Journal of Engineering Science and Technology Review 9 (2) (2016) 93-98

Research Article

JOURNAL OF
Engineering Science and Technology Review

www.jestr.org

Ship Block Transportation Scheduling Problem Based on Greedy Algorithm

Chong Wang [*,1,3], **Yun-sheng Mao** [1,2], **Bing-qiang Hu** [1,2] , **Zhong-jie Deng** [1,2] **and Jong Gye Shin** [3,4]

[1] *Wuhan University of Technology, Key Laboratory of High Performance Ship Technology, Wuhan, China*
[2] *Wuhan University of Technology, School of Transportation, Wuhan, China*
[3] *Department of Naval Architecture and Ocean Engineering, Seoul National University, Seoul, Korea*
[4] *Research Institute of Marine System Engineering, Seoul National University, Seoul, Korea*

Received 10 January 2015; Accepted 20 May 2016

Abstract

Ship block transportation problems are crucial issues to address in reducing the construction cost and improving the productivity of shipyards. Shipyards aim to maximize the workload balance of transporters with time constraint such that all blocks should be transported during the planning horizon. This process leads to three types of penalty time: empty transporter travel time, delay time, and tardy time. This study aims to minimize the sum of the penalty time. First, this study presents the problem of ship block transportation with the generalization of the block transportation restriction on the multi-type transporter. Second, the problem is transformed into the classical traveling salesman problem and assignment problem through a reasonable model simplification and by adding a virtual node to the proposed directed graph. Then, a heuristic algorithm based on greedy algorithm is proposed to assign blocks to available transporters and sequencing blocks for each transporter simultaneously. Finally, the numerical experiment method is used to validate the model, and its result shows that the proposed algorithm is effective in realizing the efficient use of the transporters in shipyards. Numerical simulation results demonstrate the promising application of the proposed method to efficiently improve the utilization of transporters and to reduce the cost of ship block logistics for shipyards.

Keywords: Greedy Algorithm; Ship Block Transportation; Block scheduling.

1. Introduction

In modern shipbuilding, a ship consists of multiple erection blocks. The “block” is the most important work-in-process aspect of the shipbuilding process. For example, a very large crude oil carrier with deadweight of 350,000 tons comprises around 120 blocks. These blocks undergo a series of construction processes such as assembly, pre-outfitting, painting, and outfitting in the factory or stockyard before erection in the dock. Block weight is generally between 200 and 300 tons, and even the individual block is over 500 tons. Ship block transportation between the factory and the stockyard mainly relies on transporters (Figure 1). For a large shipyard, nearly 500 block transportation tasks need to be completed daily. Thus, the efficiency of shipbuilding must be improved through optimal block transportation scheduling. Related studies on the optimal problem of ship block transportation scheduling in shipyards are limited, although decreasing the construction cost and improving productivity are possible through efficient transporter operation. Therefore, an efficient method is necessary to solve the aforementioned problem. The present study focuses on how to solve this problem efficiently for shipyards. The rest of this paper is organized as follows. Section 2 explains the main distinction between the aforementioned problem and the classical traveling salesman problem (TSP). Section 3 introduces the current research situation regarding this topic. Section 4 derives a mathematical model to find the optimal solution and proposes a heuristic algorithm based on greedy algorithm. Section 5 presents a computational numerical simulation experiment to evaluate the performance of the proposed algorithm. Finally, conclusions are summarized and remarks on further study are provided in Section 6.

Fig.1. Example of a transporter for moving blocks in shipyards

2. Description of the problem

When the number of transporters is fixed, the type of transporters is the same, and all blocks are predetermined to be delivered by a specific transporter; the scheduling problem is similar to a multiple TSP (m-TSP) [1][2][3][4]. In such a problem, each block is considered as a location and transporters are regarded as traveling salesmen. However, ship block transport problems in shipyards have a few differences with m-TSPTW because of the different numbers and different types of transporter.

* E-mail address: chriswang@whut.edu.cn
ISSN: 1791-2377

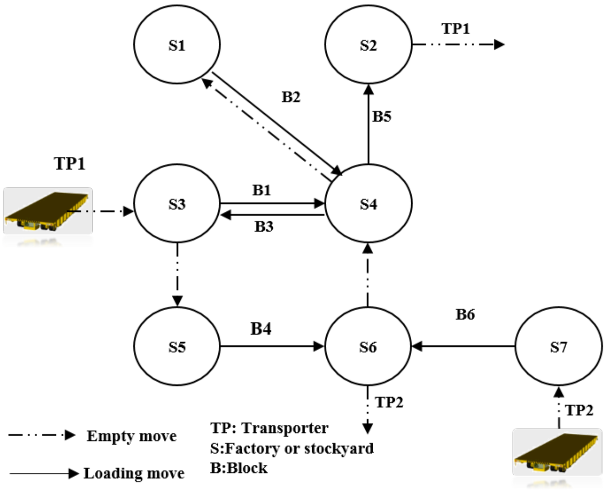

Fig.2. Description of ship block transportation problem in shipyards

In the example shown in Figure 2, B1-B2-B5 can only be transported by TP1 because of the capacity of transporter restrictions. As a result, TP1 goes through the factory or S3-S4-S1-S4-S2 to complete the transportation of B1-B2-B5. In this process, TP1 from the workplace S3 transports B1 to the destination of B1 (S4) because the destination of B1 (S4) and the departure of the next transported B2 (S1) are different. Thus, TP1 must empty travel to the departure of B2 (S1). The same TP2 sequentially goes through workplace S7-S6-S4-S3-S5 to transport B6-B3-B4 to their destinations. The origin and destination of ship block transport problems are different. Therefore, this problem is absolutely different from that of the classical TSP.

3. State of the art

Only a few studies on ship block transportation scheduling problem have been published, although the shipbuilding cost can be reduced through efficient transporter operation. The main researchers are from South Korea, which is one of the global industry leaders in shipbuilding. Joo et al. (2006)[5] first studied the block transportation scheduling of single-type transporters in shipyards. They proposed a heuristic algorithm to minimize the sum of total weighted logistic time of blocks moved by transporters. However, the assumption regarding single-type transporters is inconsistent with the shipyard objective facts. Park et al. (2013)[6] addressed the transporter scheduling problem in ship assembly block operation management by transforming this problem into parallel machine scheduling with sequence-dependent setup time and precedence constraints with the same assumption of single-type transporters. Roh et al. (2011)[7] expanded the block transportation scheduling problem of Joo et al. (2006)[5] when multi-type transporters are used. Although good solutions were not obtained, this extension preemptively decided the allocation rule of blocks to transporters and determined the sequence rule of blocks for each transporter. Kim and Joo (2012)[8] considered a block transportation scheduling problem with multi-type transporters as in the work of Roh and Cha (2011)[7]. They determined the blocks assigned to each transporter and the sequence of blocks for each transporter at the same time. However, the computation time increases significantly when the number of blocks and transporters increases. Finding a good solution in actual large-sized problems is difficult. Joo et al. (2014) [9] expanded the block transportation scheduling problem of Roh and Cha (2011)[7] and Kim and Joo (2012)[8] by generalizing the block delivery restriction on the multi-type transporter. Two metaheuristic algorithms based on a genetic algorithm and a self-evolution algorithm are proposed to avoid the occurrence of infeasible solutions caused by the block delivery restriction. The present study primarily aims to improve the efficiency of the solution process for assigning blocks to available transporters and sequencing blocks for each transporter simultaneously. The algorithm based on greedy algorithm is designed to solve the realistic scale problem with an optimal feasible solution.

4. Methodology

4.1 Mathematical optimization model

To calculate the sum of empty transporter travel time, delay time, and tardy time, $t(i)$ is set as the timing to accomplish the transport task of block i.

(1)If the transport task of block is the first task of the transporter, then $o(i)=1$, $t(i)=0$.

(2) Otherwise, $t(i)$ is the time when the transporter completed the previous block.

Empty transporter travel time ($ET(i)$) is incurred when the current position of the transporter and the origin of the next handling block are different.

(1)If the transport task of the block i is the first task of the transporter, then $ET(i)$ is the time that transporters move from the initial position to the starting position of block i.

(2) Otherwise, $ET(i)$ is the time that transporters move from the destination node of the previous block to the start node of the next block.

$$ET(i)=\begin{cases} d(ON(j),SN(i))/VE(j) & c(i)=j, o(j)=1 \\ d(AN(i'),SN(i))/VE(j) & \begin{matrix} c(i)=c(i')=j, \\ o(i)=o(i')+1\le 2 \end{matrix} \end{cases},$$

Delay time ($DT(i)$) is incurred when the blocks are collected by transporters after the required preparation time, which is represented as follows:

$$DT(i)=\max\left(t(i')+ET(i)-ST(i),0\right),$$
$$c(i')=c(i), o(i')=o(i)-1.$$

Tardy time ($LT(i)$) is incurred when the block is delivered after the required delivery time, which is represented as follows:

$$LT(i)=\max\left(ST(i)+DT(i)+UT(i)+\frac{d\left(SN(i),AN(i)\right)}{VF\left(c(i)\right)}+OT(i)-AT(i),0\right)$$
$$t(i)=ST(i)+DT(i)+UT(i)+\frac{d\left(SN(i),AN(i)\right)}{VF\left(c(i)\right)}+OT(i).$$

Block i should be transported by the transporter that has higher weight capacity than the weight of the block, that is,

$$w(i) \le W(c(i))$$

Decision variables

For the ith block:

$c(i) = j$: Block i is transported by transporter j .

$o(i) = k$: Block i is the kth transportation task of the transporter.

The evaluation index of the scheduling plan is as follows:

$ET(i)$: empty transporter travel time;

$DT(i)$: delay time;

$LT(i)$: tardy time.

Parameters

N : Based on the assumption that the workshop or block stockyard yard is simplified as a node, N is the total number of nodes.

D : Distance matrix among nodes, $D = [d(i,j)]_{N \times N}$.

Based on the assumption that n blocks need to be transported in one day, and the properties of the block transportation task are as follows:

$SN(i)$: Start node of block i , $SN(i) \in \{1,2,..,N\}$;

$AN(i)$: Destination node of block i , $AN(i) \in \{1,2,..,N\}$ ($SN_i \ne AN_i$);

$ST(i)$: Scheduled start time of block i ;

$AT(i)$: Scheduled due time of block i ;

$UT(i)$: Loading time of block i ;

$OT(i)$: Unloading time of block i ; and

$w(i)$: Weight of block i .

Based on the assumption that m transporters exist, each transporter possesses the following attributes:

$ON(j)$: Initial position of transporter j ;

$W(j)$: Deadweight of transporter j ;

$VE(j)$: No-load speed of transporter j ; and

$VF(j)$: Full-load speed of transporter j .

4.2 Objective function

The model aims to minimize the sum of total $ET(i)$, $DT(i)$, and $LT(i)$, which are represented as follows:

$$\min_{c(i),o(i)} \sum_{i=1}^{n} \left[ET(i) + DT(i) + LT(i) \right],$$
$$s.t. w(i) \le W(c(i)) .$$

The optimization variables of this model are shown as follows:

$c(i)$: Assignment variables of block transportation; and

$o(i)$: Execution sequence variables of block transportation.

First, $c(i)$ must satisfy the following constraint:

$$c(i) \in \{1,2,...,m\} .$$

Then, $c(i)$ and $o(i)$ were closely related. Based on the assumption that the set of the transportation task of transporter j is $C_j = \{i : c(i) = j\}$, for any $i' \ne i'' \in C_j$, $o(i') \ne o(i'')$.

The execution sequence variables of block transportation are as follows: $o(i)$ starts at 1 and then continues to the consecutive numbers, $O_j = \{o(i) : i \in C_j\} = \{1,2,...,|O_j|\}$.

4.3 Optimization algorithm

In this section, the main objective is to allow the original optimization problem to approximately transform the TSP. Then, an algorithm is designed to solve the problem by considering the practical significance of the problem in the shipbuilding industry.

4.3.1 Model simplification and assumption

The optimization model aims to weigh the sum of total $ET(i)$, $DT(i)$, and $LT(i)$. In fact, the delay time to accomplish the previous block transportation has a significant influence on the delay time of the next block transportation. In this study, all the previous block transportations are assumed to be on time without the impact. The concern is how to select the next block transportation task more effectively. In this case, the sum of total $ET(i)$, $DT(i)$, and $LT(i)$ for the next block transportation task is called penalty time. For any block transportation task i and j , the penalty time is $\omega(i,j)$, which means that block i is transported first and then block j is transported. This process is defined as a transfer among the block transportation tasks. Therefore, the block transportation tasks were regarded as vertices of a graph. The penalty time of the transfer was used as edge weights to construct a directed graph. All the blocks should be transported within a reasonable period, and thus, this problem is very close to TSP. Simultaneously, the initial positions of each transporter as a virtual node are added to the directed graph, considering that they had a significant effect on the penalty time of block transportation tasks. Furthermore, the penalty time of a transfer is associated with the speed of the transporter, and the dead weight of transporters is limited. Accordingly, this problem was considered in the classification with the same speed of transporters and according to load capacity to bring this problem closer to the general TSP.

The following assumptions were made: m transporters are available with the same no-load and full-load speeds; the no-load speed and full speed is VE and VF , respectively; the minimum load of the transporters is W ; n blocks are available with weight $w(i)$ less than W .

The block transportation tasks n were used as nodes. The penalty time $\omega(i,j)$ of the transfer was used as edge weights to construct a directed graph. The penalty time is the sum of $ET(i)$, $DT(i)$, and $LT(i)$ based on the assumption that the block i was transported only in time and the block j was transported immediately.

$$\omega(i,j) = \overline{ET}(i,j) + \overline{DT}(i,j) + \overline{LT}(i,j) ;$$

$$\overline{ET}(i,j) = d(AN(i), SN(j)) / VE ;$$

$$\overline{DT}(i,j) = \max(AT(i) + \overline{ET}(i,j) - ST(j), 0) ;$$

$$\overline{LT}(i,j)=\max\begin{pmatrix} ST(j)+\overline{DT}(i,j)+UT(j)+ \\ d(SN(j),AN(j))/VF+OT(j)-AT(j),0 \end{pmatrix};$$

$\omega(i,i)=\infty$ means that tasks are not transferred by themselves.

The following m virtual nodes are available: $k=n+1,n+2,...,n+m$. The main reasons for adding the virtual nodes are as follows:

a) The penalty time of transporting other blocks is different because the initial positions of the transporter are different.

b) Before the virtual nodes are set up, the path of one transporter in the graph to delivery blocks is only an ordinary path and not a loop. To construct a loop, the virtual node is introduced. After adding virtual nodes, the process of transporter to move one block are as follows: starting from the initial position, followed by the completion of other tasks, and then back to the original node. However, returning to the initial node is in fact unnecessary. The weight of returning to the original node is set as 0 after the other transportation tasks have been accomplished.

The constraints of the penalty time are as follows:

(1) The penalty time from the virtual node k to other block transportation tasks is equal to the penalty time of the transporter moving from its initial position to other block transportation tasks.

$$\omega(k,j)=\overline{ET}(k,j)+\overline{DT}(k,j)+\overline{LT}(k,j)$$

$$\overline{ET}(k,j)=d(ON(k-n),SN(j))/VE$$

$$\overline{DT}(k,j)=\max(AT(i)+\overline{ET}(k,j)-ST(j),0).$$

(2)The penalty time between virtual nodes is $\omega(k_1,k_2)$. $\omega(k_1,k_2)=\infty$, $k_1,k_2\in\{n+1,n+2,...,n+m\}$. In other words, the transporter does not move from its initial position to the initial position of the other transporters to avoid constituting a loop between virtual nodes.

(3)The penalty time from the other block transportation j to the virtual node k is $\omega(j,k)$. $\omega(j,k)=0$, $j\in\{1,2,...,n\}$, $k\in\{n+1,n+2,...,n+m\}$. In particular, the transporter finishes all transportation tasks after completing any task j .

The following decision matrix is set: $X=(x_{ij})_{(n+m)\times(n+m)}$; $x_{ij}=1$ means that the transporter moves block i and then moves block j ; $x_{ij}=0$ means that the transporter does not move between block i and block j .

According to the preceding discussion, the optimization problem is expressed as follows:

$$\min \sum_{i=1}^{n+m}\sum_{j=1}^{n+m}\omega(i,j)x_{ij}$$

$$s.t.\begin{cases}\sum_{i=1}^{n+m}x_{ij}=1, j=1,2,...,n+m, \\ \sum_{j=1}^{n+m}x_{ij}=1, i=1,2,...,n+m\end{cases}$$

where $\sum_{i=1}^{n+m}x_{ij}=1$ indicates that the in-degree of each node is 1; $\sum_{j=1}^{n+m}x_{ij}=1$ indicates that the out-degree of each node is 1. Both of them ensure that the transfer path selection consists of a number of loops. For example, one transporter and two blocks are constructed in the directed graph (Figure 3).

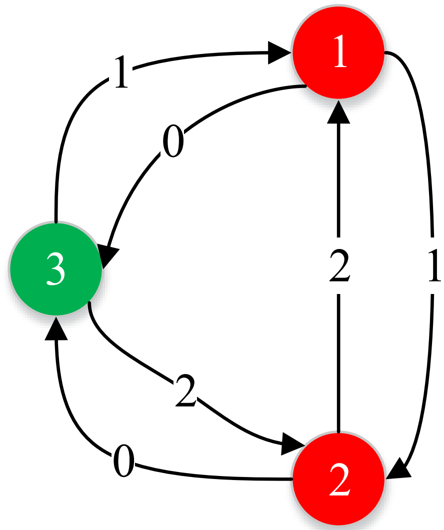

Fig.3. Directed graph of one transporter and two blocks

Ordinary nodes 1 and 2 correspond to transportation tasks 1 and 2, respectively. Virtual node 3 is the initial position of transporter 1. The weight of the graph edge is the penalty time of a transfer. The definitions are as follows:

(1) The penalty time of transporter 1 from the initial position to the completion of the block transportation task 1 is 1.

(2) The penalty time of transporter 1 from the initial position to the completion of the block transportation task 2 is 2.

(3) The penalty time of transporter 1 to complete the block transportation tasks 1 and 2 is also 1.

(4) The penalty time of transporter 1 to complete the block transportation tasks 2 and 1 is also 2.

This study aims to find the loop in which the penalty time is the minimum. Clearly, the penalty time of "3→1→2" is 2, and the penalty time of "3→2→1" is 4. Therefore, the path of "3→1→2" is better. This path means that transporter 1 moves from the initial position to complete block transportation task 1 and then completes block transportation task 2.

4.3.2 Greedy algorithm optimization

In this section, the transporters are classified according to load capacity to design a greedy algorithm for obtaining an approximate solution. In the shipyard, transporters have a variety of different specifications. The reasonable scheduling plan is that the heavyweight blocks are transported by using heavy-load transporters, and the lightweight blocks are transported by using light-load transporters. Based on this classification rule, the approximation algorithm was designed as follows:

Based on the assumption that M transporter specifications are present, the load capacity is $W_1,W_2,...,W_M$.

Step 1. Set $k = 1$, marking all block transportation tasks that are not allocated.

Step 2. If $k < M$, then go to step 3; otherwise, proceed to step 4.

Step 3. Select the block transportation tasks with weight less than W_k, and construct the assignment problem using the Hungarian algorithm for calculation. To obtain the optimal solution as the load of W_k transporter task, mark the arranged tasks as allocated status; then, proceed to step 5.

Step 4. Select the block transportation tasks with weight less than in the task without an assigned transporter, and construct TSP using Matlab for calculation. To obtain the optimal solution as the load of transporter task, proceed to step 5.

Step 5. If $k < M$, then update $k = k + 1$, and return to step 2. Otherwise, stop the procedure.

5. Result analysis and discussion

5.1 Numerical simulation

In this study, the actual production data of D shipyard are used in the numerical simulation. A total of 25 transporters are used in this shipyard, and the details are shown in Table 1.

Table 1. Detailed information on transporters in D shipyard

Load capacity	100 tons	200 tons	300 tons	350 tons	500 tons
Number	3	3	7	7	5
No-load speed	250 m/min	250 m/min	200 m/min	180 m/min	150 m/min
Full-load speed	135 m/min	120 m/min	100 m/min	75 m/min	50 m/min

This shipyard has 16 factories and 7 stockyards, that is, 23 nodes are obtained. The distance between the nodes is the same as the shipyard layout. A total of 250 block transportation tasks are present. The loading and unloading times of block transportation correspond to normal distribution from 3 min to 5 min. The scheduled start time is normal distribution from 0 min to 400 min. The formula for the scheduled due time is as follows:

$$AT(i)=ST(i)+{d(i,j)}/{UT(i)}+R$$

R is a normal distribution from 30 min to 50 min. In other words, the scheduled due time of each block transportation task includes transporter loading travel time, 10 min loading/unloading block time, and waiting time from approximately 20 min to 40 min. The numerical experiment environment consists of CPU: Intel (R) Core (TM) i5-4460 CPU 3.20 GHz; RAM: 4.0 G; 64-bit operating system; and Matlab R2014a.:\

5.2 Simulation result and discussion

According to the experimental environment, the solving time is only 0.761791 s by using the greedy algorithm. The transportation task of each transporter is shown in Figure 4.

The terms in Figure 4 are defined as follows:

LIT: No empty travel before the start of the transportation task, and the block arrives at the destination on time.

LDT: No empty travel before the start of the transportation task, but the block is collected by the transporter after the required preparation time.

LDTT: No empty travel before the start of the transportation task, but the block is collected by the transporter after the required preparation time and is delivered after the required delivery time.

EIT: Empty travel before the start of the transportation task, and the block arrives at the destination on time.

EDT: Empty travel before the start of the transportation task, but the block is collected by the transporter after the required preparation time.

EDTT: Empty travel before the start of the transportation task, but the block is collected by the transporter after the required preparation time and is delivered after the required delivery time.

In Figure 4, LIT, EIT, and EDT mean that the block arrives at the destination on time. Figure 4 indicates that most of the block transportation tasks are completed on time, and only a few of them are delivered after the required delivery time. Furthermore, the load of each transportation is balanced. The entire block transportation task is accomplished before 16:00.

6. Conclusions

This study aimed to determine when the transporter should move and which transporter should move the block from its source factory or stockyard to its destination, with the minimum sum of empty transporter travel time, delay time, and tardy time to minimize the cost of ship block logistics. A new mathematical model was established, which was a hybrid model based on TSP and AP. A heuristic method based on the greedy algorithm for the optimal solution was derived to determine the assignment of blocks for available transporters and the transportation sequence of blocks for each transporter. The result indicated that most of the block transportation tasks were completed on time. Moreover, the workload for transporters was balanced appropriately. According to the computational results, the proposed algorithm is a novel solution for ship block transportation scheduling problem in an actual shipyard. The proposed algorithm has significant potential to promote transporter efficiency to reduce the cost of ship block logistics for shipyards. In the future, an intelligent scheduling system for ship block transportation scheduling will be developed based on this method.

Acknowledgements

This work was supported in part by the scholarship from China Scholarship Council (CSC) under the Grant CSC NO.201406950021. The authors would like to express their sincere gratitude to the supporter.

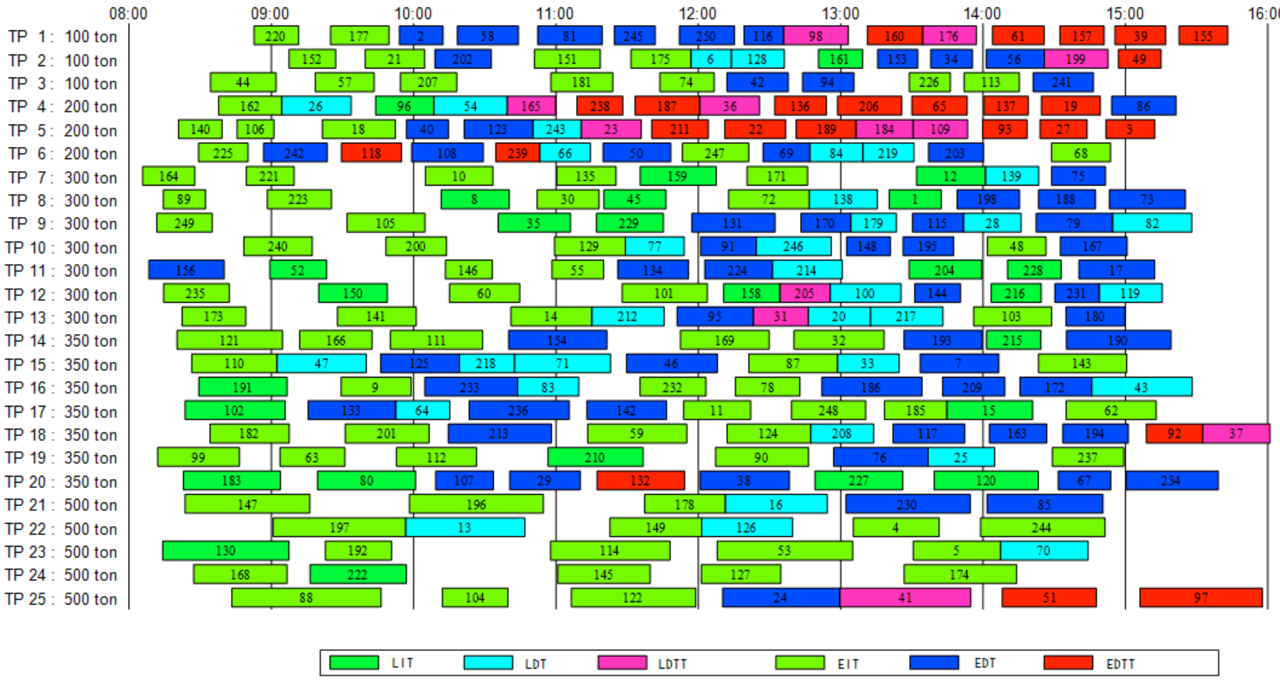

Fig.4. Result of block assignment and block transportation t sequence.

References

1. Calvo, R.W., "A new heuristic for the traveling salesman problem with time windows". *Transportation Science*, 34(1), 2000, pp.113-124.
2. Baker, E.K., "Technical note—an exact algorithm for the time-constrained traveling salesman problem". *Operations Research*, 31(5), 1983, pp.938-945.
3. Bianco, L., Dell'Olmo, P. and Giordani, S., "The Traveling Salesman Problem with Precedence Constraints and Binary Costs". *System Modelling and Optimization*,21(7), 1996, pp. 543-550.
4. Dumas, Y., Desrosiers, J., Gelinas, E. and Solomon, M.M., "An optimal algorithm for the traveling salesman problem with time windows". *Operations research*, 43(2), 1995, pp.367-371.
5. Joo, C.M., Lee, W.S., Koo, P.H. and Lee, K.B., "Transporter scheduling for block transportation in shipbuilding". *Journal of the Korea Management Engineers Society*, 11(3), 2006, pp.169-179.
6. Park, C. and Seo, J., "A GRASP approach to transporter scheduling for ship assembly block operations management". *European Journal of Industrial Engineering*, 7(3), 2013, pp.312-332.
7. Roh, M.I. and Cha, J.H., "A block transportation scheduling system considering a minimisation of travel distance without loading of and interference between multiple transporters". *International Journal of Production Research*, 49(11), 2011, pp.3231-3250.
8. Kim, B.S. and Joo, C.M., "Ant colony optimisation with random selection for block transportation scheduling with heterogeneous transporters in a shipyard". *International journal of production research*, 50(24), 2012, pp.7229-7241.
9. Joo, C.M. and Kim, B.S., "Block transportation scheduling under delivery restriction in shipyard using meta-heuristic algorithms". *Expert Systems with Applications*, 41(6), 2014, pp.2851-2858.
10. Bondy, John Adrian, and Uppaluri Siva Ramachandra Murty. "Graph theory with applications". *Bulletin of the American Mathematical Society*, 290(1), 1976, pp.120-125.
11. Edmonds, J., "Matroids and the greedy algorithm. Mathematical programming". *Mathematical Programming*,1(1), 1971, pp.127-136.
12. Pentico. "Assignment problems: A golden anniversary survey". *European Journal of Operational Research*, 176(2),2007,pp.774-793.
13. Bednorz, Witold. "Advances in greedy algorithms", Wienna: *I-Tech Education and Publishing KG*, 14(6), 2008, pp.18-26.
14. Tao, N., Jiang, Z. and Qu, S., "Assembly block location and sequencing for flat transporters in a planar storage yard of shipyards". *International Journal of Production Research*, 51(14), 2013, pp.4289-4301.