IE Interfaces
Vol. 24, No. 4, pp. 362-372, December 2011.

A Genetic Algorithm for the Container Pick-Up Problem

Shi W. Lee†

Department of Industrial Engineering, Pusan National University

컨테이너 픽업문제를 위한 유전자 알고리듬

이시우

부산대학교 산업공학과

Container pick-up scheduling problem is to minimize the total container handling time, which consists of the traveling distance and the setup time of yard cranes in a container yard. Yard cranes have to pick-up the containers which are stacked in the yard-bays to satisfy the work schedule requirement of quay crane, which loads and unloads containers on or from container ships. This paper allows the movement of multiple yard cranes among storage blocks. A mixed integer programming model has been formulated and a genetic algorithm (GA) has been proposed to solve problems of large sizes. Computational results show that the proposed GA is an effective method.

Keyword: container pick-up scheduling, genetic algorithm

1. Introduction

To transport international sea freight, a significant increase in the use of containers for general cargos has been observed over past decades. The efficiency of operations in container terminals plays an important role in providing fast and safe transportation service to customers over the world. The total cost for container management in sea freight industry was estimated to be in the region of $100 billion to $110 billion per year. Approximately 11% to 15% of this total cost, i.e., $14 billion to $16.5 billion annually, is associated with inefficiencies of container terminal operations (Behenna, 2001). In order to increase the container terminal throughput, terminal planners should use the equipment and personnel efficiently over existing terminal infrastructures.

Planning container terminal operations consists of berth scheduling, quay crane (QC) scheduling and load sequencing (Jung and Kim, 2006). Many researchers have studied QC scheduling and load sequencing to increase the container terminal throughput. This paper focuses on load sequencing of yard crane, which is one of the yard side equipment (YSE). The load sequencing problem is also known as pickup scheduling problem and generally consists of YSE dispatching process and container loading/unloading process. The pickup scheduling problem consists of two subproblems. The first one is to determine a sequence of yard-bays that each YSE visits. The other is to determine the number of containers to be picked up from each yard bay. The objective of pickup scheduling problem is to minimize the container handling time of the YSE. <Figure 1> and <Figure 2> show a typical example of container terminal and a typical layout of container terminal, respectively.

This work was supported by a 2-Year Research Grant of Pusan National University.

† Corresponding author : Shi W. Lee, Department of Industrial Engineering, Pusan National University, Busandaehak-ro 63, Geumjeon-gu, Busan, 609-735, Korea.
Fax : +82-51-512-7603, E-mail : slee7@pusan.ac.kr

Received 15 August 2011, Revised(1st: 21 October 2011, 2nd: 03 November 2011), Accepted 05 November 2011.

Figure 1. A typical example of a container terminal

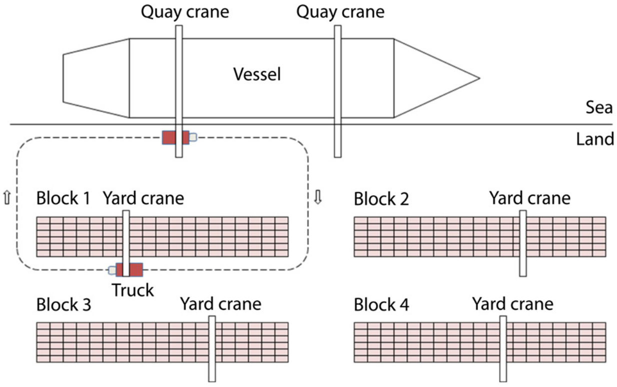


Figure 2. A typical layout of container terminal

When a container vessel arrives at a container terminal, multiple quay cranes serves it at the same time. For unloading operation, containers are discharged from the vessel and loaded onto trucks by quay cranes. The trucks unload containers at various yard-bays in the container yard. On the other hand, for loading operation, containers in the container yard are loaded onto trucks by container handling equipments, e.g. yard cranes. Then the trucks transports containers to loading areas next to quay cranes. Those containers are shipped onto a container vessel by quay cranes. <Figure 3> shows a typical flow of containers in a container terminal (Ng, 2005).

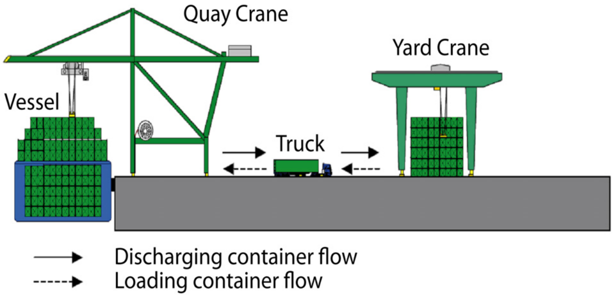


Figure 3. A typical flow of containers in container terminal (Ng, 2005)

Some container terminals might use straddle carriers to transport containers from a container yard to a quay crane instead of using trucks. Some terminals have buffer spaces to temporarily store containers for a certain period before transferring containers to the container yard or quay cranes. Therefore, operations in container terminals might be different due to their specific characteristics and unique environments.

This paper focuses on the operation of multiple yard cranes during loading operation in a container yard. Yard crane was derived from industrial overhead cranes. There are two different types of yard cranes which are rubber-tired gantry crane (RTGC) and rail-mounted gantry crane (RMGC). The first one is frequently referred to transtainer and runs on heavy-duty pneumatic-tired wheels and the latter one runs on steel wheels over fixed rails. Although these two types of yard cranes serve the same purpose, there are differences in their functions between them in addition to their wheels.

RTGC spans several rows of containers and one truck lane, and can stack containers up to six levels. RTGC is an essential container yard stacking device and is used in combination with other container handling equipment for the wharf transfer operation. RTGC is able to maximize the utilization of container stacking area better than the fork lift truck and straddle carrier (Branch, 1986).

The containers in container yards are normally arranged in long rows parallel to the wharf, with about 30 TEU slots per row, which is thus approximately 180 m long. There are usually five or six rows of containers per block, and the truck lane. At the ends of the blocks, a roadway space is provided between adjacent blocks. The wheels of the RTGC can turn throufh 90 degrees so that it can move from one storage block to another as required to meet operational needs while RGMC can only move along the fixed rails. <Figure 4> show an example of the RTGC operating over the block.

Figure 4. RTGC's operation over containers

Table 1. An example of a work schedule

Sequence(subtour index)	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Container type	A	D	E	D	G	A	B	C	B	H	H	G	F	E
Quantity	32	57	27	3	8	5	39	22	47	81	22	5	77	6

At storage blocks in the container yard, yard cranes operate for all containers handling activities, including lifting containers for trucks, storing them at storage locations, picking up containers from storage locations and putting them on the trucks. They play a very important role in the container yard and are usually the bottleneck in the container handling process (Zhang, Wan, Liu, and Linn, 2002). Therefore, in order to achieve a good throughput and a high efficiency, the deployment of yard cranes should be planned carefully. Problems of scheduling vehicles and material handling equipment have been arisen frequently in logistic systems and have also been extensively studied under various different settings (Bramel and Simchi-Levi, 1997 as cited in Ng, 2005). However, the results reported in a vast majority of research literatures were not directly applicable to a container terminal due to its unique characteristics (Ng, 2005).

A particular case of pick-up scheduling problem for a single yard crane was studied by Kim and Kim (2002). They studied the operation of a single yard crane in a single block. Their objective was to minimize the container handling time, which consists of the total traveling distance of the yard crane and the setup time of the yard crane, by determining the sequence of yard-bays to be visited and the number of containers to be picked-up at each yard-bay. Kozan and Preston (1999) studied the scheduling of transfer operation of containers without considering actual sequencing issues. The research of a single yard crane operation was also studied by Kim and Kim (1999a), (1999c) and Narasimhan and Palekar (2002).

The operation of multiple yard cranes is also considered as one of pick-up scheduling problems. This operation was studied by Jung and Kim (2006). They proposed a method to schedule loading operations for multiple yard cranes which operate over a storage block. In accordance with multiple yard cranes operate in a storage block, the interference between adjacent yard cranes is significantly important.

Recently, Lee *et al.* (2007) studied a scheduling problem of multiple yard cranes for loading containers over storage blocks. However, they limited the accessibility of each yard crane by assuming that the yard crane was not allowed to travel between two blocks during the loading process.

Linn and Zhang (2003) studied yard cranes deployment problem in each time period. In their research, multiple yard cranes were assigned to work over multiple storage blocks. However, they did not consider either the scheduling of detailed movements for each yard crane or particular tasks for each yard crane during the loading operation. The problem of scheduling multiple yard cranes in a container yard to minimize the total jobs' waiting time was also studied by Ng (2005). The author addressed the scheduling problem of yard cranes by considering the yard cranes interference among adjacent blocks.

2. Problem Description

In a container terminal, a container yard is divided into rectangular regions called blocks. Each block typically consists of 20~30 yard-bays, which store a number of containers. A container type is defined as a collection of containers which have the same size and the same delivered destination. It is assumed that each yard-bay can store only one type of containers. Examples of a work schedule and a yard map are given as shown in <Table 1> and <Figure 5>, respectively.

The work schedule shows the number of containers of each type to be picked up sequentially for quay cranes. The yard map shows the distribution of containers of each container type in the container yard. Yard bays are numbered sequentially over the blocks as shown in <Figure 5>. A "sub-tour" denotes a sequence of containers of each container type to be picked-up during loading operations.

The objective of pick-up scheduling problem is to minimize the total container handling time, which consists of the traveling distance between visiting yard-bays, and the setup time of yard cranes on the yard-bay. The traveling distance of a yard crane is dependent on the visiting sequence of yard-bays. The setup time of yard crane occurs whenever the yard crane moves from one yard-bay to another. Even though a yard crane is usually assigned to a block, it

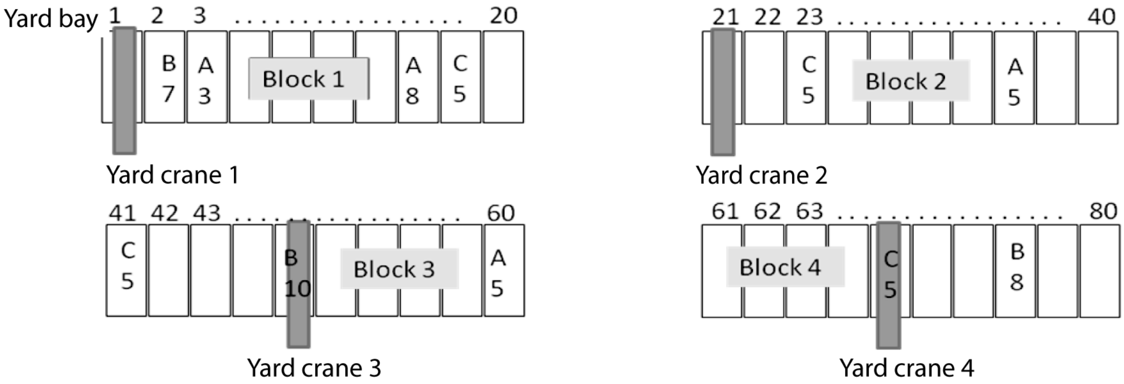


Figure 5. An example of a yard map

might be able to move to operate in the other blocks. In this paper, it is assumed that multiple yard cranes are operating in a container yard during the loading operation. It is also assumed that there are a sufficient amount of trucks for delivering containers from container yard to loading zones, which means the waiting of the trucks can be ignored.

When loading a container from the container yard to a ship, a yard crane moves back and forth along the specific block to the intended container which is stacked in a yard bay, then its hoist picks up the container and loads it onto a truck. The container has to be delivered to quay crane in the order specified in the work schedule. Therefore, all yard cranes must operate in the order prescribed by the work schedule of quay cranes. Afterwards, the truck transports the container to a corresponding quay crane, which picks up the container and loads it onto a ship. The efficiency of loading operations significantly depends on the loading schedule of the exported containers to be loaded on the ship. Thus, the efficient container handling at container terminal is important in reducing container transportation costs and keeping shipping schedules, since it has to satisfy the work schedule required by quay crane.

3. Mathematical Model

A mathematical programming formulation for operation of a single yard crane, originally proposed by Kim and Kim (2002), has been extended to a mathematical model representing the operation of multiple yard cranes over storage blocks. In addition, the set operators have been removed from their formulation. This section provides a mixed integer programming formulation for pick-up scheduling problem of operating multiple yard cranes over storage blocks.

There are several assumptions to simplify the problems under consideration. There is only one type of containers stacked in a yard bay. Each storage block has a yard crane at the first yard bay initially. In other word, the number of yard crane is equal to the number of block. When a yard crane moves to a yard-bay within or outside of the current storage block, the shortest path will be chosen.

When a yard crane moves to another storage block, penalties will be assigned. <Figure 6> illustrates two possible yard crane movements which are defined "over-column movement" and "over-row movement."

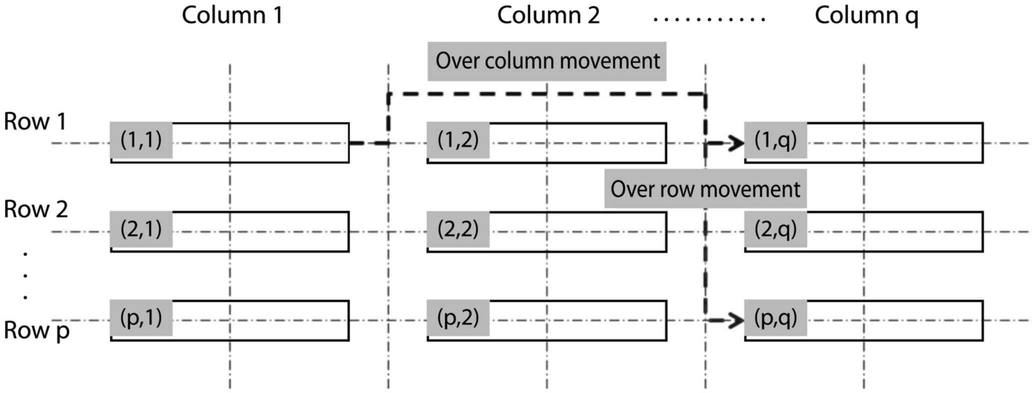


Figure 6. Yard crane movements

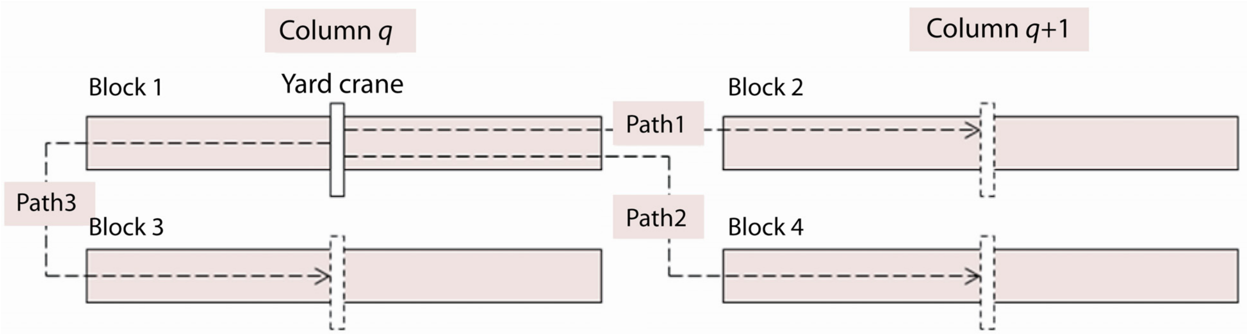


Figure 7. Potential movements of yard crane between blocks in the adjacent columns

To calculate the traveling distance from yard-bay i to yard-bay j, blocks to which yard-bays i and j belong, should be determined first to add the penalties of movements between blocks. A simple penalty rule has been used by adding different penalties subject to the types of movements and the number of rows or columns being passed.

<Figure 7> shows the potential movements of a yard crane between two blocks in the adjacent columns. If two blocks are adjacent to each other and align longitudinally, for example block 1 and block 2, a yard crane can move from one block to the other one easily without any turning movement (see Path1 in <Figure 7>). A small penalty is assessed since moving a yard crane between two adjacent blocks still causes a delay. In the other case, if a yard crane moves to another blocks on a different row, for example, between block 1 and block 4, bigger penalties are applied to this movement. The yard crane must first move out of the current block, make a 90º turn and move to the other row, then make a 90º turn again and enter the other block (see Paths 2 and 3 in <Figure 7>). This movement occupies a large amount of traffic space for an extended time period, which can obstructs the traffic on the roads between blocks and also delays other terminal operations.

Finally, the notations used in the MIP formulation are given as follows :

i, j The indices for yard bays
v The index for yard cranes
m The number of sub-tour that constitute a complete tour of a yard crane
Ω The set of all yard-bays
ϕ The set of empty yard-bays (the yard-bay that does not have any container)
l The number of container types
V The number of yard cranes
$S(h)$ The set of sub-tour numbers in which the containers of type h exist
$B_v(h)$ The set of yard-bay numbers in which the containers of type h exist, and are attended by yard crane v
c_{hj} The initial number of container of type h stack at yard-bay j
r_t^h The number of containers of type h to pick-up during sub-tour t
g_t Container type number of sub-tour t
d_{ij} Travel distance between yard-bay i and j
e_{ij} 0, if $i = j$,
1, otherwise,
t Sub-tour number, $t = 0, 1, \cdots, m, m+1$, where $t = 0$ and $m+1$, respectively, correspond to the initial and the final locations of the yard crane
S_v The initial location of the yard crane v
F_v The final location of the yard crane v
T_s The setup time of yard crane for each visit to a yard bay
T_d The traveling distance of yard crane per the distance of a yard bay length

Decision variables :
y_{vij}^t 1, if yard crane v moves from yard-bay i to yard-bay j after completing sub-tour t
0, otherwise
z_{vij}^t 1, if yard crane v moves from yard bay i to yard bay j during sub-tour t
0, otherwise
x_{vj}^{ht} The number of container of group h picked up at yard bay j during sub-tour t by yard crane v

A mixed integer programming model for multiple yard cranes over the multiple storage blocks has been presented as follows :

Minimize

$$\sum_{t=0}^{m}\sum_{v=1}^{V}\sum_{i,\,j\in\Omega}(T_s e_{ij}+T_d\,d_{ij})y_{vit}^{t}+$$
$$\sum_{t=1}^{m}\sum_{v=1}^{V}\sum_{i,\,j\in\Omega}(T_s+T_d\,d_{ij})z_{vij}^{t} \quad (1)$$

Subject to

$$\sum_{v}^{V}\sum_{j\in B_v(g_1)} y_{vS_vj}^{0}=1, \quad (2)$$

$$\sum_{v}^{V}\sum_{j\in B_v(g_m)} y_{vjF_v}^{m}=1, \quad (3)$$

$$\left(\sum_{v}^{V}\sum_{j\in\Omega} y_{vji}^{t-1}+\sum_{v}^{V}\sum_{j\in\Omega} z_{vki}^{t}\right)-$$
$$\left(\sum_{v}^{V}\sum_{j\in\Omega} y_{vij}^{t}+\sum_{v}^{V}\sum_{k\in\Omega} z_{vik}^{t}\right)\le 1,\ i\in\Omega$$
$$t=1,2,\cdots,m, \quad (4)$$

$$\sum_{i\in\Omega}\sum_{j\in\phi}\sum_{v}^{V} y_{vij}^{t}=0, \qquad t=0,1,\cdots,m \quad (5)$$

$$\sum_{i\in\Omega}\sum_{j\in\phi}\sum_{v}^{V} z_{vij}^{t}=0, \qquad t=1,2,\cdots,m \quad (6)$$

$$x_{vj}^{ht}\le M\left(\sum_{v}^{V}\sum_{k\in\Omega} z_{vkj}^{t}+\sum_{v}^{V}\sum_{i\in\Omega} y_{vij}^{t-1}\right), \quad j\in\Omega,$$
$$h=1,2,\cdots,l,\ v=1,2,\cdots,V,\ t=1,2,\cdots,m, \quad (7)$$

$$\sum_{j\in\Omega}\sum_{v}^{V} x_{vj}^{ht}=r_t^h,\ h=1,2,\cdots,l,\ t=1,2,\cdots,m \quad (8)$$

$$\sum_{t=1}^{m}\sum_{v}^{V} x_{vj}^{ht}=c_{ht}, \qquad h=1,2,\cdots,l,\ j\in\Omega \quad (9)$$

$$y_{vij}^{t}\in\{0,1\}$$
$$i,j\in\Omega,\ v=1,2,\cdots,V,\ t=1,2,\cdots,m, \quad (10)$$

$$z_{vij}^{t}\in\{0,1\}$$
$$i,j\in\Omega,\ v=1,2,\cdots,V,\ t=1,2,\cdots,m, \quad (11)$$

$$x_{vj}^{ht}\ge 0,$$
$$j\in\Omega,\ i,j\in\Omega,\ v=1,2,\cdots,V,\ t=1,2,\cdots,m, \quad (12)$$

The objective function (1) is to minimize the total container handling time which is sum of the traveling distances of all yard cranes and the setup time which occurs whenever a yard crane moves between yard-bays. Constraints (2) and (3) represent the flow conservation of yard cranes at initial and final locations. Constraint (4) represents the flow conservation at the other locations. Constraint (5) and (6) ensure that a yard crane will not move to yard-bays that do not have any container. Constraint (7) implies that containers at a yard bay can be picked up only by a yard crane at that yard bay. Constraint (8) ensures that the number of containers picked up in a sub tour must be equal to the number of containers requested by the work schedule. Constraint (9) ensures that the number of containers picked up during the whole tour must be equal to the initial number of containers at each yard bay for every container type. Constraints (10) and (11) ensure y_{vij}^{t} and z_{vij}^{t} to be binary variables. Constraint (12) ensures non-negativity.

4. A genetic Algorithm

The pickup scheduling problem is difficult to be solved optimally in a reasonable amount of time even though the problems are small. The GA has been applied successfully in many combinatorial optimization problems. However, it does not guarantee the optimality due to its stochastic nature, but it can rapidly locate good or near-optimal solutions in significantly less time. The proposed GA was developed to efficiently solve the problems of medium and the large sizes. The pickup scheduling problem has work schedule and yard map as input data. A typical work schedule is given in <Table 2> and shows the sequence of the number of containers of each type to be picked-up. The yard map is a layout of the containers distribution in the container yard. The yard map can be given as <Table 3>. It shows the number of containers of each container type which are stored in each yard-bay. For example, the work schedule in <Table 2> includes a loading operation for picking up three container types, which are type A, B, and C. In this example, sub-tour 1 requires three containers of type A; sub-tour 2 requires six containers of type B; and so on. <Table 3> shows the number of containers of each type, in which are stacked at each yard-bay, for instance, yard-bay 1 stores five containers of type A; yard-bay 3 stores two containers of type C; and so on.

Table 2. An example of a work schedule

Subtour Number	1	2	3	4	5
Container type	A	B	C	B	A
Quantity	3	6	5	4	7

Table 3. An example of the initial number of containers in each yard-bay

Container type	Yard bay index							
	1	3	4	7	8	10	12	13
A	5				3			2
B			4	2			4	
C		2				3		

The gene representation in the proposed genetic algorithm is a one-dimensional array, which is divided to as many portions as the number of container types. Each portion contains two different sectors as shown in <Figure 8>. The first sector, sub-tour sector, represents random sub-tour indices for loading containers of the corresponding container type. The second sector, random yard-bay sector, contains randomly generated sequences of visiting yard-bay indices, which have containers of the corresponding container type. <Figure 9> shows an example of the chromosome which consists of container type A, B, and C.

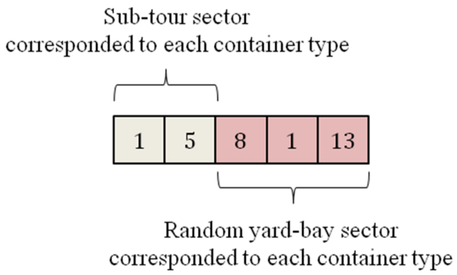

Sub-tour sector
corresponded to each container type
1
5
8
1
13
Random yard-bay sector
corresponded to each container type

Figure 8. The example of a genetic representation for a single container type

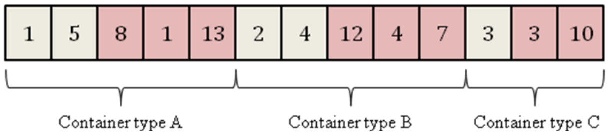
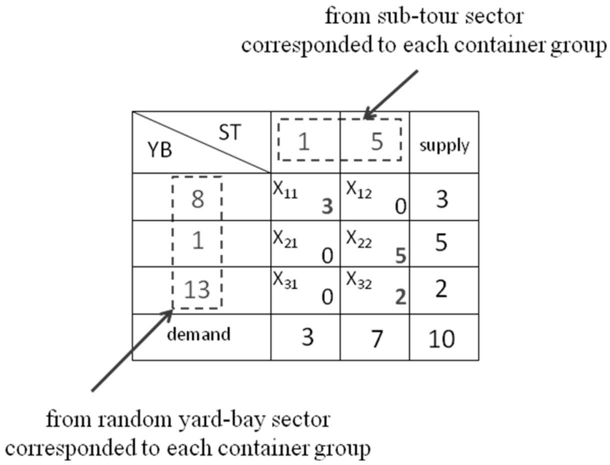


Figure 9. The example of a chromosome of container type A, B, and C

In <Figure 9>, the chromosome includes 3 types of containers which are container type A, B, and C. According to the work schedule in <Table 2>, there are five sub-tours, which are sub-tours 1 to 5 in the work schedule. Each sub-tour has the number of containers of a specific container type to be picked up. Containers of type A is assigned to sub-tour 1 and 5; containers of type B is assigned to sub-tour 2 and 4; and containers of type C is assigned to sub-tour 3. Note that the indices of sub-tour in each container type in the chromosome are not necessary to be in a numerical order. It is assumed that the containers are randomly distributed in the container yard. Each yard-bay can store only one type of container. In the example, containers of type A are stored in yard-bay 1, 8, and 13; containers of type B are stored in yard-bay 4, 7, and 12; and containers of type C are stored in yard-bay 3, and 10. However, in the chromosome, the randomly generated visiting sequences of yard-bays for container type A, B, and C are {8, 1, 13}, {12, 4, 7}, and {3, 10} respectively.

Once the chromosome is given, the number of containers to be picked-up at each yard-bay can be determined by the method in transportation problem called Northwest corner rule (Hillier and Lieberman, 2005).


from sub-tour sector
corresponded to each container group
from random yard-bay sector
corresponded to each container group

YB \ ST	1	5	supply
8	X_{11} 3	X_{12} 0	3
1	X_{21} 0	X_{22} 5	5
13	X_{31} 0	X_{32} 2	2
demand	3	7	10

Figure 10. The example of the Northwest Corner Rule

<Figure 10> demonstrates the example of Northwest Corner Rule table for obtaining the number of containers type A to be picked-up at each yard-bay for a specific sub-tour. The strings {1, 5}, and {8, 1, 13} are brought from the sub-tour sector and the yard-bay sector in the chromosome shown in <Figure 3>. From the given work schedule in <Table 2>, sub-tour 1 and 5 require three and seven containers of type A, respectively. For container type A supplies, <Table 3> illustrated that yard-bay 8, 1, 13 store three, five, and two containers respectively.

The procedure to obtain the number of containers to be picked-up from each yard-bay during a specific sub-tour is explicitly illustrated as follow. First, start in the upper left hand of the tableau (X_{11}) and assign the number to the variable by considering demand and supply in the corresponding sub-tour column and yard-bay row respectively. Then, move one column to the right, if the demand of that sub-tour is satisfied, and the current yard-bay still has any supply remaining. Otherwise, move one row down to the next yard-bay. In the example, the demand of sub-tour 1 is three containers which equals to the supply at yard-bay 8, so $X_{11} = 3$. Thereafter, move one column to the right which is for sub-tour 5. Since there is no supply remaining in yard-bay 8, then move one row down to yard-bay 1. Allocate five

Table 4. An example of the final pick-up schedule for yard cranes

Sequence (subtour index)	1	2		3		4		5	
Container type	A	B		C		B		A	
pick-up yard-bay	8	12	4	3	10	4	7	1	13
pick-up quantity	3	4	2	2	3	2	2	5	2

Table 5. Characteristics of each various problem size

Characteristics	Problem size		
	Small	Medium	Large
Number of container types	3~4	6~8	8~10
Number of blocks	4~6	10~12	12~14
Number of yard bays per block	5~10	15~17	20~25
Number of subtours	5~10	12~16	18~22

containers to variable X_{22} which equals to the supply at yard-bay 1. Afterward, move one row down to yard-bay 13 since there are two more containers required in this sub-tour. Finally, the last variable is equal to the number of containers remain in that yard-bay, $X_{32} = 2$. Then, apply the Northwest Corner Rule procedure to every container type to obtain the number of containers to be picked-up at each yard- bay corresponding to the specific sub-tour. Afterward, the final pick-up schedule for yard cranes is constructed as illustrated in <Table 4>.

After obtaining the number of containers to be picked up at each yard-bay, the containers are picked up by the order specified in the work schedule. One of yard cranes in the container yard which locates nearest to the yard-bay, where stored the specific container type required by the work schedule, is assigned to do the task. Then, the pick-up location of the yard crane becomes the new initial location of the next task. Continue the procedure until the last sub-tour in the work schedule is achieved.

The fitness function of each feasible solution in the population is evaluated as an inverse of sum of the total handling time performed by all yard cranes. The roulette wheel selection method is applied in the proposed GA. Afterward, two parents are chosen for next genetic operations.

A single cut crossover has been used to each sector in a chromosome. Mutation is an which helps the algorithm avoid a local convergence by preventing the population of chromosomes from becoming too similar to each other. Once the chromosome is selected, the reciprocal exchange mutation operator has been applied to each sector in a chromosome.

The proposed GA stops when a specified maximum number of the generation is reached.

5. Computational Results

Various problems which include small, medium, and large sized problem have also been randomly generated under the specific criteria. Numerical experiments have been conducted to demonstrate the performance of the proposed GA. The proposed GA has been implemented by Microsoft Visual Basic 2005 using Microsoft Visual Studio.NET Framework Version 2.0.

Initial experiments demonstrated the robust and best results when the following GA parameters are used; crossover rate 0.8 and mutation rate 0.05. Hence, these GA parameters are used for remaining experiments.

In the experiment, three different sizes of problems, which are small, medium, and large size problems, have been generated under the specific criteria. The considered criteria in this research are the number of container types, the number of blocks, the number of yard-bay per block, and the number of sub-tour. A "problem generator" has been developed to generate the problems. For each problem size, five different examples have been randomly generated based on the criteria illustrated in <Table 5>.

These parameter ranges has been used to generate the test problems of various sizes. The corresponding parameters for each problem are summarized in <Table 6>. Note that the percentage of allocation spaces is defined as the ratio of yard-bays storing containers over the total yard-bays and it has been kept at least 60%. The set of yard-bays, the numbers of containers in yard bays, and work schedules for quay cranes were also randomly generated based on these characteristics. <Table 7> shows the number of yard-bays which store the containers of each type. For instance, in problem S1, containers of type A are distributed in five yard bays, containers of type B are distributed in four yard bays, and so on. Afterward, computational experiments have been conducted to demonstrate the effectiveness and efficiency of the proposed GA.

The details of randomly generated work schedules and yard maps of each problem are omitted o for simplicity. The experiment was conducted ten times for each problem to demonstrate the effectiveness of the proposed GA. In the experiment, the proposed GA was performed in the following conditions. The population size was 50. The crossover rate and the mutation rate were 0.8 and 0.05, respectively. Maximum number of generations was set to be 5000. <Table 5>, <Table 8> shows the computational results of the proposed GA including the minimum objective value, the computational time (time in seconds), and the terminated generation.

Table 6. Parameter values of each problem

Problem	Number of container types	Number of subtours	Number of containers	Number of blocks	Number of yard-bay per block	Percentage of allocation spaces
Small1	3	5	36	4	5	65.00
Small2	3	6	57	4	6	91.67
Small3	4	10	192	6	10	93.33
Medium1	6	12	315	8	15	68.33
Medium2	6	15	416	10	16	80.63
Medium3	8	16	475	10	17	84.71
Large1	8	18	650	12	20	98.33
Large2	9	20	734	12	20	87.50
Large3	10	22	930	14	25	79.43

Table 7. The number of yard-bays which store each container type

Problem	Container type										Total
	A	B	C	D	E	F	G	H	I	J	
Small1	5	4	4								13
Small2	10	10	2								22
Small3	16	3	18	19							56
Medium1	12	20	8	7	24	11					82
Medium2	22	32	23	29	10	13					129
Medium3	12	23	8	16	13	37	8	27			144
Large1	42	21	47	35	35	4	25	27			236
Large2	13	50	21	26	25	19	14	15	27		210
Large3	44	46	29	39	21	35	18	16	8	22	278

Table 8. Computational results of the proposed GA (time in seconds)

	Run 1		Run 2		Run 3		Run 4		Run 5	
	Result	TG*	Result	TG	Result	TG	Result	TG	Result	TG
Small1	97	29(3)	97	55(3)	97	24(3)	97	260(5)	97	30(3)
Small2	154	443(10)	152	135(8)	154	206(6)	153	393(9)	154	384(8)
Small3	530	344(43)	539	189(34)	529	333(45)	536	52(29)	528	535(45)
Medium1	929	816(83)	925	150(39)	923	617(42)	929	237(76)	929	653(79)
Medium2	1341	683(179)	1360	688(182)	1348	727(205)	1381	545(157)	1357	240(138)
Medium3	1562	695(203)	1575	157(134)	1555	842(212)	1557	483(203)	1564	543(168)
Large1	2560	261(231)	2575	64(148)	2566	116(185)	2587	148(266)	2559	86(283)
Large2	2550	291(191)	2568	325(378)	2565	285(359)	2552	157(236)	2550	291(191)
Large3	3639	92(493)	3668	332(712)	3641	186(258)	3640	199(531)	3668	332(712)

*TG stands for the terminated generation.

Table 9. Results of the experiments of the proposed GA

Problem	Best Solution Value	Worst Solution Value	Average Solution Value	Variance
Small1	97.0	97.0	97.0	0.000
Small2	152.0	154.0	153.1	0.994
Small3	528.0	539.0	533.4	3.534
Medium1	923.0	934.0	927.6	3.307
Medium2	1341.0	1381.0	1363.5	15.020
Medium3	1555.0	1575.0	1563.7	6.516
Large1	2559.0	2587.0	2573.8	10.902
Large2	2550.0	2568.0	2559.2	7.131
Large3	3639.0	3668.0	3653.7	10.698

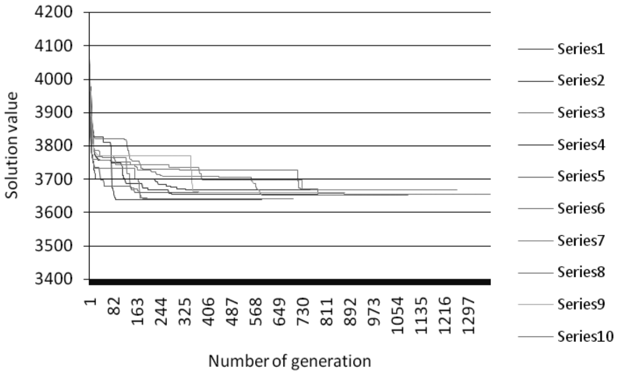


Figure 11. An example of the variation of the solution of a large sized problem (L5)

<Table 8> shows five runs for test problems using the proposed GA and that the proposed GA can achieve the best solution for the small sized problem in a short period of time. <Table 9> summarizes the best solution values, the worst solution values, and the average solution values of 10 runs for all test problems. It shows that the variances of the solution values for the small size problems are low. However, as the problem size increases, the variance of the solution values are also increases consecutively.

The variation of the solutions of the largest problem (Large3) by the proposed GA is shown in <Figure 11>. The graph shows the convergence of the solutions over the generation. The proposed GA is terminated since the solution is no longer improved in 500 generations.

6. Conclusions

The MIP model of operating a single yard crane over a block has been extended to the MIP model of operating multiple yard cranes over storage blocks. A genetic algorithm for solving the pick-up scheduling problem has been proposed. The computational results from the experiments show that the proposed GA is an effective method for solving the pickup scheduling problem of operating multiple yard cranes over storage blocks during a loading operation in a container yard.

References

Behenna, P. (2001), Container Asset Visibility : The Key to Reducing Ocean Carrier Operating Costs, *International Asset Systems*, 1-9.

Jung, S. H. and Kim, K. H. (2006), Load scheduling for multiple quay cranes in port container terminals,*Journal of Intelligent Manufacturing*, 17, 479-492.

Kim, K. Y. and Kim, K. H. (1999a), A routing algorithm for a single straddle carrier to load export containers onto a containership, *International Journal of Production Economics*, 59, 425-433.

Kim, K. H. and Kim, K. Y. (1999b), Routing straddle carriers for the loading operation of containers using a beam search algorithm, *Computers and Industrial Engineering*, 36 , 109-136.

Kim, K. H. and Kim, K. Y. (1999c), An optimal routing algorithm for a transfer crane in port container terminals, *Transportation Science*, 33(1), 17-33.

Kim, K. Y. and Kim, K. H. (2002), Heuristic algorithms for routing yard-side equipment for minimizing loading times in container terminals, *Naval Research Logistics*, 50, 498-514.

Kozan, E. and Preston, P. (1999), Genetic algorithms to schedule container transfers at multimodal terminals, *International Transactions in Operational Research*, 6, 331-329.

Lee, D. H., Zhi, C., and Meng, Q. (2007), Scheduling of two-transtainer systems for loading outbound containers in port container terminals with simulated annealing algorithm, *International Journal of Production Economics*, 107, 115-124.

Linn, J. R. and Zhang, C. Q. (2003), A heuristic for dynamic yard crane deployment in a container terminal, *IIE Transactions*, 35, 161-174.

Narasimhan, A. and Palekar, U. S. (2002), Analysis and algorithms for the transtainer routing problem in container port operations, *Transportation Science*, 36(1), 63-78.

Ng, W. C. (2005), Crane scheduling in container yards with inter-crane interference, *European Journal of Operational Research*, 164, 64-78.

Zhang, C., Wan, Y. W., Liu, J., and Linn, R. J. (2002), Dynamic crane deployment in container storage yards, *Transportation Research*, Part B, 36, 537-555.

Shi W. Lee

Seoul National University Industrial Engineering (BS)
Seoul National University Industrial Engineering (MS)
The Pennsylvania State University Industrial Engineering (Ph.D.)

Position : Department of Industrial Engineering, Pusan National University

Interests : Operations Research, Combinatorial Optimization, Process Modeling and Optimization