6. Before a square node can be generated, all of the ships, whose maximum berth type that can process them is smaller than the berth type that is represented by this square node, must have been arranged before.

All of the above six rules are used in our branch and bound algorithm to eliminate the nodes that can not improve the value of the total tardiness. Our branch and bound has the similar procedure as the one by Liu and MacCarthy (1991), in which a depth-first search strategy is used and the immediate descendants of the active node are explored in non-decreasing order of their lower bounds. If the lower bound at one node is larger than or equal to the minimal objective value obtained by all of the feasible solutions having been found, this node and all of its brothers are eliminated.

5. COMPUTATION EXPERIMENTS

To test the performance of the algorithm and study the characteristics of solutions, computational experiment has been carried out with real problem instances, which were collected from the raw material management centre in the Baosteel.

5.1 The selection of experiment data

According to the definition of berth types, raw material docks in the Baosteel can be classified into four berth types, which have been shown in figure 1.

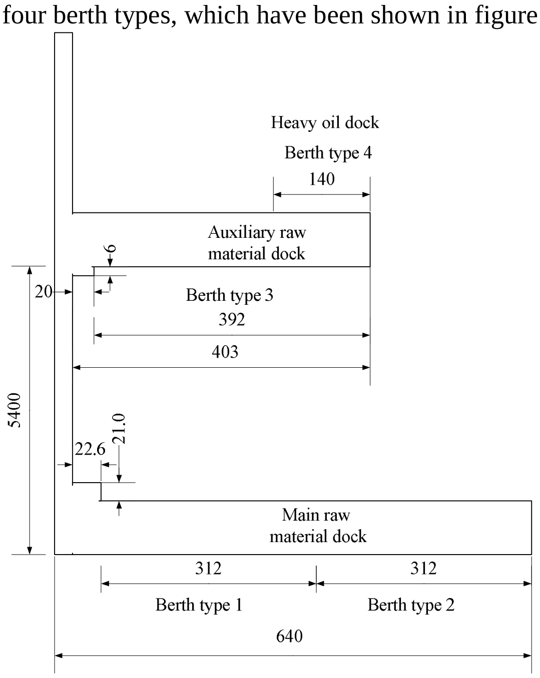

Fig. 1. Berth types in the Baosteel

Ships allocated to berth type 1 must be serviced on the river side of the dock. Because the unloaders in berth type 2 are double-arm, ships allocated to this type can be serviced on either side of the main raw material dock. Ships allocated to berth type 3 must be discharged on the river side of the auxiliary raw material dock. Ships allocated to berth type 4 must carry out their discharging operation at heavy oil dock locating on the land side of the auxiliary raw material dock.

In the Baosteel, incoming ships are directed by the raw material day planning to pull into docks. Although the day planning is made once everyday, it is related to all about ten incoming ships on the average within a three-day horizon. The probabilities of the ship type being ore, auxiliary materials, coal, and heavy oil are 38.8%, 19.0%, 35.8%, and 6.4% respectively. Since the minimal time unit of the real berth planning in the Baosteel is one hour, the planning horizon T is selected as 72 for a three-day horizon. We select ten examples from the real day berth planning to test the performance of the presented branch and bound algorithm. Ship number varies from 8 to 14 so that they can include different dock situations.

5.2 Computation results

In the following we select an example with 14 ships to illustrate the computation result. The resulting berth allocation is drawn in figures 2 and 3, in which figure 2 shows the situation in the main raw material dock and figure 3 shows the one in auxiliary raw material dock and heavy oil dock. In figure 2 and 3, the numeric value at the left corner of each rectangle indicates a ship. Numbers enclosed by circles denote unloaders allocated to each ship. And the two rectangles filled with dots in figure 3 represent the tardy ships.

Table 1 summarizes the results for all the ten examples. Within the time horizon of three days, the average total number of ships is 11.1.

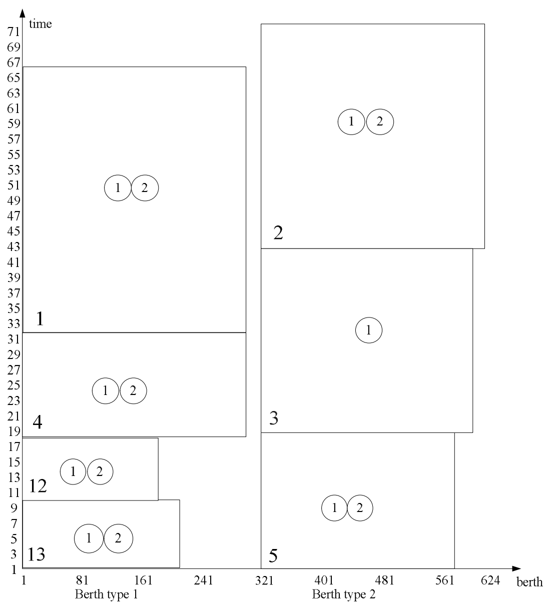

Fig. 2. Berth allocation in the raw material dock

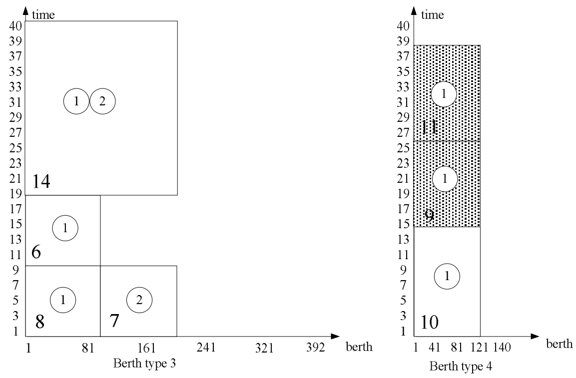

Fig. 3. Berth allocation in auxiliary material dock and heavy oil dock

Table 1. Ten examples of Baosteel day material planning

Problems	Ships	Tardiness	Tardiness number	Running time (s)
1	8	0	0	0.92
2	9	0	0	2.97
3	10	17	3	59.0
4	10	7	2	16.44
5	10	7	2	15.66
6	11	12	2	5.25
7	12	13	3	23.7
8	13	18	4	118.3
9	14	15	2	264.23
10	14	12	2	443.38

From the data shown in table 1, we can conclude that:
(1) With the number of ships increasing, tardiness time and tardiness number are increasing. This can be explained that the resource is relatively scarce when the incoming ships are congested on one day. This makes the berth allocation more important to minimize the tardiness and reduce demurrage cost.
(2) Due date has a great impact on the running time of the branch and bound algorithm. This may be because tardiness is large when due date is small so that the objective value is large. This results in larger running time as the difference of upper bound and lower bound is large.
(3) Optimal solutions can be obtained for real three-day berth allocation problems in the Baosteel in an acceptable running time. And such a solution size is in accordance with the ship number in a general day berth planning of the Baosteel.

6. CONCLUSIONS

Berth allocation is a key factor to improve the productivity of the docks. Unlike previous studies using Expert System to berth allocation, this paper formulates a novel integer programming model that fully expresses the characteristics of real berth allocation problem in the Shanghai Baosteel Complex. A lower bound scheme based on Lagrangian Relaxation of resource capacity constraints is incorporated into a branch and bound algorithm, along with new branching rules, to solve this berth allocation problem. Computation experiment on ten real berth allocation problems in the Baosteel indicates that the branch and bound algorithm can obtain an optimal solution within a reasonable time.

ACKNOWLEDGEMENTS

This research is partly supported by National Natural Science Foundation for Distinguished Young Scholars of China (Grant No. 70425003), National Natural Science Foundation of China (Grant No. 60274049) and (Grant No. 70171030), Fok Ying Tung Education Foundation and the Excellent Young Faculty Program of the Ministry of Education, China. The authors would like to thank raw materials center in Baosteel for providing a lot of production information and data.

REFERENCES

Chen J, Lee C-Y. (1999). General Multiprocessor Task Scheduling. *Naval Research Logistics*, **46**, 57-74.

Fisher ML (1981). Lagrangian relaxation method for solving integer programming. *Management Science*, **27**, 1-18.

Imai A, Nishimura E, Papadimitriou S (2003). Berth allocation with service priority. *Transportation Research Part B*, **37**, 437-457.

Kao C, Li D-C, Wu C, Tsai, C-C(1990). Knowledge-based approach to the optimal dock arrangement. *International Journal of Systems Science*, **21**, 2209-2215.

Kao C, Lee HT (1996). Discrete time parallel-machine scheduling: a case of ship scheduling. *Engineering Optimization*, **26**, 287-294.

Li C-L, Cai X, Lee, C-Y (1998). Scheduling with multiple-job-on-one-processor pattern. *IIE Transactions*, **30**, 433-445.

Liu J, MacCarthy BL (1991). Effective heuristics for single machine sequencing problem with ready times. *International Journal of Production Research*, **29**, 1521-1533.

Luh PB, Gou L, Zhang Y, Nagahora T, Tsuji M, Yoneda K, Hasegawa T, Kyoya Y, Kano T (1998). Job shop scheduling with group-dependent setups, finite buffers and long time horizon. *Annals of Operations Research*, **76**, 233-259.

Suzuki K, Taka Y, Hiyama T (1996). Coastwise transportation planning and administration system. *Kawasaki steel technical report*, **33**, 30-37.