Available online at www.rajournals.in
RA JOURNAL OF APPLIED RESEARCH
ISSN: 2394-6709
DOI:10.31142/rajar/v4i6.05
Volume: 04 Issue: 06 June -2018

International Open Access
ICV- 74.25
Impact Factor: 5.985
Page no.- 1739-1745

Optimal Job Scheduling of Multiple Rail Cranes in Rail Stations

Nguyen Vu Anh Duy[1], **Nguyen Le Thai**[2]

[1,2]Faculty of Mechanical Engineering Technology, Ho Chi Minh City University of Food Industry, Ho Chi Minh City, Vietnam

ARTICLE INFO

Published Online: 07 June 2018

Corresponding Author: **Nguyen Vu Anh Duy**

ABSTRACT

This paper considers the scheduling problem of multiple rail cranes to load and unload inbound and outbound containers to and from wagons of trains within rail stations. We not only assign working areas to cranes, but also determine the job sequence of each crane. We minimize the maximum completion time (makespan) of all rail cranes. Dual-cycle operations of cranes are applied and the re-handling work of containers is also considered. A branch-and-bound algorithm is developed to find an optimal solution. A simulated annealing algorithm is designed to obtain near optimal solutions of large-sized problems. Numerical examples are studied to investigate the performance of these algorithms.

KEYWORDS: Branch and bound; simulated annealing; rail crane; scheduling

I. INTRODUCTION

The volumes of goods being transported internationally has increased rapidly in the last few decades. Rail transportation is an important mode of global transportation. Although it is limited to rail hubs, it is generally considered to be a relatively safe and environmentally friendly transportation mode, which can efficiently move massive quantities of containers over long distances. The rail station is critical to the efficiency of rail transportation. Figure 1 shows a general rail stations layout. In rail stations, outbound containers are unloaded and inbound containers are loaded using rail cranes, forklifts and reach stackers, and the waiting time of trains in rail stations is directly dependent on the operation of these systems.

This paper deals with the job sequencing problem of rail cranes in rail stations located at seaport container terminals. We not only assign the working areas but also determine the job sequences, minimizing the maximum completion time of all rail cranes. Dual-cycle operations are applied. If the wagon onto which an inbound container will be loaded is not empty, the inbound container is moved to temporary storage area. The inbound container is loaded onto the wagon after the wagon is empty. This is referred to as a "re-handling" case.

The latest review of container processing in rail stations was presented by Boysen et al. (2013) with the assignment of container movements by cranes and determining the sequence of container movements. Alicke (2002) considered the logistical problems relating to the loading and unloading of trains in a theoretical intermodal terminal called Mega Hub, where containers are transferred between trains and shuttle cars by cranes. That article not only dealt with the assignment problem of containers to cranes, but also addressed sequencing problems relating to containers. That author formulated an optimization model based on Constraint Satisfaction, and developed various heuristics to minimize the maximum lateness of all trains. Boysen and Fliedner (2010) discussed how to determine the static crane areas in rail – road transshipment yards. In contrast, Boysen et al. (2010) addressed the static crane areas in rail – rail transshipment yards. Both of these studies used dynamic programming to find the optimal working area for each rail crane. However, they did not determine the optimal job sequencing of rail cranes. None of these researches assumed the dual – cycle operation of rail cranes. Jeong and Kim (2011) applied the dual – cycle operation of a crane in order to determine the job sequence of the rail crane and the parking positions of container trucks. Nevertheless, they considered the operation of a rail crane which just moves in one direction along the train. Pap et al. (2012) developed a branch-and-bound algorithm to find the optimal job sequence of a rail crane under a dual-cycle operation, although they did not consider re-handling cases for containers. Guo et al. (2013) dealt with the scheduling problem of loading and unloading containers between a train and container yards. Multiple gantry cranes were scheduled in a rail station with constraints of safety distance, as well as a non-crossing requirement being placed on the cranes. However, that article assumed the one dimension travel (gantry travel) of the cranes, and did not consider dual-cycle operation or the re-handling containers. Nguyen and Yun (2014) considered the job scheduling of a rail crane in rail terminals. The paper studied the transfers of containers between trains and trucks by a rail crane. In addition, re-handling and the dual-cycle operation are considered.

The remainder of this paper is organized as follows: The scheduling problem of multiple rail cranes in a rail station is presented in detail in Section 2. Branch and bound simulated annealing algorithms are proposed in Sections 3 and 4, respectively. Numerical examples are studied in Section 5. Section 6 concludes the study.

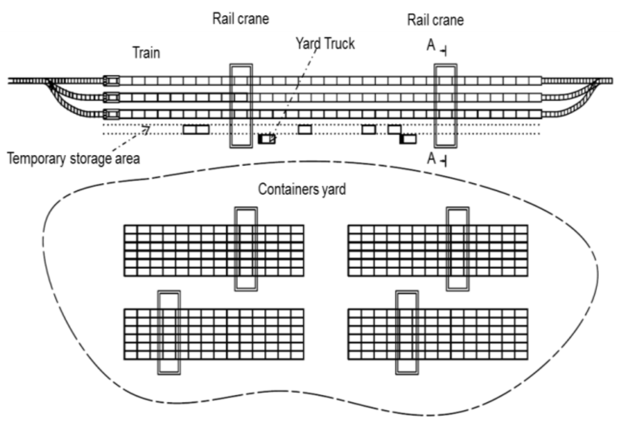

Figure 1. Bird's eyes view of a rail station.

II. THE SCHEDULING PROBLEM OF MULTIPLE RAIL CRANES IN A RAIL STATION

In this section, the scheduling problem of multiple rail cranes in a rail station is represented in detail. We focus on the rail station located in seaport container terminals, as is shown in Figure 1. The locations, where trains park, are called "rail tracks," and places where trucks park for loading and unloading containers (inbound container or outbound containers) are called "transfer points". The "transfer track" is a row of transfer points, which would most likely be in the form of a road of trucks.

Supposing that there are T trains on the rail tracks, and they are being handled by K rail cranes. Figure 2 shows the notation of the multiple rail cranes scheduling problem. In this problem, we not only consider the makespan of all rail cranes, $C_{\max}$, but also determine the working area of each rail crane, which is denoted by the right-hand side boundary of the area, b_k for $k = 1,...,K$. Travelling time of a crane between two adjacent wagons is t_x and the travelling time of a crane between two adjacent trains is t_y.We consider 16 operation cases of cranes, which are detailed in Table 1. In table 1, i and j denote the container indices. I and O are the sets of inbound containers and outbound containers. $x(\ )$ and $y(\ )$ indicate the position of a container in a two-dimensional area. The travelling time of a crane between to position (x, y) and (x', y') is shown by $d_{x',y'}^{x,y} = Max\left(t_x\left|x - x'\right|, t_y\left|y - y'\right|\right)$. $R_i = 1$ means that the container i needs to be re-handled. Otherwise, it will have a value of "0". Furthermore, we assume that

i. Trains are available for processing simultaneously (at time zero).
ii. A truck carries only one container at a time.
iii. Temporary storage space is unlimited.
iv. The loading position of each inbound container is given.
v. Travelling times of cranes are constant.

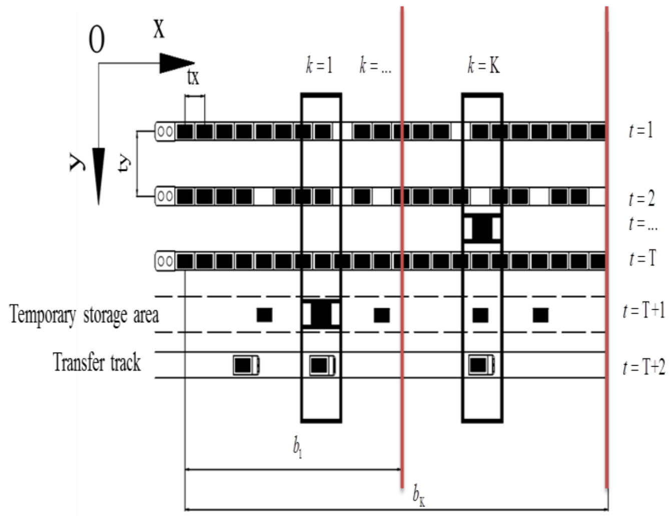

Figure 2. Notation of the scheduling problem of multiple rail cranes.

III. BRANCH-AND-BOUND ALGORITHM

In this section, a branch-and-bound algorithm (B&B) is developed to find the optimal solutions of the multiple rail cranes scheduling problem. The solutions of B&B are the sequence of containers assigned to all cranes. The sequences are constructed by assigning each available container to the crane which has the earliest completion time.

The following notation is used to describe the solution process.

Q The set of containers assigned to all cranes

P^k The set of containers which were assigned to crane k

R^k The set of containers which have not been assigned to the sequence of crane k , but which are located in the working area of crane k

$$R^k = \left\{i \in P^k : l_k^l \le x(j) \text{ and } x(i) \le l_k^r\right.$$

S_k The sequence of containers assigned to crane k

c_k Completion time of the current node in the search tree by crane k .

l_k^c The last position along the x direction of crane k after performing the container sequence S_k

l_k^l The left boundary of crane k after performing the sequence S_k; $l_1^l = 1$.

l_k^r The right boundary of crane k after performing the sequence S_k; $l_K^r = W$.

The process of the branch-and-bound algorithm (B&B) is as follows:

Step 1: The root node of the search tree is determined as follows:

$Q = \varnothing$, all job sequences of cranes are empty, $S_k = (\ )$, $c_k = 0$ for $k = 1,...,K$.

The heuristic, “Travel to the Right,” is used to generate the initial upper bound, and to set it as a current upper bound (CUB).

Step 2: The node which has the smallest lower bound value is selected from the search tree.

Step 3: The crane with smallest c_k is selected, and the P^k is generated.

$$P^k = \begin{cases} \{ i \in I \cup O \setminus Q : x(i) < l_{k+1}^l \} \text{ for } k = 1 \\ \{ i \in I \cup O \setminus Q : x(i) < l_{k+1}^l \text{ and } x(i) > l_{k-1}^r \} \\ \text{ for } k = 2,...,K-1 \\ \{ i \in I \cup O \setminus Q : x(i) > l_{k-1}^r \} \text{ for } k = K \end{cases} \tag{1}$$

All feasible children nodes of the current node are generated by adding each element of P^k to S^k.

Step 4: Calculate lower bounds of all feasible children nodes

Step 5: If containers sequences of all cranes within a node are not empty, the upper bound of the node is assessed. If upper bound of a node is smaller than CUB, the CUB is updated.

Step 6: If the lower bound of a feasible node is less than or equal to the CUB, the feasible children node is added to the search tree as a new branch.

Step 7: If all containers at the new branch are assigned to cranes, the algorithm stops. Otherwise, return to Step 2.

1. Lower bounding (LB) procedure

In this section, we develop a lower bound for the multiple rail cranes scheduling problem. The notation for calculating the lower bound of the objective value is illustrated in Figure 3. The working areas of all cranes are separated into many partitions based on the boundaries of rail cranes. The boundaries of a crane’s working area of a node in a search tree are the farthest wagon on the left (“left boundary”), and the farthest wagon on the right (“right boundary”), that were assigned to a crane. The unassigned area located outside the boundary of crane k is denoted as L^k for $k = 1,...,K\text{-}1$, which is used to express the travel time of cranes along the area. The unassigned areas are separated into many segments based on the location of containers in those areas. The lengths of the m^{th} segments are denoted as l_m^k for $k = 1,...,K-1$, which is expressed by the travel time of cranes.

The lower bound of the scheduling problem of multiple rail cranes is calculated as follows:

$$LB = Max\left(\bar{C}_{\max}, \left(\sum_{k=1}^{K} \bar{C}_k + \tau \right) / K \right) \tag{2}$$

where:

$$\bar{C}_{\max} = \max_{k=1,...,K} \bar{C}_k \tag{3}$$

$$\bar{C}_k = c_k + t_k \text{ for } k = 1,...,K \tag{4}$$

$$t_k = 2t_{\min}^k + t_{\max}^k + \sum_{i \in R^k} d_{x(i),T+2}^{x(i),y(i)} \text{ for } k = 1,...,K \tag{5}$$

$$\tau = \sum_{k=1}^{K-1} \left(L^k - l_{\max}^k \right) + \sum_{i \in I \cup O \setminus Q} d_{x(i),T+2}^{x(i),y(i)} \tag{6}$$

$$L^k = \sum_{n=1}^{N} l_m^k \text{ for } k = 1,...,K-1 \tag{7}$$

and

$$l_{\max}^k = Max\left(l_m^k \right) \text{ for } k = 1,...,K-1 \tag{8}$$

The following explanations demonstrate why this calculation is the lower bound of the makespan of all cranes. In equation (2), $\bar{C}_{\max}$ is the maximum among the lower bounds of the operation time of cranes, $\bar{C}_k$, within their working areas. $\bar{C}_k$ is the lower bound of the operation time of crane k because c_k is the total time required to handle all containers in the job sequence of crane k, and t_k, the minimum travelling time of the crane k to handle all remaining containers in its working zone.

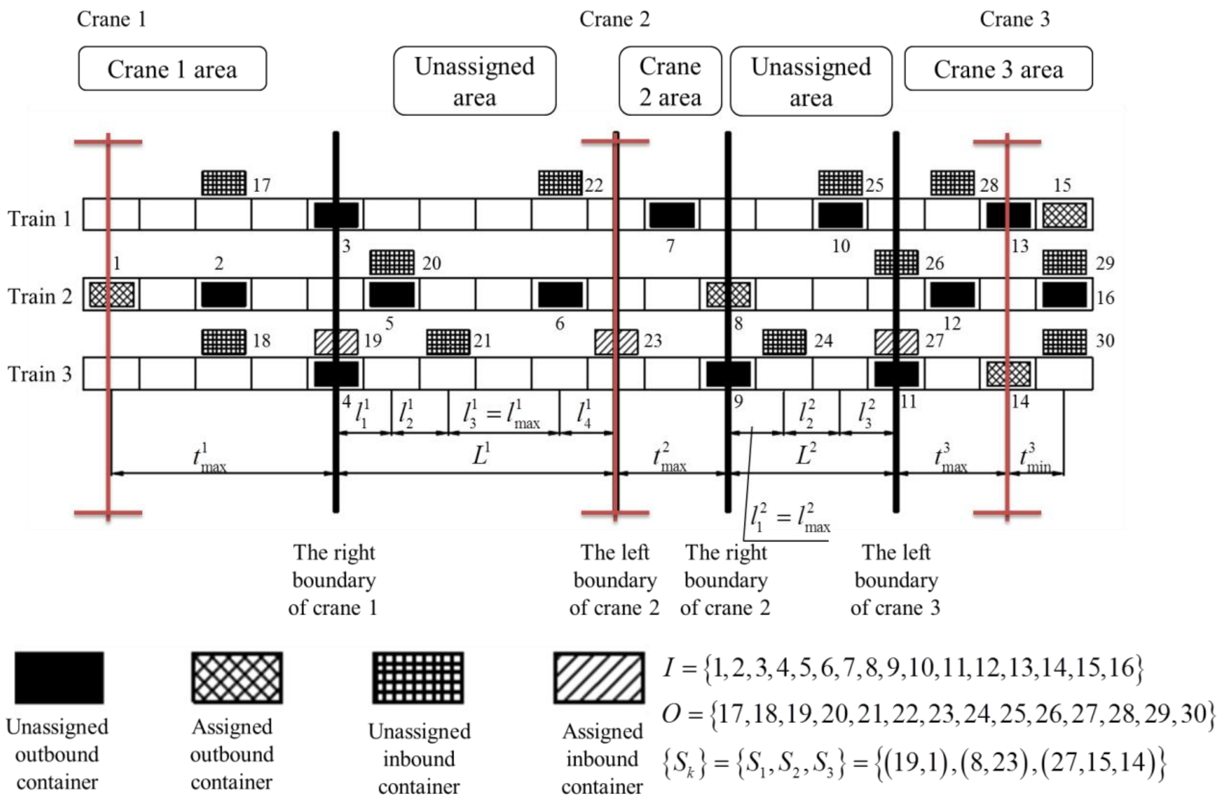

Figure 3. An illustration of notation for the lower bound of the multiple rail cranes scheduling problem.

In (5), the value of t_k includes two elements as $2t^k_{\min} + t^k_{\max}$ and $\sum_{i \in P^k} d^{x(i),y(i)}_{x(i),T+2}$.

The following explanations demonstrate why this calculation is the lower bound of the makespan of all cranes. In equation (2), $\bar{C}_{\max}$ is the maximum among the lower bounds of the operation time of cranes, $\bar{C}_k$, within their working areas. $\bar{C}_k$ is the lower bound of the operation time of crane k because c_k is the total time required to handle all containers in the job sequence of crane k, and t_k, the minimum travelling time of the crane k to handle all remaining containers in its working zone. In (5), the value of t_k includes two elements as $2t^k_{\min} + t^k_{\max}$ and $\sum_{i \in P^k} d^{x(i),y(i)}_{x(i),T+2}$. The first element is the minimum travelling time of the gantry from the lowest boundary to the farthest boundary. The second element is the minimum travelling time required for a crane to load and unload a container. The calculation of t_k has ignored all travelling times of the crane between containers and all re-handling movement times of the crane, and hence, $\bar{C}_{\max}$ is the lower bound of the makespan. Similarly, to t_k, τ is the minimum travelling time of cranes to finish all the containers within the zone, which have not been assigned to any crane. Thus, the summation of τ and all $\bar{C}_k$ divided by K is the lower bound of the makespan.

2. Upper bounding (UB) procedure

The upper bound of the makespan of all cranes is obtained by the heuristic "Travel to the Right" (Jeong and Kim, 2011). At a node, the set of containers assigned to crane k is generated as $V^k = \{i \in P^k : x(i) \geq l^l_k\}$. The dual-cycle operation of a crane is constructed by the matching phase, and the job sequences of cranes are obtained by the 'Travel to the Right' procedure. The next two subsections will describe these procedures in detail.

2.1. The matching phase

In the matching phase, inbound containers and outbound containers are matched within a single operational cycle. The matching phase results in sets of inbound and outbound containers being scheduled as tasks for cranes. There are two types of tasks as 1) single task, in which the cranes need to perform single-cycle operations, and 2) dual tasks, in which the cranes need to perform dual-cycle operations. In this process, the 'Hungarian method' is applied to match inbound and outbound containers together. The cost matrix is calculated as follows:

$$c_{ij}=\begin{cases} d_{x(i),y(i)}^{x(i),T+2}+d_{x(j),y(j)}^{x(i),y(i)}+d_{x(j),T+2}^{x(j),y(j)} & \text{for } i\in I \text{ and } j\in O \\ pd_{x(i),y(i)}^{x(i),T+2}+d_{x(j),y(j)}^{x(i),y(i)}+pd_{x(j),T+2}^{x(j),y(j)} & \text{for } i\in I \text{ and } j\in O \\ 3d_{x(i),T+2}^{x(i),y(i)} & \text{for } i\in I \text{ and } j\in O \\ d_{x(i),y(i)}^{x(i),T+2} & \text{for } i\in I \\ d_{x(i),y(i)}^{x(i),T+2} & \text{for } i\in O \end{cases} \quad (9)$$

(i) The inbound container, i, is either on the left of the outbound container, j, or is located closer to the transfer track than j,

$x(i)<x(j)$;or $x(i)=x(j)$ and $y(i)>y(j)$

(ii) The inbound container, i, is either on the right of the outbound container, j, or is located further from the transfer track than j,

$x(i)>x(j)$;or $x(i)=x(j)$ and $y(i)<y(j)$. In this case, the containers have a greater likelihood of being re-handled. Hence, we assign a penalty value "p" to avoid these issues.

(iii) The inbound container, i, and the outbound container, j, are located at the same place on the trains, $x(i)=x(j)$ and $y(i)=y(j)$.

(iv) The crane performs a single cycle operation with an inbound container.

(v) The crane performs a single cycle operation with an outbound container.

2.2. Sequencing phase

In this phase, the cranes travel from the left to the right and serve those containers which are closer to the transfer track first. The process of this algorithm is as follows:

Step 1: $\{S_k\}$ for $k=1,\dots,K$ is the current node in the search tree, which needs to be calculated the upper bound. Let $C_k{}^1$ be the set of all tasks assigned to crane k in the matching phase. Let $C_k{}^2$ be the final task sequence of crane k. Let $C_k{}^3$ be the set of all tasks including re-handling work.

Step 2: A crane k is chosen in the order of 1 to K. Set $C_k{}^2=\varnothing$ and $C_k{}^3=\varnothing$.

Step 3: Choose a task in $C_k{}^1$ that is located furthest to the left and nearest the transfer track. Remove the task in $C_k{}^1$. The task is added to $C_k{}^2$. If a task needs to be re-handled, the task is added to $C_k{}^3$.

Step 4: If $C_k{}^1$ is not empty, repeat Step 3. Otherwise, go to Step 5

Step 5: Choose the first task in $C_k{}^3$ and remove it from $C_k{}^3$. Add the task into the sequence of $C_k{}^2$ to minimize the finishing time of crane k.

Step 6: If $C_k{}^3$ is not empty, go to Step 5. Otherwise, go to Step 7.

Step 7: If all cranes tasks are completed, the algorithm stops. Otherwise, go to Step 2.

IV. SIMULATED ANNEALING ALGORITHM

The simulated annealing algorithm is developed to find near optimal solutions of the multiple rail crane scheduling problem. This section will describe operators of the SA.

1. Initial solution construction

A solution of the scheduling problem consists of two parts as 1) the working area and 2) the job sequence of each crane. The initial working areas are assigned to cranes based on the balance of the workload. The "workload" is the summation of all travelling times to discharge outbound containers or to retrieve inbound containers (Boysen and Fliedner 2010) We use the example in the previous section to illustrate how to construct the initial solution.

We assume that the travelling time between two adjacent trains is one-time unit, and that the temporary storage area is located between the transfer track and the railway track. Table 1 presents the travelling time of cranes to handle the containers. Total workload is 41 time units. Therefore, the working area of the first crane is from wagon number 1 to wagon number 5. The working area of the second crane is from wagon number 6 to wagon number 10. The workload of the first crane is 20. The workload of the second crane is 21. Table 3 illustrates a solution for this example. The first part represents the working areas of the cranes.

Table 1. An example of inbound and outbound containers.

Container	Type	Train	Wagon	Workload	Container	Type	Train	Wagon	Workload
1	Inbound	1	1	3	9	Inbound	1	9	3
2	Inbound	2	1	2	10	Outbound	2	1	2
3	Inbound	1	2	3	11	Outbound	1	2	3
4	Inbound	2	3	2	12	Outbound	1	6	3

5	Inbound	1	5	3	13	Outbound	2	8	2
6	Inbound	2	5	2	14	Outbound	2	8	2
7	Inbound	1	6	3	15	Outbound	1	9	3
8	Inbound	1	7	3	16	Outbound	2	9	2

Table 2. Solution representation.

Crane	Boundary	Job sequence								
1	5	1	2	11	6	5	10	3	4	
2	10	7	12	13	9	15	16	8	14	16

Table 3. Numerical examples.

No	Number of trains	Inbound/ Outbound Containers	Number of cranes	B&B		SA		
				Makespan, (LB/UB)	CPU Time (sec.)	Makespan	(Std.)	CPU Time (sec.)
1	1	12/8	2	50.00	1	50.00	(0.00)	128
2	1	16/16	2	66.00	417	66.00	(0.00)	165
3	1	24/21	2	78.00	2348	78.00	(0.00)	222
4	1	30/31	2	94.00/98.00	**	96.00	(0.00)	279
5	2	14/17	2	77.00/79.00	**	79.80	(1.11)	149
6	2	36/28	2	118.00/140.00	*	130.00	(0.00)	292
7	2	49/48	2	152.50/192.00	*	171.45	(1.10)	409
8	2	64/61	2	190.00/215.00	*	212.80	(2.12)	543
9	3	25/23	2	111.00/138.00	**	120.00	(0.00)	234
10	3	51/46	2	183.00/242.00	*	211.80	(1.61)	408
11	3	72/81	2	269.00/327.00	*	314.00	(5.48)	664
12	3	97/97	2	323.00/369.00	*	384.15	(8.16)	933
13	1	12/8	3	36.00	1201	36.00	(0.00)	165
14	1	16/16	3	46.00	215	46.00	(0.00)	191
15	1	24/21	3	50.67/54.00	**	53.30	(2.36)	233
16	1	30/31	3	62.67/64.00	**	64.00	(0.00)	362
17	2	14/17	3	49.33/53.00	**	53.00	(0.00)	193
18	2	36/28	3	77.00/97.00	**	87.30	(2.64)	367
19	2	49/48	3	101.00/126.00	*	114.70	(2.32)	562
20	2	64/61	3	126.33/141.00	*	142.60	(2.64)	661
21	3	25/23	3	72.67/91.00	**	83.35	(2.18)	247
22	3	51/46	3	118.33/152.00	**	142.20	(3.14)	554
23	3	72/81	3	178.33/217.00	*	211.70	(2.03)	866
24	3	97/97	3	215.33/256.00	*	252.85	(7.14)	1133

Note: * Computation time exceeds 86400 seconds.

** The computer is out of memory

The numbers in this part denote the wagon numbers of the last right hand side areas area which was handled. The second and third parts show the work sequencing of cranes handling inbound and outbound containers, and the number denotes the container number. In Table 2, the inbound containers 2, 7 and 9, and the outbound containers 11, 12 and 15, require re-handling.

2. Neighborhood solutions

Two operations are designed to generate neighborhood solutions. The first operation is a "zone operator", which is used to determine a new working area for a crane. The second operator is "sequence operator" used to generate a new job sequence for the crane. If a random number in the range of (0, 1) is less than or equal to α , the first operation is chosen. Otherwise, the second operation is applied. In the zone operator, the boundary of a crane is first adjusted by adding randomly -1, 0 or 1. The containers that do not satisfy the boundary are assigned to other cranes. In the sequence operator, a pairwise interchange is carried out to adjust the job sequence, where two inbound containers or two outbound containers are randomly selected and swapped.

3. Acceptance criterion

Once a neighborhood solution is generated, the following criterion is used to accept or reject it. Let $\Delta = Q(C) - Q(NB)$ where C, NB and Q(*) are the current solution, neighborhood solution and makespan of solution (*), respectively. A neighborhood solution is accepted as the current solution if $\Delta \geq 0$ or $r \leq e^{\frac{\Delta}{T_l}}$, where r is a uniform random number in (0,1) and T_l represents the current temperature. The cooling down function is the geometric update scheme introduced by Lundy and Mees (1986).

V. NUMERICAL EXAMPLE

In order to evaluate the performance of the SA, and the B&B algorithms, 24 problems were randomly generated by uniform distribution. The probabilities that a wagon carries a container (an inbound or an outbound container) are 0.25, 0.50, 0.75 and 0.95, respectively. The examples were tested on a computer with an AMD Athlon CPU 3.00 GHz, and 4 GB memory, and the algorithms are coded in Java. The parameters of SA are chosen, α as 0.3, T_0 as 20, π as 0.999999819 and ε as 0.001. In Korea, rail stations usually have three rail tracks, while each train contains an average of thirty-three wagons. Table 3 shows the mean values of makespans, standard deviations and computation times of the 24 problems, as obtained by these methods; the branch-and-bound algorithm, the simulated annealing algorithm. For small-sized problems, all methods gave the same results. For large-sized problems, the branch-and-bound could not return an optimal solution, while the SA algorithm returned optimal solutions within 20 minutes.

VI. CONCLUSION

In this paper, we considered a multiple crane scheduling problem in which working areas are assigned to cranes and the job sequence of each crane is considered. Moreover, the dual-cycle operation of cranes was incorporated. The makespan of cranes was used as an optimization criterion, and a simulated annealing algorithm was proposed to find near optimal solutions. The results of this algorithms were compared with the results from the B&B. We investigated the performance of algorithms using numerical examples. In generally, the simulated annealing algorithm can give the results in several minutes. Further research may include the study of rail crane scheduling problems with the additional consideration of train timetables, or with the integration of truck scheduling.

VII. REFERENCES

1. Alicke, Knut. 2002. "Modeling and optimization of the intermodal terminal Mega Hub." OR Spectrum 24 (1):1-18.
2. Boysen, N., and M. Fliedner. 2010. "Determining crane areas in intermodal transshipment yards: The yard partition problem." European Journal of Operational Research 204 (2):336-342.
3. Boysen, N., M. Fliedner, and M. Kellner. 2010. "Determining fixed crane areas in rail-rail transshipment yards." Transportation Research Part E-Logistics and Transportation Review 46 (6):1005-1016..
4. Boysen, Nils, Malte Fliedner, Florian Jaehn, and Erwin Pesch. 2013. "A survey on container processing in railway yards." Transportation Science 47 (3):312-329. Carlo, HéctorJ, IrisF A. Vis, and KeesJan Roodbergen. 2013. "Seaside operations in container terminals: literature overview, trends, and research directions." *Flexible Services and Manufacturing Journal*:1-39.
5. Guo, Peng, Wenming Cheng, Zeqiang Zhang, Min Zhang, and Jian Liang. 2013. "Gantry Crane Scheduling with Interference Constraints in Railway Container Terminals." International Journal of Computational Intelligence Systems 6 (2):244-260.
6. Jeong, B. J., and K. H. Kim. 2011. "Scheduling operations of a rail crane and container deliveries between rail and port terminals." Engineering Optimization 43 (6):597-613.
7. Kim, Kap Hwan, and Young-Man Park. 2004. "A crane scheduling method for port container terminals." European Journal of Operational Research 156 (3):752-768.
8. Lundy, M., and A. Mees. 1986. "Convergence of an annealing algorithm." Mathematical Programming 34 (1):111-124.
9. Nguyen, Duy Vu Anh, and Won Young Yun. 2014. "Optimal job scheduling of a rail crane in rail terminals." International Journal of Industrial Engineering: Theory, Application and Practice 21(3): 129-140.
10. Pap, E., G. Bojanic, N. Ralevic, M. Georgijevic, and V. Bojanic. 2012. "Crane scheduling method for train reloading at inland intermodal container terminal." Intelligent Systems and Informatics (SISY), 2012 IEEE 10th Jubilee International Symposium on, 20-22 Sept. 2012.