J Mar Sci Technol (2010) 15:190–200
DOI 10.1007/s00773-009-0080-3

ORIGINAL ARTICLE

# Optimal scheduling of block lifting in consideration of the minimization of traveling distance while unloaded and wire and shackle replacement of a gantry crane

**Myung-Il Roh · Kyu-Yeul Lee**

Received: 26 August 2009 / Accepted: 12 November 2009 / Published online: 16 December 2009


**Abstract** Nowadays, shipyards are making every effort to efficiently manage their resources such as gantry cranes, transporters, and block stock yards. The scheduling of block lifting for a gantry crane has been manually performed by an expert shipyard manager for many years. Such a practice, however, can lead to an undesirably long time to produce scheduling plans. In addition, the quality of the scheduling plans may not be optimal. To improve the overall process, a block lifting scheduling system for a gantry crane was developed in this study by using optimization techniques. A block lifting scheduling problem was first formulated as a multiobjective optimization problem. For a gantry crane, minimization of the traveling distance while unloaded and wire and shackle replacement between block lifting was considered. An optimization algorithm based on the genetic algorithm was then proposed and implemented so as to solve the problem. To evaluate the efficiency and applicability of the developed system, the system was applied to an actual block lifting scheduling problem in a shipyard. Compared to results achieved by manual scheduling by an expert manager, the results of this work show that blocks can be more efficiently lifted by a gantry crane when the developed system is applied.

**Keywords** Lock lifting scheduling · Gantry crane · Genetic algorithm · Multiobjective optimization problem · Production planning · Shipbuilding

M.-I. Roh (✉)
School of Naval Architecture and Ocean Engineering,
University of Ulsan, Daehak-Ro 102, Nam-Gu,
Ulsan 680-749, Korea
e-mail: miroh@ulsan.ac.kr

K.-Y. Lee
Department of Naval Architecture and Ocean Engineering,
Research Institute of Marine Systems Engineering,
Seoul National University, Shinlim-Dong,
Seoul 151-742, Korea
e-mail: kylee@snu.ac.kr

**List of symbols**

| | |
|---|---|
| $\alpha$ and $\beta$ | Weighting factors |
| $D_i$ | Traveling distance of block $i$ from the initial position to the target position |
| $d_{i,i+1}$ | Traveling distance of the crane while unloaded from block $i$ to the next block $i + 1$ |
| $d_{i,W}$ | Traveling distance of the crane while unloaded from block $i$ to the wire and shackle stockyard for replacement |
| $d_{W,i+1}$ | Traveling distance of the crane while unloaded from the wire and shackle stockyard to block $i + 1$ after replacement |
| $f_i$ | Lifting finish time of block $i$ |
| $l_i$ | Available lifting time of block $i$ (start of the lifting time slot) |
| $N$ | Total number of blocks |
| $p_j$ | Lifting start time of block $j$ of high priority |
| $p_k$ | Lifting start time of block $k$ of low priority |
| $r_{i,i+1}$ | Whether wire and shackle replacement is necessary when traveling from block $i$ to block $i + 1$ (unnecessary = 0, necessary = 1) |
| $R_u$ | Penalty coefficients |
| $s_i$ | Lifting start time of block $i$ |
| $T_e$ | Daily work finish time of the shipyard |
| $T_i$ | Traveling time of block $i$ from the initial position to the target position |

$t_{i,i+1}$ Traveling time of the crane while unloaded from block $i$ to the next block $i + 1$

$t_{i,W}$ Traveling time of the crane while unloaded from block $i$ to the wire and shackle stockyard for replacement

$T_r$ Necessary time for wire and shackle replacement

$t_{W,i+1}$ Traveling time of the crane while unloaded from the wire and shackle stockyard to block $i + 1$ after replacement

$u_i$ Planned lifting finish time of block $i$ (end of the lifting time slot)

$V_{il}$ Moving speed of the crane while unloaded in the longitudinal direction of the building dock

$V_{it}$ Moving speed of the crane while unloaded in the transverse direction of the building dock

$V_{ll}$ Moving speed of the crane while loaded in the longitudinal direction of the building dock

$V_{lt}$ Moving speed of the crane while loaded in the transverse direction of the building dock

## 1 Introduction

### 1.1 Background

A ship is a huge structure comprising a large number of hull structural parts. Thus, in contrast to the approach used in automobile manufacturing, a ship cannot be constructed all at once. The ship is first divided into a number of small blocks called assembly blocks at the design stage. Each assembly block is made by joining subassemblies and parts in the assembly shop near the building dock. Large blocks, which are called erection blocks, are made by joining several small blocks together. Then, the erection blocks are moved onto the building dock and welded to each other according to a suitable sequence, which is called the block erection, to complete the final assembly of the ship. The construction process of a ship is similar to the process in which a large product is made up of a number of parts like Lego blocks.

The most important resources of a shipyard are the building docks and cranes that are used to lift the erection blocks of the ship. In general, several ships are concurrently constructed in one building dock. Thus, if the traveling distance while unloaded of the gantry crane becomes long when a certain ship is under construction, the occupation time of the building dock by the subject ship increases, and this consequently has a big impact on the productivity of the shipyard. A gantry crane known as a Goliath crane is used for lifting the erection blocks in the building dock. The gantry crane has the shape of a gate. At either end of the girder of the crane are its legs, and these travel on rails which are laid on the ground. Figure 1 shows an example of a gantry crane for lifting erection blocks.

The gantry crane lifts the erection blocks, moves them to the target position, and then lowers them, placing the blocks in the preerection (PE) area near the building dock one by one At this time, the traveling distance of the crane while unloaded changes according to the block lifting sequence, i.e., the sequence in which the crane lifts, relocates, and lowers the blocks. Figure 2 shows an example of the traveling distance of the crane while unloaded in a block lifting sequence. As shown in this figure, if block 2 is moved after block 1, the traveling distance becomes shorter than that for the reverse sequence. Thus, to minimize the occupation time of the building dock by the ship under construction, optimal management of crane operations is important because the increase in cost and time can in turn cause delays in the overall production schedule.

Thus, it is difficult to systematically manage the operation of a gantry crane over a long period. Usually, an expert manager in the department of crane operations manually determines the block lifting sequence based on experience. However, it may be quite difficult for the manager to optimally determine the scheduling of block lifting because scores of blocks need to be lifted every day. If there is a decision-assisting system available for this task, the manager can produce an optimal block lifting schedule by investigating more alternatives in a shorter time.

### 1.2 Related study

Research related to optimal scheduling of block lifting in the shipbuilding industry is not prevalent, even though it is


**Fig. 1** Example of a gantry crane for lifting ship erection blocks

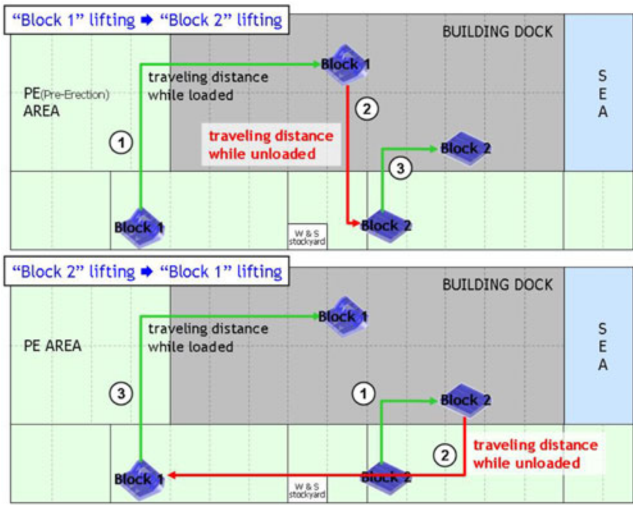


**Fig. 2** The traveling distance while unloaded of a gantry crane depends on the block lifting sequence. *W&S* wire and shackle

possible to increase the productivity of a shipyard and to reduce the building cost of a ship through efficient crane operation. However, some research on optimal crane operation in a container terminal has been conducted.

Daganzo [1, 2] and Peterkofsky and Daganzo [3] proposed a heuristic algorithm for optimal management of a quay crane in a container terminal. Kim and Kim [4] presented an algorithm for optimal management of a transfer crane for a given working schedule and container loading condition. Similarly, Ng and Mak [5] proposed an algorithm for the optimal management of a transfer crane for loading and unloading work having different setup times. In addition, Kim and Park [6] developed an algorithm for determining the optimal storage space in order to minimize the number of relocations of containers when loading and unloading the containers. Kim and Kim [7] proposed an optimal routing algorithm to minimize the travel time of a transfer crane and the setup time at a yard bay in a container terminal. The researchers presented an algorithm for determining the optimal visit sequence in the yard bay which the transfer crane should follow and the optimal number of containers which the transfer crane should carry at the yard bay.

However, there are limitations in applying these existing studies as they do not directly apply to the block lifting scheduling of a gantry crane which is used for ship construction in a shipyard. First, the traveling method of a gantry crane is different from those of quay cranes and the transfer crane in a container terminal. A gantry crane can travel restrictively on the rails laid near the building dock; in contrast, quay cranes and transfer cranes can travel more freely in a larger space. In addition, quay cranes and transfer cranes carry only ISO standardized cargoes, called sea containers, having a fixed size and maximum weight; therefore, they do not need to replace wires, shackles, or other items during the transportation of containers. The gantry crane, however, carries nonstandardized cargoes called erection blocks. These erection blocks vary in size and weight, and thus the replacement of wires and shackles is often needed; consequently, this fact must be considered in the scheduling of block lifting.

In this study, a decision-assisting system for optimal scheduling of block lifting was developed based on an optimization technique in order to solve the subject problems. The problem for optimal scheduling of block lifting was mathematically formulated and an optimization algorithm based on the genetic algorithm was proposed to effectively solve the problem. To evaluate the efficiency and applicability of the developed system, the system was then applied to actual shipyard problems and the results were compared with those of manual scheduling.

## 2 Optimal block lifting scheduling problem

### 2.1 Block lifting scheduling problem

In the block lifting scheduling problem examined in this study, the goal is to find the optimal block lifting sequence for a gantry crane (hereinafter, a crane) in shipyards. The design objective is to maximize the total time the crane spends lifting blocks through a minimization of the traveling distance of the crane while unloaded. Wire and shackle replacement during the operation of the crane should also be minimized. These objectives will be accomplished while satisfying the constraints of the lifting time slot, the sequential erection priority (lifting priority) of the blocks, and others. The input data given by an expert manager are indicated below.

#### *2.1.1 Data on the gantry crane*

- Specifications of the crane (e.g., the maximum lifting weight, the moving speed in the longitudinal and transverse directions, service time)
- Initial position of the crane.

#### *2.1.2 Data on the blocks*

- Total number and ID of the blocks to be lifted by the crane
- Weight of each block
- Necessary time for lifting and lowering each block
- Necessary time for joining (welding) each block
- Initial position and target position of each block

- Lifting time slot (start to end time available) of each block
- Lifting priority among the blocks
- Total number and specification of wires and shackles for lifting each block.

#### 2.1.3 Miscellaneous data

- Information on the shipyard (e.g., building dock layout and lot number)
- Position of wire and shackle stockyards
- Information on the working time (e.g., daily work finish time, idleness time of the crane, replacement time of wires and shackles).

Here, the lot number indicates an ID for each position in the building dock and PE area.

### 2.2 Mathematical formulation of the block transportation scheduling problem

The problem described in the previous section is mathematically formulated as follows:

$$\text{Minimize } F_1 = \sum_{i=0}^{N-1} \{(1 - r_{i,i+1}) \cdot t_{i,i+1} + r_{i,i+1} \cdot (t_{i,W} + t_{W,i+1})\}, \text{ total traveling time of the crane while unloaded} \tag{1}$$

and

$$\text{Minimize } F_2 = \sum_{i=0}^{N-1} (r_{i,i+1} \cdot T_r), \text{ total necessary time for wire and shackle replacement} \tag{2}$$

Subject to

$$g_1 = l_i - s_i \le 0 \tag{3}$$

$$g_2 = f_i - u_i \le 0 \tag{4}$$

$$g_3 = p_j - p_k \le 0 \tag{5}$$

$$g_4 = f_N - T_e \le 0 \tag{6}$$

$i = 0, \ldots, N-1$ and $j, k = 1, \ldots, N$

In the above expressions, Eq. 3 shows that block $i$ should be lifted after the start of the lifting time slot ($l_i$ is the start of the lifting time slot). Equation 4 shows that block $i$ should be lifted before the planned lifting finish time of the block ($f_i$ is the end of the lifting time slot). Equation 5 indicates that the blocks should be lifted according to their lifting priority. This means that blocks of a higher priority are lifted earlier. Equation 6 indicates that the lifting work for all blocks should finish before the daily work finish time of the shipyard.

Since the problem described by Eqs. 1–6 has two objective functions ($F_1$ and $F_2$), it is a multiobjective optimization problem. However, it can be converted to a single-objective optimization problem by using the weighting method [8]. Applying weighting factors ($\alpha$ and $\beta$) to the objective functions ($F_1$ and $F_2$) in Eqs. 1 and 2 yields the following form:

Minimize

$$\begin{aligned} F &= \alpha \cdot F_1 + \beta \cdot F_2 \\ &= \alpha \cdot \sum_{i=0}^{N-1} \{(1 - r_{i,i+1}) \cdot t_{i,i+1} + r_{i,i+1} \cdot (t_{i,W} + t_{W,i+1})\} \\ &\quad + \beta \cdot \sum_{i=0}^{N-1} (r_{i,i+1} \cdot T_r) \end{aligned} \tag{7}$$

The weighting factors $\alpha$ and $\beta$ are equivalent to the tradeoff between the total lifting time of the blocks ($F_1$) and the total necessary time for wire and shackle replacement of the crane ($F_2$). Therefore, the ratio of $\alpha$ and $\beta$ can vary the optimization result. In this study, suitable values for $\alpha$ and $\beta$ in each example were chosen by trial and error, so that $F_1$ and $F_2$, the objective functions, have the same effect on the optimization result.

Using a penalty function method, the constrained optimization problem as defined in Eqs. 3–7, can be converted to an unconstrained optimization problem:

Minimize

$$\begin{aligned} F' &= \alpha \cdot \sum_{i=0}^{N-1} \{(1 - r_{i,i+1}) \cdot t_{i,i+1} + r_{i,i+1} \cdot (t_{i,W} + t_{W,i+1})\} \\ &\quad + \beta \cdot \sum_{i=0}^{N-1} (r_{i,i+1} \cdot T_r) + \sum_{u=1}^{4} \{R_u \cdot \max(g_u, 0)\}. \end{aligned} \tag{8}$$

## 3 Proposed algorithm for the block lifting scheduling problem

### 3.1 Overview of the proposed algorithm

The optimization algorithm proposed in this study is based on the genetic algorithm (GA) that is currently widely used for facility layout design, scheduling, and also in other fields. The GA is classified as an evolutionary search and optimization technique that is suitable for evolutionary design processes. The GA attempts to find the best solution by generating a collection ("population") of potential solutions ("individuals"). Through selection, crossover, and mutation operations, it is expected that more optimal solutions will be generated from the current set of potential solutions. This iteration continues until the algorithm finds an acceptable solution. Details about the GA can be found in the literature [9, 10]. The proposed algorithm based on the GA in this study was implemented in the C++ language and is shown schematically in Fig. 3.

### 3.2 Representation of the scheduling of block lifting

Scheduling of block lifting can be represented as a chromosome by an encoding process in the GA. The chromosome can likewise be represented as the scheduling of block lifting by a decoding process. In this study, a method to model the scheduling of block lifting in a chromosome of one-dimensional array type, including the block lifting sequence for the crane, is used; the method is shown in Fig. 4. The number of genes in the chromosome equals the number of blocks to be lifted by the crane. Figure 4 shows an example of the scheduling of block lifting (for ten blocks) together with a corresponding representation of the chromosome.

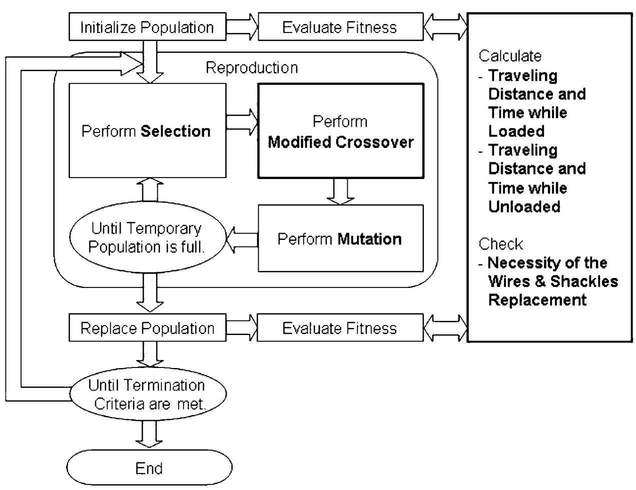


**Fig. 3** Flow diagram of the proposed algorithm for scheduling of block lifting of a gantry crane

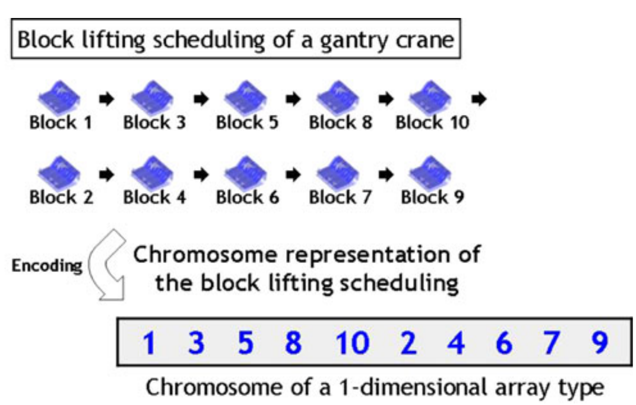


**Fig. 4** Example of the scheduling of block lifting and the corresponding representation of the chromosome

### 3.3 Calculation of the traveling distance of the gantry crane between blocks

The crane can move a block in the longitudinal and transverse directions of the building dock. When moving in the longitudinal direction, the whole body of the crane travels on wheels installed in the legs. When moving in the transverse direction, the body of the crane is fixed and only the trolley on the upper body moves. Figure 5 shows the movement of a crane alongside the building dock of a shipyard. According to this movement pattern of the crane, the traveling distance between starting to move two blocks that are processed consecutively, i.e., the traveling distance while loaded and the traveling distance while unloaded, can be simply calculated using the rectilinear distance method. Suppose that the moving speeds while loaded in the longitudinal and transverse directions are $V_{ll}$ and $V_{lt}$ and the moving speeds while unloaded (idle) in the longitudinal and transverse directions are $V_{il}$ and $V_{it}$, then the traveling time of block 1 ($T_1$) for lifting and the traveling time while unloaded (i.e., the time required to move to block 2 after relocating block 1) can be calculated as shown in Fig. 6.

Meanwhile, the crane picks up a block not via a direct connection but by means of wires, shackles, and lugs. In other words, the wires, shackles, and lugs are necessarily arranged between the crane and the block, as shown in Fig. 7. Here, a lug means a structural part which has already been welded to the block before lifting. The total number and specifications of the wires and shackles change according to the weight of the block to be lifted. If the weight difference between sequent blocks for lifting is large, the wires and the shackles used for lifting the previous block cannot be used for lifting the next block, and

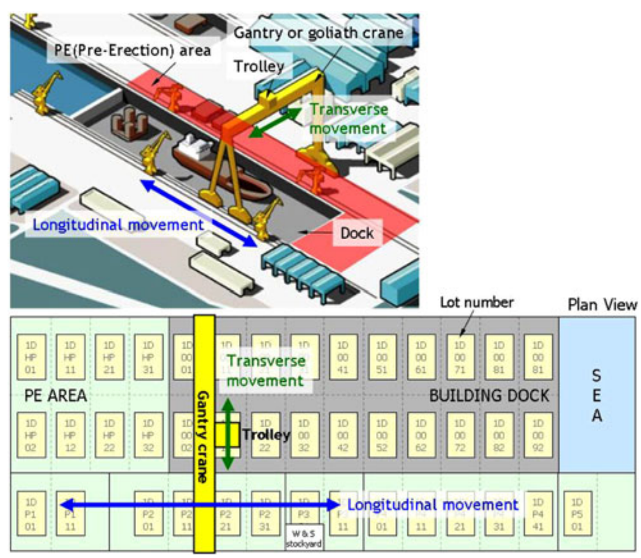


**Fig. 5** Movement of a gantry crane alongside the building dock of a shipyard

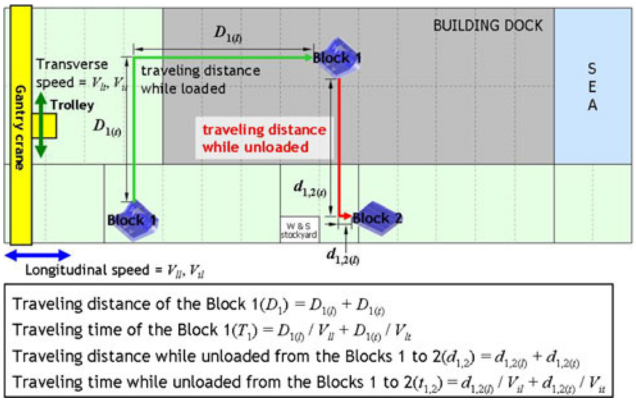


**Fig. 6** Example of calculating the traveling distance and time of a gantry crane using the rectilinear distance method

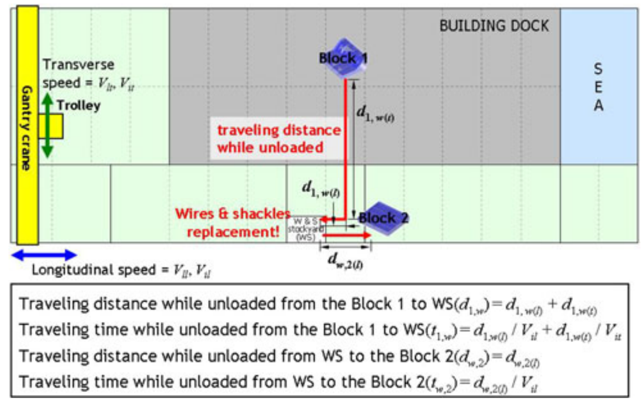


**Fig. 8** Example of calculating the traveling distance of a gantry crane requiring wire and shackle replacement

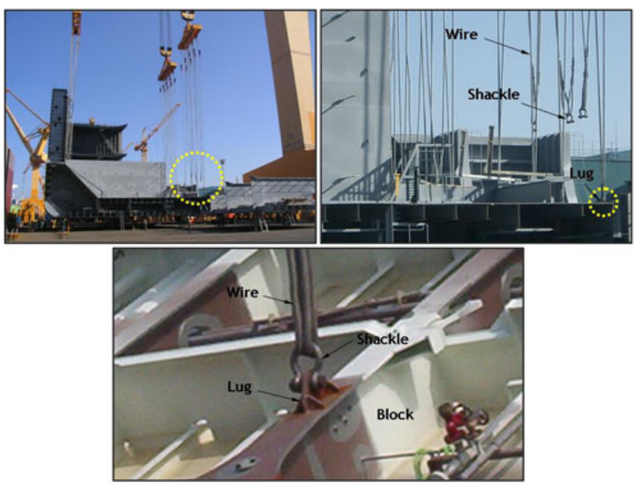


**Fig. 7** Connection between a gantry crane and a block by means of wires, shackles, and lugs

thus they should be replaced before lifting the next block. Accordingly, the crane must visit a wire and shackle stockyard in order to have them replaced. In this case, the traveling distance between blocks should include the traveling distance between the crane's location and the wire and shackle stockyard. Figure 8 shows an example of calculating the traveling distance of the crane considering wire and shackle replacement. As shown in this figure, it is necessary to replace the wires and shackles for lifting block 2 after lifting block 1. Thus, the traveling distance increases due to the visit to the wire and shackle stockyard.

### 3.4 Improved genetic operations

As mentioned above, the proposed algorithm for the scheduling of block lifting is based on the GA. In the GA, three genetic operations, known as selection, crossover, and mutation, are typically used to generate new individuals ("children"). In this study, these operations of the GA were improved to solve efficiently the block lifting scheduling problem. These operations were incorporated into the proposed algorithm.

#### *3.4.1 Selection operation*

The selection operation is a process performed to select two individuals ("parents") from the current population to take part in the next genetic operation. A proportionate selection method was employed, as it is the most popular of the stochastic selection methods. The technique is sometimes called the roulette wheel selection method. In the proportionate selection method, an individual is selected based on selection probability $p_{\text{selection}}(i)$. This is shown in Eq. 9, where $\text{Ft}(i)$ is the fitness value of the $i$th individual:

$$p_{\text{selection}}(i) = \frac{\text{Ft}(i)}{\sum_i \text{Ft}(i)} \tag{9}$$

The probability of selecting an individual from the current population is purely a function of the individual's relative fitness. A fitness function (Ft) for an individual can be formulated as follows:

$$\text{Ft} = -F' \text{ or } \text{Ft} = \frac{1}{F'} \quad (\text{if } F' > 0) \tag{10}$$

In this study, the second formula of Eq. 10 was used. By this selection method, individuals having a higher fitness value will participate in the creation of the next generation more often than those having a lower fitness value.

#### *3.4.2 Crossover operation*

The crossover operation is a process performed to generate new individuals (children) from two individuals (parents) selected by the selection operation. A modified crossover operation is used in this work [11]. The modified crossover operation is based on the assumption that the individual having the higher fitness value of the two parents should pass on more genes to the child. This operation is simultaneously applied to each parent. The integer $s1$ represents the number of genes of the first parent to be replaced with those of the second parent to create the chromosome of the child. The $s1$ genes to be replaced (the s1 positions) in the first parent are randomly selected. This serves to generate the first child. $s1$ is determined based on the first parent's fitness when compared with the second parent's fitness:

$$s1 = \frac{\{\mathrm{Ft}(p1) + \mathrm{Ft}(p2)\} - \mathrm{Ft}(p1)}{\mathrm{Ft}(p1) + \mathrm{Ft}(p2)} \times n(\text{discard decimals}),\ s2 = n - s1 \tag{11}$$

In Eq. 11, $\mathrm{Ft}(p1)$ is the fitness of the first parent, $\mathrm{Ft}(p2)$ is the fitness of the second parent, $s2$ is the number of genes from the first parent to be transmitted to the first child, and $n$ is the number of genes in the chromosome of the first or second parent. From Eq. 11, the parent having the higher fitness has a smaller $s1$ value than the other parent due to the assumption mentioned above. For example, if $\mathrm{Ft}(p1)$ is greater than $\mathrm{Ft}(p2)$, the value of $s1$ is smaller than $s2$, and thus the first parent endows more genes than the second parent to the child. The next step in this operation is for the genes in the $s2$ positions of the first parent to be transmitted to the corresponding positions of the first child. Finally, the genes in the $s1$ positions are reordered according to the order of the corresponding genes in the second parent. The genes are then transmitted to the corresponding positions of the first child. Similar steps are applied to the second parent to generate the second child. Figure 9 shows an example of the modified crossover operation for generating the first and second children.

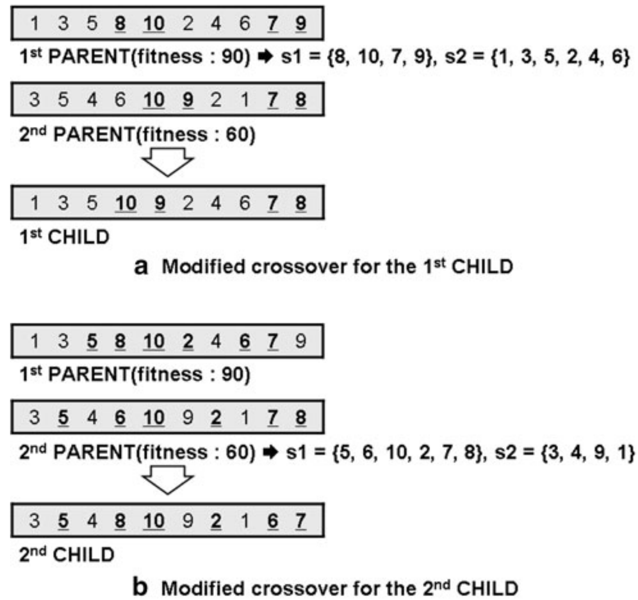


**Fig. 9** Example of the modified crossover operation for generating the first and second children. **a** For the first child, $s1$ (the number of genes to be replaced) is 4 and $s2$ (the number of genes to be kept) is 6. The *arrays* shown for $s1$ and $s2$ represent the requisite number of genes chosen at random from the first parent's chromosome. **b** For the second child (i.e., the procedure is now carried out for the second parent), $s1$ is 6 and $s2$ is 4

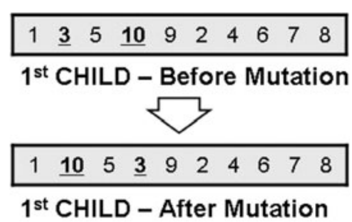


**Fig. 10** Example of the mutation operation applied to the first child

### 3.4.3 Mutation operation

The mutation operation, which can be considered as self-crossing, is used to increase population diversity. The mutation operation provides the possibility of exploring portions of the design space that are not represented in the genetic makeup of the current population. The mutation operation is simultaneously applied to each child generated from the crossover operation. The mutation operation occurs with a very low probability (typically $p_{\mathrm{mutation}} = 0.01$ from Grefenstette's study [12]). Two genes of each child are randomly selected and are exchanged. Shown in Fig. 10 is an example of the mutation operation applied to the first child.

### 3.4.4 Comparative test of the improved genetic operations with conventional genetic operations

To evaluate the efficiency of the improved genetic operations (particularly the modified crossover), a comparative test was performed with conventional genetic operations. For a more accurate comparison, the objective function ($F'$ in Eq. 8), the chromosome structure, and the representation method of the scheduling of block lifting were the same. The testing was performed for one example shown in Table 2, see below, and the maximum number of iterations was set at 500 with a population size of 500. Both genetic operations were run 100 times for the example on a Pentium IV system (2.0 GHz, 1 GB RAM) and the results obtained are summarized in Table 1 where the best among 100 objective function values, the mean of 100 objective function values, and the mean of 100 computation times are shown. From Table 1 it can be seen that the improved genetic operations are superior to the conventional ones, as there are 2.5% better values of mean objective functions produced by the improved genetic operations, and it required 28% less computation time.

### 3.4.5 *Influence of weighting factors in the objective function of the block transportation scheduling problem*

To evaluate the influence of weighting factors $\alpha$ and $\beta$ in the objective function of the block transportation scheduling problem, a parametric test for the weighting factors was performed by varying the values of $\alpha$ and $\beta$. As mentioned, the weighting factors $\alpha$ and $\beta$ are equivalent to the tradeoff between the total lifting time of the blocks ($F_1$ in Eq. 1) and the total necessary time for wire and shackle replacement of the crane ($F_2$ in Eq. 2). Therefore, the ratio of $\alpha$ and $\beta$ can vary the optimization result. By varying the values of $\alpha$ and $\beta$, a number of optima for the problem, called the Pareto optimal set, can be obtained. In this study, the parametric test was performed for one example shown in Table 2 by varying the values of $\alpha$ and $\beta$ from 0 to 1. Figure 11 shows the Pareto optimal set for this example. In this figure, optimum A represents the result obtained when the total traveling time of the crane while unloaded was considered as the objective function ($\alpha = 1.0$ and $\beta = 0.0$). Optimum B represents the result obtained when the total necessary time for wire and shackle replacement was considered as the objective function ($\alpha = 0.0$ and $\beta = 1.0$). Optimum C represents the result obtained when both factors were considered at the same time ($\alpha = 0.67$ and $\beta = 0.33$). These values for optimum C were chosen by trial and error, so that $F_1$ and $F_2$, the objective functions, have the same effect on the optimization result.

**Table 1** Comparison of computational results of the improved genetic operations and conventional genetic operations for 100 runs

| | Result of the improved genetic operations | Result of the conventional genetic operations |
|---|---|---|
| Best objective function value | 105.78 | 105.78 |
| Mean objective function value | 105.78 | 108.43 |
| Mean computation time (s) | 3.38 | 4.71 |

## 4 Development of a block lifting scheduling system

In this study, a block lifting scheduling system was developed based on the proposed algorithm in order to evaluate the efficiency and applicability of the proposed algorithm. Figure 12 shows the configuration of the block lifting scheduling system developed in this study. The developed system consists of four modules: an optimization algorithm module, a graphical user interface (GUI) module, a reporting module, and a visualization module. The optimization algorithm module is based on the GA and a core module for generating the optimal scheduling of block lifting. The GUI module is a tool for providing various input data for the scheduling of block lifting. The reporting module is a tool for making a table of the optimal scheduling of block lifting. Finally, the visualization module is a tool for visualizing the optimal scheduling of block lifting as an animation. Figure 13 is a screenshot of the block lifting scheduling system developed in this study. The developed system is a day-to-day optimization system for an expert manager responsible for crane operations. The expert manager can use the developed system on the day or before the actual block lifting operations.

**Table 2** Comparison of the performance on a specific day resulting from manual scheduling by an expert manager and automatic scheduling by the developed system

| | Result of manual scheduling | Result of the developed system | |
|---|---|---|---|
| | | Before optimization | After optimization |
| Total traveling time | 3 h 20 min | 3 h 41 min | 2 h 38 min |
| Total traveling time while unloaded | 2 h 6 min | 2 h 26 min | 1 h 24 min |
| No. of the wire and shackle replacements | 9 | 11 | 5 |
| Scheduling of block lifting (ID of the blocks) | 80A → 182 → 10C → 63M → 62M → 171 → 152 → 164 → 174 → 161 → RUD → 183 → 192 → 193 → 625 → 626 → 635 → 636 | 174 → 164 → 63M → 62M → 10C → 193 → 183 → 625 → 626 → 80A → 182 → 192 → 152 → 171 → 161 → RUD → 636 → 635 | 10C → 174 → 164 → 62M → 63M → 80A → 182 → 192 → 152 → 161 → 171 → RUD → 183 → 193 → 625 → 626 → 635 → 636 |

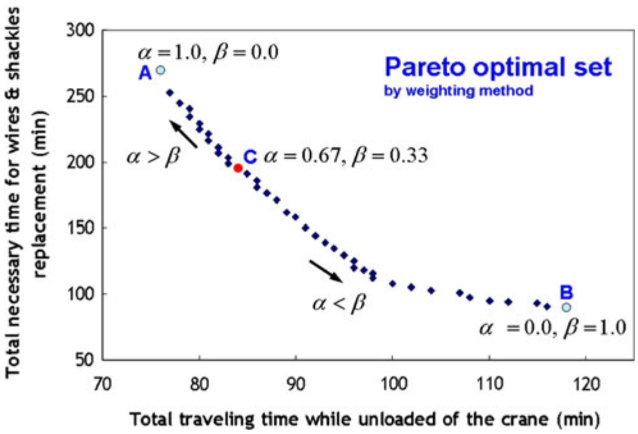


**Fig. 11** Pareto optimal set obtained from the parametric test for the weighting factors $\alpha$ and $\beta$ (defined in Eq. 7)

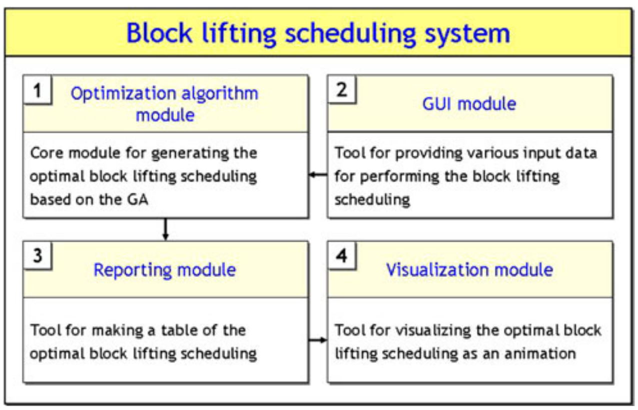


**Fig. 12** Configuration of the block lifting scheduling system developed in this study. *GA* genetic algorithm, *GUI* graphical user interface

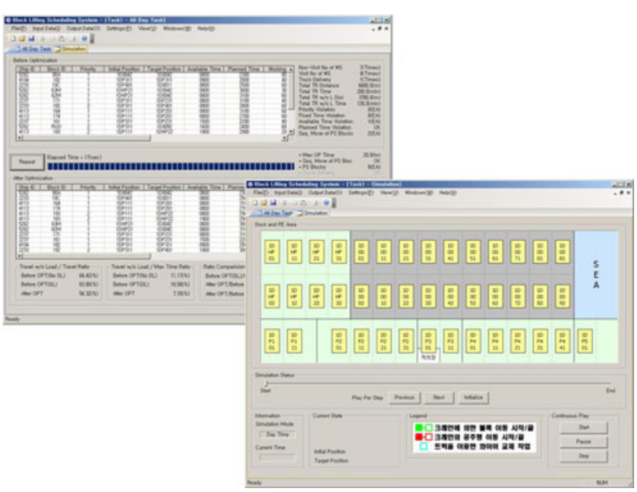

**Fig. 13** Screenshot of the block lifting scheduling system developed in this study

## 5 Application of the block lifting scheduling system

To evaluate the efficiency and applicability of the developed system, the system was applied to actual shipyard problems and the results were compared with manual scheduling by a manager of the department of crane operations.

Table 2 shows the comparison of the performance on a specific day derived from the manual scheduling by an expert manager and the automatic scheduling by the developed system (after optimization). In this example, the total number of blocks to be lifted by the crane was 18. The automatic scheduling was performed on a Pentium IV system (2.0 GHz, 1 GB RAM) with the maximum number of iterations set at 500 with a population size of 500. In addition, $\alpha = 0.67$ and $\beta = 0.33$ were used for the weighting factors in this example. It took about 3.13 s to obtain the optimum and about 3.02e−5 s for one evaluation of the objective function. As shown in the table, for manual scheduling, the total traveling time and the traveling time of the crane while unloaded were 3 h 20 min and 2 h 6 min, respectively. Thus, for the manual scheduling, the traveling time while unloaded amounted to about 63% of the total traveling time of the crane, and about 10.5% of the daily work time (20 h) of the crane. From this, we can see that the majority of the service time of the crane is wasted when manual scheduling is used. In contrast, for automatic scheduling by the developed system, the total traveling time and the traveling time of the crane while unloaded were 2 h 38 min and 1 h 24 min, respectively. Thus, for automatic scheduling, the traveling time while unloaded amounted to about 53% of the total traveling time of the crane, and about 7% of the daily work time of the crane. Regarding the number of wire and shackle replacements during lifting of the blocks, nine replacements were necessary in the case of manual scheduling, whereas the developed system required only five replacements. The last row of the table shows the scheduling of block lifting by manual scheduling and by automatic scheduling. Here, a 3-character code represents the ID of the blocks. From these results, we can see that the developed system can reduce the total traveling time of the crane and minimize the number of the wire and shackle replacements during operation. Thus, it is possible to more efficiently use the crane with the developed method.

Table 3 compares the performance of the manual scheduling by an expert manager and the automatic scheduling by the developed system over a six-day period. The test environment was the same as for Table 2. The values of the idleness ratio in the table denote the ratio (%) between the traveling time of the crane while unloaded and daily work time (20 h) of the crane. From these results, we can see that the developed system can considerably reduce

**Table 3** Comparison of the idleness ratio of the crane and the number of wire and shackle replacements over a 6-day period

| | Result of manual scheduling | | Result of the developed system | |
|---|---|---|---|---|
| | Idleness ratio (%) | No. of wire and shackle replacements | Idleness ratio (%) | No. of wire and shackle replacements |
| Day #1 (19 blocks) | 15.8 | 9 | 7.9 | 5 |
| Day #2 (18 blocks) | 11.1 | 7 | 5.7 | 4 |
| Day #3 (22 blocks) | 20.3 | 14 | 13.2 | 9 |
| Day #4 (18 blocks) | 12.2 | 9 | 9.8 | 6 |
| Day #5 (20 blocks) | 13.2 | 10 | 7.8 | 6 |
| Day #6 (17 blocks) | 13.1 | 8 | 8.1 | 5 |
| Average | 14.3 | 10 | 8.8 | 6 |

the traveling time of the crane while unloaded and the number of wire and shackle replacements.

## 6 Conclusions and future work

In this study, an optimization algorithm for the scheduling of block lifting was proposed in order to increase the productivity of a gantry crane being used for ship construction in a shipyard, and a block lifting scheduling system was developed based on the proposed algorithm. First, the block lifting scheduling problem was mathematically formulated by considering the minimization of the traveling distance of the crane while unloaded and minimizing the wire and shackle replacements required during lifting of the blocks. An optimization algorithm based on the GA was implemented in order to solve the problem. Finally, the developed system was applied to an actual block lifting scheduling problem in a shipyard. The computational results were compared to those of manual scheduling by an expert manager in the department of crane operations. From this comparison, the applicability of the developed system was shown and discussed. When compared to manual scheduling by a manager, we can see that the developed system can reduce the total traveling time of the crane and minimize the number of wire and shackle replacements during operation. That is to say, it is possible to more efficiently use the crane with the developed scheduling method.

In this study, a block lifting scheduling problem in a static environment was considered as initial research on the scheduling of block lifting. However, in the actual scheduling of block lifting, urgent block lifting tasks often occur among the planned tasks. Thus, it is necessary to propose and develop an algorithm and a flexible system which can produce reliable results also in a dynamic environment. The actual erection sequence of the blocks (bottom, top, port, starboard, and/or center, as the case may be) should facilitate the gap and alignment control of the structures and equipment, which substantially affects the quality and production efficiency of the ship. Even though a lifting priority might be given explicitly by an expert manager to a specific block over another specific block, the actual lifting priority among other blocks might not be clear. In other words, there might be other invisible priority restrictions between blocks depending on the case (sequence). If the original block lifting sequence, which is given by an expert manager, is changed significantly, the new sequence based on the original set of priority conditions can be unrealistic. However, it is impractical to request an expert manager to create a complete set of lifting priorities for all cases beforehand. At present, the only way to determine whether the new sequence is realistic is therefore to request a detailed review by the expert manager. Thus, as future work, it is necessary to study how to effectively present the invisible lifting priority between each block. In addition, further research is needed regarding the modification and improvement of the proposed algorithm for application to more realistic block lifting scheduling problems in a shipyard. In other words, problems having more blocks should be analyzed. The results obtained from the proposed algorithm can be improved by developing new genetic operations for the GA and testing the refined algorithm.

## References

1. Daganzo CF (1989) The crane scheduling problem. Transp Res B 23(3):159–175
2. Daganzo CF (1990) Crane productivity and ship delay in ports. Transp Res Rec 1251:1–9
3. Peterkofsky RI, Daganzo CF (1990) A branch and bound solution method for the crane scheduling problem. Transp Res B 24(3):159–172
4. Kim KH, Kim KY (1999) An optimal routing algorithm for a transfer crane in port container terminals. Transp Sci 33(1):17–33
5. Ng WC, Mak KL (2005) An effective heuristic for scheduling a yard crane to handle jobs with different ready times. Eng Optim 37(8):867–877
6. Kim KH, Park KT (1998) A dynamic space allocation method for outbound containers in carrier-direct system. In: Proceedings of the third annual international conference on industrial

engineering theories, applications and practice, vol 2. Hong Kong, 28–31 December, pp 859–867

7. Kim HG, Kim CH (2007) A study on the optimal routing problem for a transfer crane. In: Proceedings of the annual spring meeting on Korean Operations Research and Management Science Society/Korean Institute of Industrial Engineers. Jinju, Korea, 25–26 May, pp 497–505
8. Cohon JL (1978) Multiobjective programming and planning. Academic Press Inc., New York
9. Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. Addison Wesley, Reading
10. Davis L (1991) Handbook of genetic algorithms. Van Nostrand-Reinhold, New York
11. Lee KY, Han SN, Roh MI (2003) An improved genetic algorithm for facility layout problems having inner structure walls and passages. Comput Oper Res 30(1):117–138
12. Grefenstette J (1986) Optimization of control parameters for genetic algorithms. IEEE Trans Syst Man Cybern 16(1):122–128