

Minimizing Empty Trips of Yard Trucks in Container Terminals by Dual Cycle Operations

Vu Duc Nguyen

Department of Industrial Engineering
Pusan National University, 30 Jangjeon-dong, Geumjeong-gu, Busan 609-735, South Korea
Tel: +82-51-510-2419, Fax: +82-51-512-7603, Email: duc.ise@pusan.ac.kr

Kap Hwan Kim[†]

Department of Industrial Engineering
Pusan National University, 30 Jangjeon-dong, Geumjeong-gu, Busan 609-735, South Korea
Tel: +82-51-510-2419, Fax: +82-51-512-7603, Email: kapkim@pusan.ac.kr

Received, October 14, 2009; Revised, December 8, 2009; Accepted, December 8, 2009

Abstract. One of the most important objectives of the schedules in a container terminal is to minimize the ship operation time, which consists of discharging and loading operation times. Recently, dual cycling techniques have been used for improving terminal operations, especially for reducing the total empty trips of handling equipment. The main focus of this study is to reduce the empty trip times of yard trucks with minimum delays for ship operations. A heuristic algorithm, modified from a previous algorithm, is proposed to solve this problem. A simulation study is conducted to evaluate the effect of different types of discharging and loading schedules and different locating methods for discharging containers in terms of the performance of the system, including the percentage of the dual cycle operations of yard trucks.

Keywords: Dual Cycling, Ship Operation Schedule, Container Terminal, and Simulation

1. INTRODUCTION

The transport of containers via seaport container terminals has rapidly increased over the last decade. Operation efficiency of container terminals has long been considered a challenging problem. An important factor for evaluating the efficiency of container terminal operations is the ship operation time. Ship operations consist of discharging and loading operations. There are three main types of handling equipment involved in ship operations: quay cranes (QCs), yard trucks (YTs), and yard cranes (YCs). For a discharging operation, a container is unloaded from a ship by a QC and transported to the assigned block by a YT. In the yard, it is picked up and stacked in storage blocks by a YC. For a loading operation, a container is handled in a way exactly opposite to the discharging operation. Figure 1 shows the layout of a container terminal assumed in this study.

This study attempts to reduce the empty trips of YTs by combining the travel of a YT for discharging a container with that of a YT for loading a container. The main concept is that the empty trip of YTs can be reduced by (1) simultaneous determination of the vehicle

dispatching and storage location of the containers to be discharged and (2) increasing the degree of mixture for loading and discharging operations by QCs.

With regard to the first point, storage locations of the containers to be discharged are usually predetermined in advance before the ship operation begins in practice. However, the strategy suggested in this paper is that the storage locations of the containers to be discharged should be determined during the discharging operations. During the assignment of delivery orders to YTs, the storage locations of the containers to be discharged are considered to be decision variables.

With regard to the second point, in practice, the loading operation can only begin after all the discharging operations are completed; this results in few chances for combining the travel of loading YTs with discharging YTs to reduce the empty trips. This study analyzes the effect of mixing loading and discharging operations for QCs on reducing the empty trips of YTs.

In most previous researches, the vehicle dispatching and storage location problems were separately considered. For the vehicle dispatching problem, Kim and Bae (2004) presented a mixed integer programming

[†] : Corresponding Author

model and a heuristic algorithm for an automated guided vehicle (AGV) schedule problem for minimizing the total delay times of QCs and the total travel times of AGVs. The model assumed that the sequence of discharging and loading by QCs and the storage location of containers were given in advance. Grunow *et al.* (2006) introduced a simulation study of AGV dispatching strategies in a seaport container terminal, where AGVs can be used in either single- or dual-carrier mode. They compared a typical online dispatching strategy adopted from flexible manufacturing systems with a pattern-based offline heuristic model. Briskorn *et al.* (2006) presented an alternative formulation of the AGV assignment problem that did not include due times and was based on a rough analogy to inventory management; they proposed an identical algorithm for solving the formulation. Lee (2007) introduced an exact dynamic programming algorithm for selecting dwell points in a tandem-loop multiple-vehicle AGV system. The objective is to minimize the maximum response time for all pickup requests in a given shift. The algorithm considers time restrictions on the availability of vehicles during the shift. Nguyen and Kim (2009) proposed a mathematical formulation for solving the dispatching problem for automated lifting vehicles (ALVs). A heuristic algorithm was suggested, and the solutions of the heuristic algorithm were compared with optimal solutions. Numerical experiments were performed to analyze and evaluate the performance of the heuristic algorithm.

Bish *et al.* (2001) analyzed the container location and vehicle dispatching problems simultaneously, which is called the vehicle-scheduling-location problem. The problem was to assign each container to a yard location and dispatch vehicles to the containers so as to minimize the time needed to discharge all the containers from the ship. A heuristic algorithm was developed by decomposing the vehicle-scheduling-location problem into two isolated steps, where the first step determined the location assignments by ignoring the vehicle schedule and the second step determined the vehicle schedule for the location arrangements obtained from the first step. However, the solution approach did not still solve the container location and vehicle dispatching problems in an integrated manner.

Lee *et al.* (2009) proposed a novel approach that integrated the container location and vehicle dispatching problems into one. The objective was to minimize the weighted sum of the total delay of requests and the total travel time of the YTs. A hybrid insertion algorithm was designed for obtaining effective problem solutions.

Recently, dual cycle operations have been performed for improving ship operations in container terminals. Dual cycle operations were considered in previous studies by Goodchild and Daganzo (2006 and 2007). In their papers, they suggested a scheduling method for stacks under a single hatch cover and showed that Johnson's rule can be applied to sequencing the discharging and loading tasks for stacks. They also provided a method

to evaluate the effect of dual cycle operations on the reduction in the number of cycles during the ship operation and also analyzed the impact of the dual cycle operations on the land side operations. Zhang and Kim (2009) extended the study conducted by Goodchild and Daganzo (2006) to the sequencing problem not only for stacks under a hatch cover but also for hatches. They attempted to minimize the number of operation cycles for a QC when discharging and loading containers in a ship bay by maximizing the number of dual cycle operations.

This paper provides the following new contributions: (1) unlike previous studies which dealt with dual cycle operations of QCs (Goodchild and Daganzo, 2006 and 2007; Zhang and Kim, 2009), this paper addresses the dual cycle operation of YTs; (2) a new operation strategy in which the dispatching decision and the storage location decision are made simultaneously is suggested; and (3) the impact of the QC operation type on the YT travel distance is analyzed.

This paper is organized as follows. The relationship between YT dispatching, storage location, and the type of ship operation are discussed in Section 2. Section 3 introduces modifications of a previous mixed integer programming model and a heuristic algorithm (Kim and Bae, 2004) required for solving the new integrated decision-making problem. In Section 4, a simulation study is implemented to analyze the effect of different operation strategies on the performance of the system in dynamic situations. Finally, Section 5 provides some concluding remarks and gives a brief summary of the further studies.

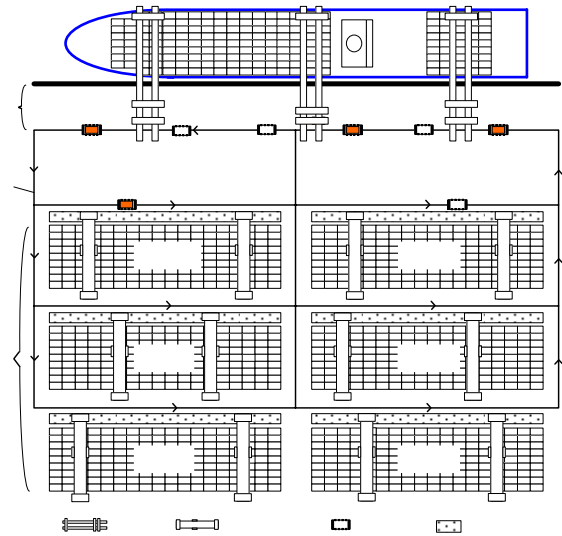


Figure 1. Layout of a container terminal.

2. SHIP OPERATIONS AND DUAL CYCLE OPERATIONS

Before a ship arrives at a port container terminal,

all information regarding the inbound and outbound containers is sent to the terminal by a shipping agent. Then, based on this information, a sequence list of the discharging and loading operations for individual containers is then made. When the ship actually arrives, ship operations are usually performed on the basis of the loading and discharging sequence list.

At the quay side, a discharging operation by a QC ends with the release of a container onto a YT, while a loading operation by a QC starts with the pickup of a container from a YT. Dual cycle operations at the quay side can be performed to reduce the ship operation time. Figure 2 shows a comparison between single and dual cycle operations. In single operations, when a QC handles loading operations after all discharging operations have been finished—which is a common practice currently—an empty trip is needed for every cycle. In dual cycle operations, the QC performs a discharging operation after finishing a loading operation. This means that the discharging and loading operations are combined into one cycle to reduce empty trips in comparison with single cycle operations. A similar procedure can be followed for YTs and YCs.

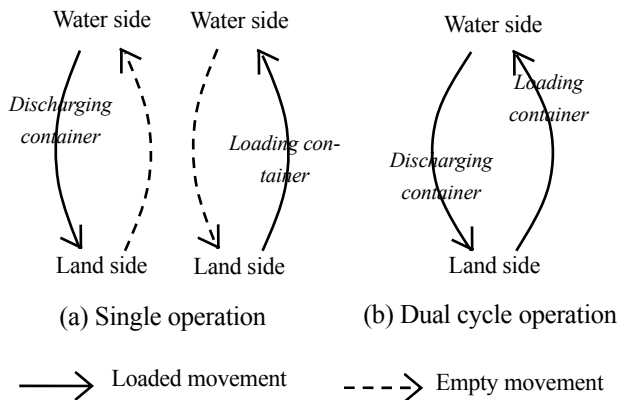


Figure 2. Comparison between single and dual cycle operations.

Figure 3 illustrates three different types of QC operations. In the first type of QC operation shown in Figure 3(a), the discharging and loading operations are performed independently and thus there is no chance for YTs to perform their operation in dual cycles. In the second type of QC operation shown in Figure 3(b), although each QC performs either discharging or loading operations only, when one QC is discharging and another is loading, there is a chance, for YTs to perform their delivery operations in dual cycles. In the third type of QC operation shown in Figure 3(c), the QCs perform their operations in dual cycles and thus the discharging and loading operations for a QC are completely mixed. Thus, this type has the highest chance for YTs to perform their operations in dual cycle. A detailed description of the third type of QC operation is illustrated in

Figure 4, which describes a part of the schedule for QC 01. This figure shows how the dual cycle operation becomes possible by pairing the loading operations for one stack with the discharging operations for another stack.

QC 01	Discharging	Bay 01	Bay 03	Bay 05			
	Loading				Bay 05	Bay 03	Bay 01
QC 02	Discharging	Bay 07	Bay 10	Bay 12			
	Loading				Bay 12	Bay 10	Bay 07

(a) Discharging first

QC 01	Discharging	Bay 01		Bay 03		Bay 05	
	Loading		Bay 01		Bay 03		Bay 05
QC 02	Discharging	Bay 07	Bay 10		Bay 12	Bay 14	
	Loading			Bay 10			Bay 14

(b) Mixture of bays

QC 01	Discharging	Bay 01	Bay 03	Bay 05			
	Loading		Bay 01	Bay 03	Bay 05		
QC 02	Discharging	Bay 07	Bay 10	Bay 12	Bay 14		
	Loading			Bay 10		Bay 14	

(c) Mixture of stacks

Figure 3. Three different types of QC operations.

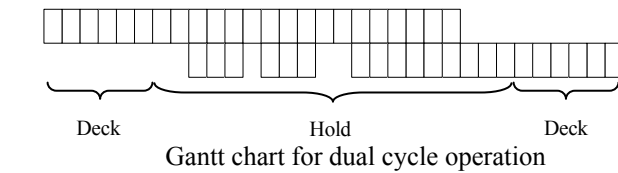
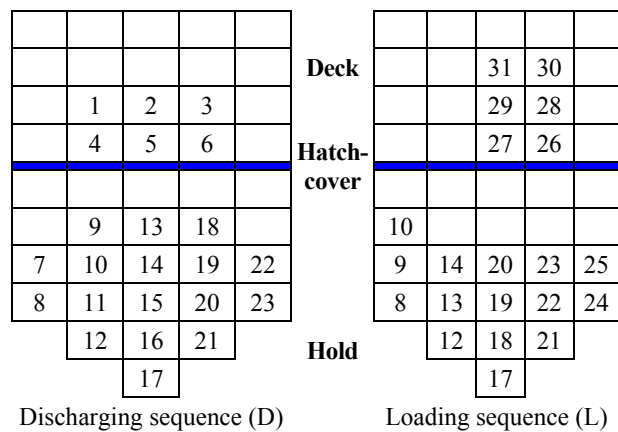


Figure 4. Operation sequence and Gantt chart for dual cycle operations performed by QC 01.

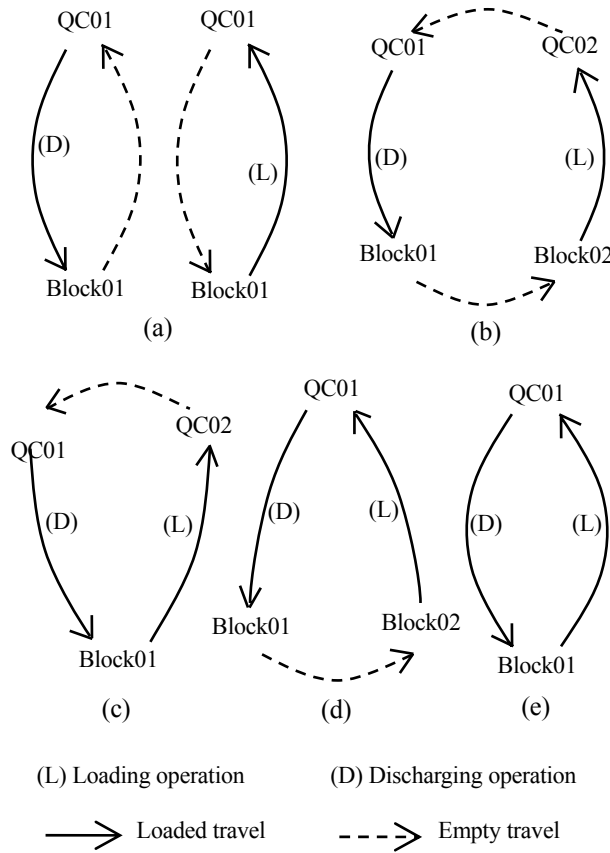


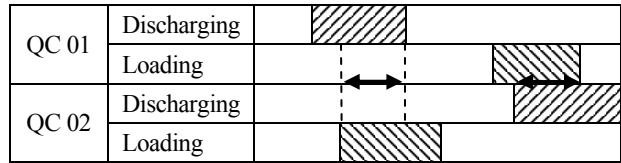
Figure 5. Illustration of different types of operations performed by YT.

There are five different types of YT operations, as shown in Figure 5. Figure 5(a) illustrates single operations in which half of the travels are empty. All YTs assigned to the QCs have to perform the delivery operations in single cycles when (1) all the QCs are performing discharging operations; (2) they are performing only loading operations; or (3) two QCs performing loading and discharging operations are too far apart from each other. Figure 5(b) illustrates YT dual cycle operations where YTs have to make empty trips between two QCs and two blocks. This type of YT operation is possible when two QCs are performing different types of operations simultaneously—as shown in the overlapped part indicated by the arrow in Figure 6(a)—and YTs are delivering containers between blocks different from each other. When the blocks assigned to the two QCs are the same, the operation type shown in Figure 5(c) is performed.

Figure 5(d) illustrates the YT dual cycle operation where YTs have empty trips between two blocks but no trip between QCs. This type of YT operation is possible during the dual cycle interval shown in Figure 6(b) and when the blocks assigned to QCs are different from each other. The operation shown in Figure 5(e) can be realized

when a QC performs loading and discharging operations in dual cycles, and the storage locations of the containers to be discharged are the same as those for the containers to be loaded.

This study took advantages of the opportunities for YT dual cycle operations shown in Figures 6(a) and 6(b). In addition, this study also attempted to change the YT operation from the case shown in Figures 5(a) and (b) to that shown in Figure 5(c) or from Figure 5(d) to that shown in Figure 5(e). This was done by assigning the storage locations of the containers to be discharged at the same time when delivery tasks were assigned to YTs, which is called the “postponed assignment of storage locations” strategy in this paper.



(a) Intervals for dual cycle operation performed by YTs serving different QCs



(b) Intervals for dual cycle operation performed by YTs serving the same QCs

Figure 6. Two different cases of QC operations.

Table 1. Example of a sequence list of QC 01.

QC 01					
Task sequence	Type [†]	Ship location [‡]	Yard location [‡]	Operation cycle time	Earliest event time
1	L	11/03/01	01/21/3/2	130	0
	U	11/04/10	*		
2	U	11/04/09	*	110	230
3	L	11/03/02	02/21/3/1	70	240
4	L	11/03/03	01/17/4/2	170	350
	U	11/04/08	*		
...

Notes) [†]L-loading; U-unloading; LU-dual cycle; [‡]ship bay no./row no./tier no.; [‡]yard block no./yard-bay no./row no./tier no.; *The yard location of discharging operations is determined during dispatching decision.

Table 1 shows an example of a sequence list for ship operations performed by QC 01, which comprises the discharging operations, loading operations, and the dual cycle operations performed at the quay (in the second column). The ship locations of the containers are given in the third column. In the fourth column, the yard locations of the loading operations are given, while

those of the discharging operations are determined during the dispatching decision. The last two columns show the operation cycle times and the earliest QC starting times.

3. DISPATCHING DELIVERY TASKS TO YTs

This section introduces a dispatching procedure modified from the algorithm suggested by Kim and Bae (2004). The following assumptions were introduced into this study:

- (1) A pooling strategy for YTs, in which a YT can deliver containers for any QCs, was applied to the dispatching process.
- (2) All YTs were the same and transferred one container at a time.
- (3) YT congestion in on driving lanes was not considered.

The original version of the algorithm (Kim and Bae, 2004) is summarized here. However, some modifications were needed to integrate the decisions for vehicle dispatching and locating discharging containers.

The following notations are used in formulation.

e_i^k = An event representing the moment that a YT transfers the i^{th} task of QC k . The QC task can be classified as a loading, discharging, or dual cycle operation. When the i^{th} task of QC k is a loading operation

or a dual cycle operation, the event e_i^k corresponds to the beginning of the pickup of a container from a YT by QC k . When the i^{th} task of QC k is a discharging operation, it corresponds to the beginning of the release of a container onto a YT by QC k .

y_i^k = The event time of e_i^k which is a decision variable.

s_i^k = The earliest possible event time of y_i^k .

V = The set of YTs.

K = The set of QCs.

e_j^o = The starting event of YT j , $j \in V$.

e_j^f = The stopping event of YT j , $j \in V$.

m_k = The number of tasks for QC k .

$l(e_i^k)$ = The location where the event e_i^k occurs. $l(e_j^o)$ represents the initial location of YT j . $l(e_i^k)$ represents the position where the i^{th} container of QC k is to be transferred. $l(e_j^f)$ represents the location where a YT completes its final delivery task.

Bp_j^l = The location where the j^{th} succeeding loading container of QC l will be picked up.

Bs_i^k = The location where the i^{th} discharging container of QC k is stacked.

Bl_j^l = The location where the i^{th} discharging container of QC k is stored and where the j^{th} succeeding loading container of QC l is to be picked up.

t_{ki}^j = The pure travel time from $l(e_i^k)$ to $l(e_j^l)$.

c_{ki}^j = The time required for an YT to be ready for e_j^l

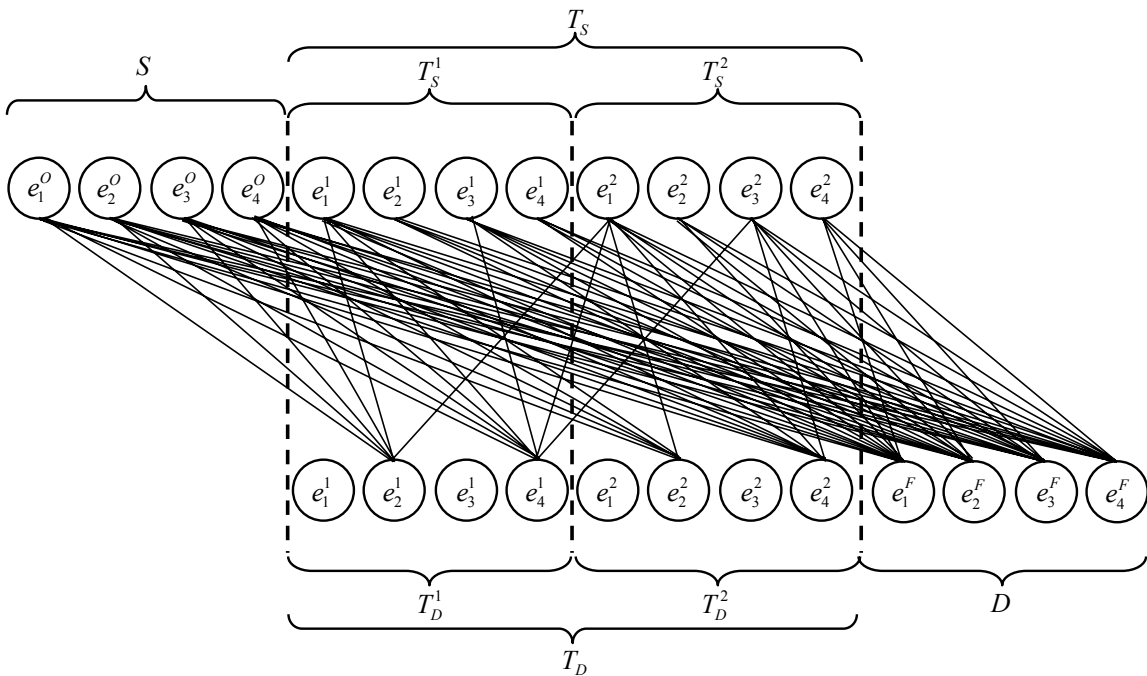


Figure 7. Dispatching graph for QC 01 and QC 02.

after it experiences e_i^k .

S = The sets of e_j^o , $j \in V$.

D = The sets of e_j^f , $j \in V$.

T = The set of e_i^k for $i=1, 2, \dots, mk$ and $k \in K$.

x_{ki}^{lj} = A decision variable that becomes 1 if e_i^k is assigned to e_j^l , for $k \in K'$ and $l \in K''$, where $K' = \{O\} \cup K$, $K'' = \{F\} \cup K$. For k and $l \in K$, the assignment of e_i^k to e_j^l implies that the YT, which have just delivered the i th container of QC k , is scheduled to deliver the j th container of QC l . An example of a solution of x_{ki}^{lj} s is shown in Figure 10.

Figure 7 illustrates a graph consisting of tasks for two QCs and four YTs. Nodes in T_S^k correspond to supply nodes from tasks for QC k , while those in T_D^k represent demand nodes from tasks for QC k . Note that the total number of supply nodes in T_S and the total number of demand nodes in T_D are equal; this is denoted by T . Nodes in S represent available YTs, while nodes in D represent YTs after finishing their assigned tasks. Note that the graph has nodes on the upper side to represent events supplying empty YTs and nodes on the lower side to represent events demanding empty YTs. Dispatching can be defined as finding a one-to-one match between nodes on the upper and lower sides. However, a match is only possible when there is an arc from a node on the upper side to the node on the lower side. An arc can only be connected from a node on the upper side (e.g. node e_i^k) to another node on the lower side (e_j^l) when a YT can reach the location of node e_j^l before the event time of e_j^l after leaving the location of node e_i^k at the event time for e_i^k . Figure 7 illustrates the graph for the tasks of QC 01 and QC 02 on which all possible arcs are connected.

Figure 8 illustrates a feasible match. The algorithm described in this paper attempted to find a feasible match with minimizing the total delay in QC operations as the

primary objective and the total travel time of vehicles as the secondary objective.

Normally, the storage location of a discharging container is predetermined by a planner in the terminal. The storage space is allocated to a group of discharging containers considering various factors, including expected future congestions at each block, space plans for other vessels, and ship operation plans. Based on the space allocation, the storage location of an individual discharging container is determined in real-time. In other words, we assume that the storage locations of discharging containers are already determined but that they may be changed in real time. Even in this case, the storage location can be changed within the boundary of the total space allocation to a group of discharging containers specified by the planners.

Suppose that a storage location of a discharging container is to be changed so that the travel distance from the block for a preceding discharging container to the pickup location for a succeeding loading container can be minimized. We may have several candidate blocks which have remaining allocated space for the group of discharging containers. In this case, we can select the block for the discharging container located on the shortest route to the loading container in the candidate blocks.

With this strategy for locating discharging containers, we can use the same mathematical model and apply the same heuristic algorithm to solve the integrated dispatching problem after modifying t_{ki}^{lj} and c_{ki}^{lj} for each of the following cases. The modification is summarized in Table 2 and Figure 9. For example, in case (1) of Figure 9, the travel time should be evaluated considering the travel time for a YT from a loading task to the block Bp_j^l and the travel time from the block to the QC l of the j^{th} loading container. Case (2) of Figure 9 shows the travel time for a YT visiting the QC for the next discharging container after completing a loading container. In case (3), the travel time includes the time for a YT trip to the block

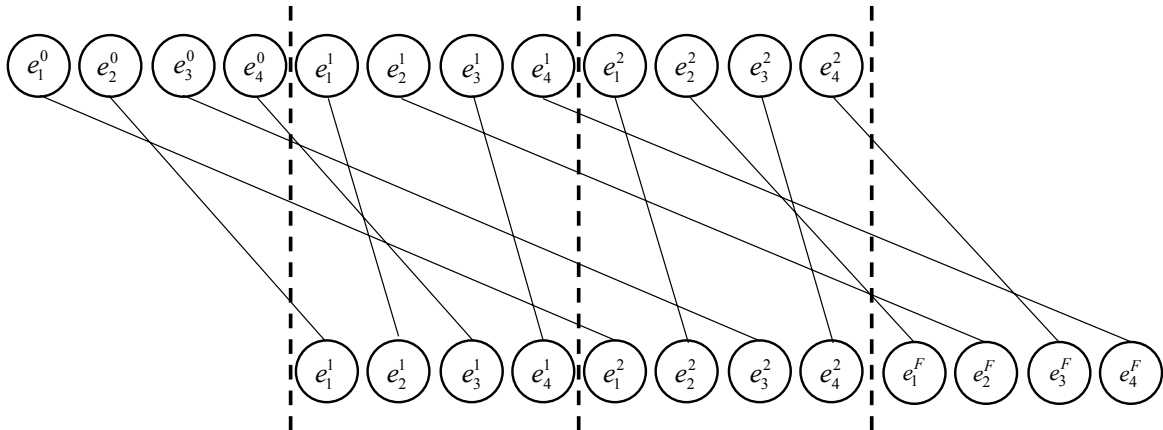


Figure 8. Illustration of a solution.

Bs_i^k and the time to travel from the block to the QC l of the j^{th} succeeding discharging container of QC l . Case (4) illustrates the time for a YT trip to the block Bl_j^l and the travel time from the block to the QC l of the j^{th} loading container.

Specifically, if e_i^k and e_j^l are related to dual cycle and loading operations, respectively—as shown in case (g) of Table 2—the starting event for evaluating c_{ki}^{lj} is then the pickup of a loading container of the i^{th} dual cycle operation of QC k by QC k . c_{ki}^{lj} includes the transfer time for a loading and a discharging container of the i^{th} dual cycle operation of QC k , the travel time from the apron to the location of the next container (the j^{th} operation of QC l) in the yard, the release time of the container by YC, and the travel time of the YT to QC l . Note that the storage location of the discharging container of the i^{th} dual cycle operation of QC k is assigned to the same storage block as that of the loading container for the j^{th} operation of QC l .

Table 2. Various cases for modifications of parameters.

Cases	Operation sequence	Modifications
(a)	Loading → Loading	(1)
(b)	Loading → Discharging	(2)
(c)	Loading → Dual cycle (LU)	(1)
(d)	Discharging → Loading	(4)
(e)	Discharging → Discharging	(3)
(f)	Discharging → Dual cycle (LU)	(4)
(g)	Dual cycle (LU) → Loading	(4)
(h)	Dual cycle (LU) → Discharging	(3)
(i)	Dual cycle (LU) → Dual cycle (LU)	(4)

Let α be the travel cost per unit time of an YT, and β be the penalty cost per unit time for the delay in the completion time. It is assumed that $\alpha \ll \beta$. Also, let m_O and m_F equal to $|V|$. Then, the dispatching problem is formulated as follows (Kim and Bae, 2004).

Minimize

$$\alpha \sum_{k \in K'} \sum_{i=1}^{m_k} \sum_{l \in K''} \sum_{j=1}^{m_l} t_{ki}^{lj} x_{ki}^{lj} + \beta \sum_{k \in K} (y_{m_k}^k - s_{m_k}^k)^+ \quad (1)$$

Subject to

$$\sum_{l \in K''} \sum_{j=1}^{m_l} x_{ki}^{lj} = 1, \quad \forall k \in K', i = 1, \dots, m_k \quad (2)$$

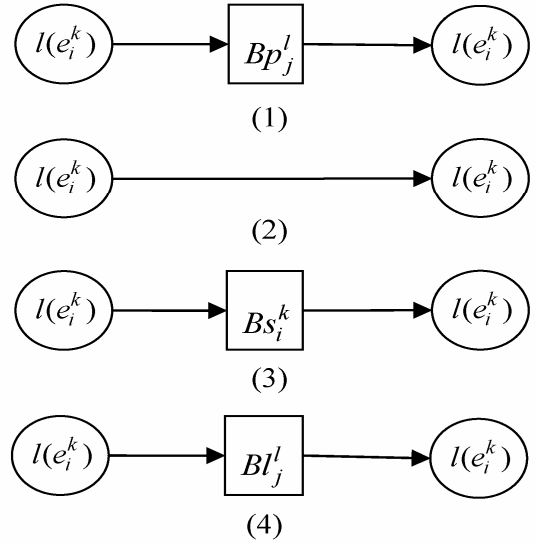
$$\sum_{k \in K'} \sum_{i=1}^{m_k} x_{ki}^{lj} = 1, \quad \forall l \in K'', j = 1, \dots, m_l \quad (3)$$

$$y_j^l - (y_i^k + c_{ki}^{lj}) \geq M(x_{ki}^{lj} - 1), \quad (4)$$

$$\forall k \in K', l \in K, i = 1, \dots, m_k, j = 1, \dots, m_l$$

$$y_{i+1}^k - y_i^k \geq s_{i+1}^k - s_i^k, \quad (5)$$

$$\forall k \in K, i = 1, \dots, m_k - 1$$



$l(e_i^k)$ = The location where event e_i^k occurs.

Figure 9. Modifications of t_{ki}^{lj} s and c_{ki}^{lj} s.

$$y_i^k \geq s_i^k, \quad \forall k \in K', i = 1, \dots, m_k \quad (6)$$

$$x_{ki}^{lj} = 0 \text{ or } 1,$$

$$\forall k \in K', l \in K'', i = 1, \dots, m_k, j = 1, \dots, m_l \quad (7)$$

where, $s_i^O = 0$ for $\forall i \in V$.

Figure 10 illustrates a solution to the example in Table 1. The solid arcs show the assignments for YT 01, while the dotted arcs are for YT 02. Note that e_1^1 and e_4^1 are dual cycle operations. By the first solid arc, node e_1^O in the supply source is connected to node e_1^1 in the demand source, which is a dual cycle operation. YT 01 starts the empty trip to the pickup position of a loading container for the 1st task of QC 01 and performs the loaded trip to QC 01. At quay, QC 01 picks up and transfers this loading container to the ship; it then delivers a discharging container for the 1st task of QC 01 to YT 01. After receiving a discharging container, YT 01 goes to a block that is assigned as follows. To force the dual cycle operations, the assignment of a storage location to a discharging container is made when the arc linking to the next task is created. If the next task is a loading task, then the storage location of the discharging container is the block for the succeeding loading container; otherwise, it is assigned to the block on the shortest route. Therefore, the storage location of a discharging container of the 1st task is the same block location with a loading container of the 4th task (as a dual cycle operation in Figure 10). Next, node e_1^1 in the supply source is connected to node e_4^1 in the demand source, which implies that after YT 01 completes the 1st task of QC 1, it performs the containers related to the 4th task of QC 01. Finally, node e_4^1 in the supply source is connected to node e_1^F in the demand source; this can be

interpreted as that YT 01 stopping after it completes all assigned tasks. In a similar way, the solution for YT 02 can be interpreted as follows. YT 02 delivers a discharging container for the 2nd task of QC 01, then a loading container for the 3rd task of QC 01; it then stops. Note that the storage location of a discharging container for the 2nd task is the same block location as a loading container for the 3rd task (as a dual cycle operation in Figure 10).

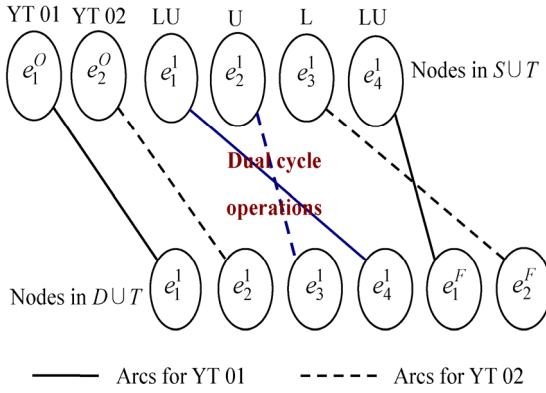


Figure 10. Graphical representation of a solution of x_{ki}^j s.

In this section, a heuristic algorithm for solving the integrated YT dispatching and storage location problem is summarized as follows. In the initial stage, y_i^k are assumed to be equal to s_i^k ; events (tasks) are then sequenced in increasing order of y_i^k . The heuristic algorithm attempts to assign the first QC task to a vehicle, then the first two events to vehicles, then the first three events to vehicles, and so on. In a heuristic procedure with precedence constraint, if a QC event exists which cannot be assigned to any vehicle, then the QC event times must be delayed until every candidate QC event is assigned to a vehicle. This process is continued until all QC events are assigned to vehicles. During the assignment of QC tasks to vehicles, the storage location of the discharging tasks is determined as follows: if the next task of the vehicle assigned to the discharging task is a loading task, then the storage location is allocated to the block nearest to that of the loading task; otherwise, the storage block for the discharging container is determined to minimize the travel distance of the YT between the preceding and succeeding QC. Finally, the assignment problem is solved to minimize the total travel time.

Denote the j^{th} event in the sequence as event j , the event time of event i as y_i for QC operations, c_{ij} as the time required for a YT to be ready for event j after it experiences event i -this corresponds to $c_{ki}^j - t_{ij}$ as the pure travel time from the location of event i to the location of event j , and x_{ij} as the decision variable for the assignment of event i to event j . Let T_ξ be a subset of T , which includes only the first ξ events in the sequence. Then, the constraint subset ξ of precedence constraints

can be written as follows:

Constraint subset ξ

$$\sum_{i \in S \cup T_\xi} x_{ij} = 1, \forall j \in D \cup T_\xi \quad (8)$$

$$\sum_{j \in D \cup T_\xi} x_{ij} = 1, \forall i \in S \cup T_\xi \quad (9)$$

$$y_j - (y_i + c_{ij}) \geq M(x_{ij} - 1), \forall i \in S \cup T_\xi, j \in T_\xi \quad (10)$$

$$x_{ij} = 0 \text{ or } 1, \forall i \in S \cup T_\xi, j \in D \cup T_\xi \quad (11)$$

The detailed heuristic algorithm can then be described as follows.

Step 0. Initializing. Set $y_i^k = s_i^k$ and $y_i^O = 0$ for all YTs. Set $\xi = 0$.

Step 1. Next Task. $\xi = \xi + 1$. If $\xi > m$ (m = the total number of tasks in sequence T), then go to *Step 4*. Otherwise, sequence the events in increasing order of y_i^k and go to *Step 2*.

Step 2. Feasibility Checking. Check the existence of a feasible solution to constraints subset ξ , Eq. (8)~Eq. (11). If there is a feasible solution, then go to *Step 1*. Otherwise, go to *Step 3*.

Step 3. Delaying Event Time.

$$\text{Let } \pi_{i\xi} = \min_{i \in S \cup T_{\xi-1}} [\max\{c_{i\xi} - (y_\xi - y_i), 0\}].$$

And denote the event time of event (ξ) be y_ξ^y .

Then, update $y_j^y = y_j^y + \pi_{i\xi}$, for $j \geq \lambda$. Go to *Step 2*.

Step 4. Task Assignment. Evaluate t_{ij} . Solve the assignment problem with the objective of minimizing the total travel time subject to constraint subset $|T|$. Stop.

Feasibility Check. In this step, for given values of y_i^k , the feasibility can be checked by solving a maximum cardinality matching problem in the bipartite graph (Evans and Minieka, 1992). When the maximum cardinality is the same as $|S \cup T|$, the constraint subset ξ of Eq. (8)~Eq. (11) has a feasible solution. In this study, the storage location of the loading tasks was given while the discharging tasks were undetermined. Thus, the dispatching problem is simultaneously solved with the storage location problem for discharging tasks. While solving the matching problem, the storage location of discharging tasks is determined as follows. If the location is matched with a loading task, then the storage location of the discharging task is allocated to that of the loading task. Otherwise, the storage block for discharging container is determined to minimize the travel distance for the YT between the preceding and succeeding QC.

Delaying Event Time. To satisfy constraint subset Eq. (8)~Eq. (11), one or more additional x_{ij} must be al-

lowed to become 1 by relaxing constraint Eq. (10). In other words, the event time for event ξ is delayed so that at least one $x_{i\xi}$, for $i < \xi$, becomes 1, which is denoted as $\pi_{i\xi}$. The process is repeated until the current constraint subset becomes feasible.

To evaluate the proposed heuristic algorithm, the optimal solutions obtained from the model formulation were compared with those from a heuristic algorithm. A number of problems were generated by changing the number of QCs within 1 to 2, the number of YTs within 1 to 3, and the number of operation tasks for each QC within 4 to 30. Model formulations were solved using ILOG CPLEX[®] 10.0.0. The ratio of the objective value of the solution yielded by the heuristic algorithm to that for the solution yielded by ILOG CPLEX[®] ranged from 1.000 to 1.223; on average, it was 1.018. The ratio of the computational time for the heuristic algorithm to that for ILOG CPLEX[®] ranged from 0.00001 to 0.0465; on average, it was 0.0077. The computational time was found to be sensitive to the total number of operation tasks. Therefore, it can be concluded that the heuristic algorithm showed a good performance.

4. SIMULATION EXPERIMENTS

4.1 Simulation Modeling

The operation of the container terminal was modeled in detail. When a ship arrives, it is assigned to a berth if there is an available berth for the ship to enter. Otherwise, the ship must wait until a berth becomes available. When the ship enters a berth, a number of QCs are assigned to the ship. A loading and discharging sequence is then generated for each QC. By referring to the specified sequence, QCs can start to unload and load containers.

A number of YTs are also assigned to QCs to transport containers between the quay and yard. All QCs share all vehicles together; that is, a pooling strategy is used to dispatch vehicles. However, in dynamic situations, because the operation times of container handling equipment are uncertain, it is meaningless to schedule all the delivery tasks in the generated sequences. Therefore, it is sufficient to look ahead to a small number of future tasks but not all for dispatching. This means that only the most imminent tasks are considered in the dispatching algorithm of this study.

The heuristic procedure is triggered whenever a YT becomes idle or completes its assigned task. For dispatching decisions, information on delivery tasks and YTs is necessary. A control system was developed in this study to monitor the progress of ship operations. It is used to estimate and update the event times of ship operations whenever a YT becomes idle or completes a delivery task. Following the result of the heuristic algo-

gorithm, the dispatching decision is fixed for only the newly available vehicle, which immediately starts its trip to the pickup position of the newly assigned task.

The operation scenarios modeled in the simulation program are described below. At the quay side, QCs unload or load containers only when YTs are available under QCs. When a YT arrives at a designated QC, it must wait for either pickup or drop-off of a container by a QC. At the yard side, a YT first arrives at a transfer point (TP) at the side of the block. The YT waits for a YC to pick up an inbound container or release an outbound container. When a YT finishes transporting a container, it returns to the parking area to await the next assignment. In the heuristic algorithm, to reduce the YT empty trip time, when a YT arrives at a certain storage block to unload a container, it should be assigned to a next loading container from this block. This means that the heuristic algorithm attempts to perform dual cycle operations for YTs.

For loading, a YC picks up a loading container from a specified slot in the yard. For discharging, when a YC receives an inbound container from a vehicle, it stacks the container in a slot in the yard. The loading and discharging operations continue until all containers are transferred between the ship and storage yard. When all operations are completed, the ship leaves the berth.

In this section, a simulation model is introduced as follows. The layout of a hypothetical container terminal is shown in Figure 1. The wharf has one berth and three QCs. The yard consists of six storage blocks; two YCs of the same size are deployed to each. Each block has TPs on which YTs can wait for transferring containers. The travel times of YTs between pickup and drop-off points are assumed to be constant, and traffic congestion for YTs in driving lanes is not considered. About 150 containers were transferred by each QC during a simulation run.

Two different dispatching methods—using pre-assigned storage locations for the discharging containers and using postponed assignments in which storage locations of the discharging containers were determined during the dispatching process—were compared with each other in simulation experiments.

To test the effect of the three different types of QC operations, three data sets, each of which corresponded to a different type of QC operation, were prepared. The first type is shown in Figure 3(a), where a QC begins the discharging operation only after all the loading operations assigned to the QC are completed; this is denoted as “loading after discharging” (LAD). The second is illustrated in Figure 3(b) where a QC can load into a bay and discharge from the next one. In this case, discharging and loading operations by two QCs can be combined to construct a YT dual cycle operation; this is called “combining between QCs” (CBQ). The third type shown in Figure 3(c), where a QC can perform dual cycle operations; this is called the “complete dual cycle” (CDC).

The number of YTs in the system was set to be 12, 15, 18, and 21. When the dispatching heuristic algorithm is triggered, the number of future tasks included in the tasks to be dispatched was set at 6 for each QC. The number of replications of the simulation run for each combination of levels of factors was set to 10. The performance measures considered in the simulation experiments were the total QC delay time, the total YT travel time, and the total empty YT travel time.

4.2 Simulation Results and Analyses

Comparison between two different methods with different data sets

Various performance measures were compared with each other for two storage location strategies of pre-assignment and postponed assignments of storage locations. In addition, the effects of three different types of QC operations on the performance measures were tested.

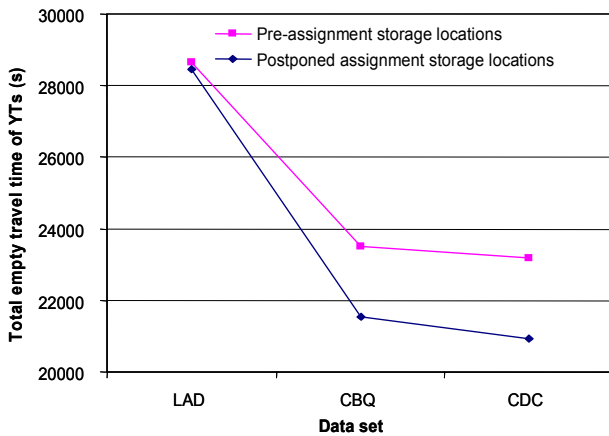


Figure 11. Comparisons between two methods in the total YT empty trip time with different types of QC operations.

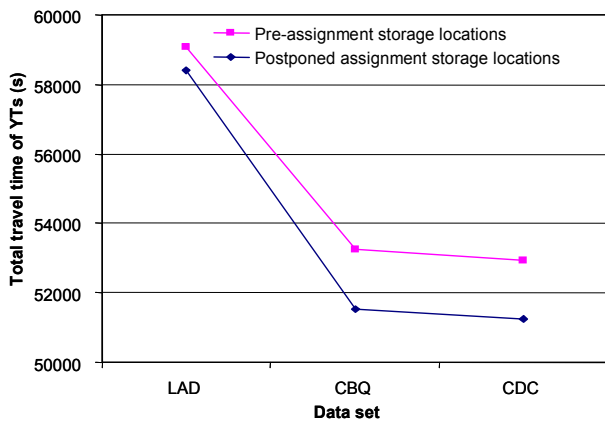


Figure 12. Comparisons between two methods in the total YT travel time with different types of QC operations.

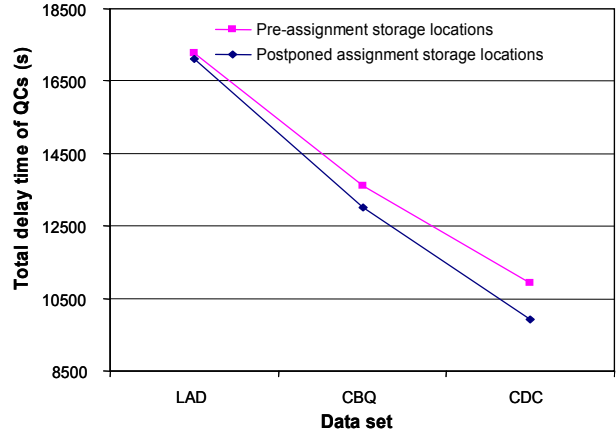


Figure 13. Comparisons between two methods in the total QC delay time with different types of QC operations.

Figure 11 shows the effects of different combinations of QC operation types and storage location strategy on the total YT empty trip time. The total empty trip time for YTs in both methods was significantly reduced when the operation type of QCs changed from LAD to CBQ and from CBQ to CDC. In addition, the total YT empty trip time was reduced by far more when the strategy of “postponed assignment of storage locations” was used rather than the strategy of the pre-assignment storage location.

In other words, the dispatching heuristics with postponed assignment of storage locations was better than that with the pre-assignment of storage locations with regards to the total YT empty trip time. The improvements between the two methods in the total YT empty trip time for LAD, CBQ, and CDC were 0.77%, 8.39%, and 9.67%, respectively. Similarly, the effects of different types of QC operation on the total YT travel time of the two methods are shown in Figure 12. The improvements between the two methods in the total YT travel time for LAD, CBQ, and CDC were 1.12%, 3.18%, and 3.23%, respectively.

Figure 13 illustrates the effects of different data sets on the QC delay time for the two methods. The total QC delay time of both methods was also reduced as the number of dual cycle operations increased. However, the reductions for the method with postponed assignment of storage locations was significant when the number of dual cycle operations increased in data sets. The improvements between the two methods in the total QC delay time for LAD, CBQ, and CDC were 0.88%, 4.43%, and 9.30%, respectively.

Effects of the change of the number of YTs on two different methods

Figures 14 and 15 shows the changes in the total YT empty trip time and total travel time for the two methods. As shown in Figure 14, the total YT empty trip

time for both methods was reduced as the number of YTs increased. However, the dispatching heuristic with postponed assignment of storage locations significantly outperformed pre-assigning storage locations with regards to total YT empty trip time for various numbers of YTs. For example, the ratio of the total YT empty trip time for the dispatching heuristic with postponed assignment of storage locations to the time with pre-assigned storage locations was 90.33% when the number of YTs was four.

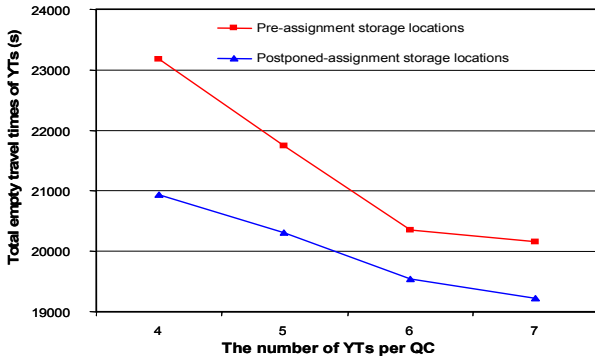


Figure 14. Effects of the number of YTs on the total YT empty trip time of the two methods.

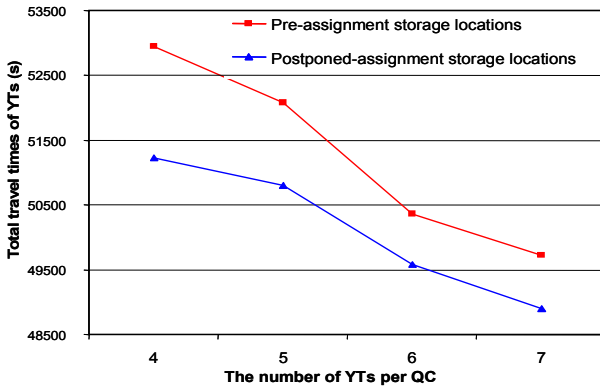


Figure 15. Effects of the number of YTs on the total YT travel time of the two methods.

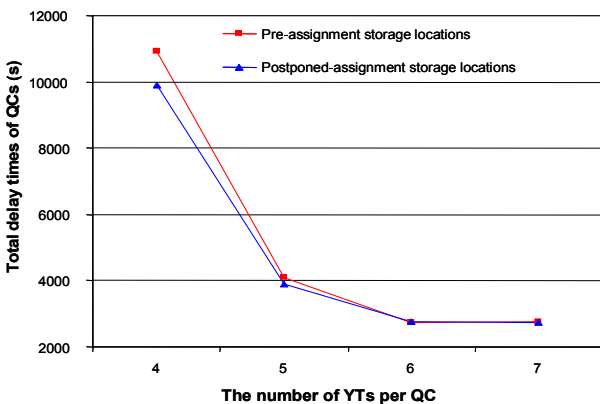


Figure 16. Effects of the number of YTs on the total QC delay time of the two methods.

Similarly, the changes in the total YT travel time of the two methods are shown in Figure 15. When the number of YTs was four, the ratio of the total YT travel time for the dispatching heuristic with postponed assignments of storage locations to the time for pre-assignment of storage locations was 96.77%.

Figure 16 illustrates the changes in the total QC delay time of the two methods for various values of the number of YTs per QC. Obviously, the total QC delay time of the two methods rapidly reduced as the number of YTs increased. When the number of YTs exceeded a certain level (six YTs, as indicated in Figure 16), the reductions were negligible in both cases; this was supported by a statistical test with a significance level of 5%. This means that six YTs were sufficient for supporting the QC operations. When comparing the two methods in terms of the total QC delay time, it can be concluded that the dispatching heuristic with postponed assignment of storage locations is better than that with pre-assigned storage locations. For example, when the number of YTs was four, the ratio of the total QC delay time with the dispatching heuristic with postponed assignment of storage locations to that time under that with pre-assignment of storage locations was 90.70% or the improvement between methods was 9.30%. Furthermore, the differences between two methods were reduced as the number of YTs increased; in other words, if the number of YTs was large enough for transferring containers from QC operations, there was no effect of the storage location method on the total QC delay time.

Table 3 illustrates the average percentage of different YT operation types for both dispatching methods with different data sets. The results show that the average percentage of type (a) rapidly reduced as the number of dual cycle operations by QC (as shown in data sets LAD, CBQ, and CDC) increased for both methods. The average percentage of the other types increased as the number of QC dual cycle operations increased for both two methods. Figure 17 shows the improvement in percentage of different YT operation types for the dispatching method with the postponed assignment of storage locations; CBQ and CDC data sets were compared with of the LAD data set. For example, the improvements between CDC and LAD in percentage for the YT operation types (a), (b), (c), (d), and (e) were -39.97%, 19.29%, 5.66%, 11.48%, and 3.41%, respectively.

Figures 18(a) and (b) show the trend when converting YT operations from type (a) to types (b) and (c) and from type (d) to (e) for the two dispatching methods, respectively. This results from that the postponed assignment changes the YT operations from types (a) and (b) to (c) or from (d) to (e).

Figure 19 illustrates the average empty trip times per container of different types of YT operations with the dispatching method using postponed assignments of storage locations for different data sets. The results show that the average empty trip time per container of YT operation type (a), as shown in Figure 5(a) (single cycle operations)

was the worst in comparison with the other types of YT operations, which were dual cycle operations. For type (b), shown in Figure 5(b), there were empty trips between the QCs and blocks. Types (c) and (d) Figures 5(c) and (d) had empty trips between the QCs and between the blocks, respectively. For type (e) Figure 5(e), no empty trips were required.

Table 3. Average percentage of different types of YT operations with different data sets (%).

Dispatching method with	Data sets	Types of operations by YT				
		Type (a)	Type (b)	Type (c)	Type (d)	Type (e)
Postponed assignment of storage locations	LAD	95.84	3.08	0.88	1.73	0.85
	CBQ	65.12	19.07	6.33	9.05	3.54
	CDC	55.88	22.37	6.54	13.21	4.26
Pre-assignment of storage locations	LAD	95.68	2.99	0.99	2.28	0.52
	CBQ	63.90	22.61	4.39	8.68	2.80
	CDC	57.20	21.79	5.69	14.00	3.67

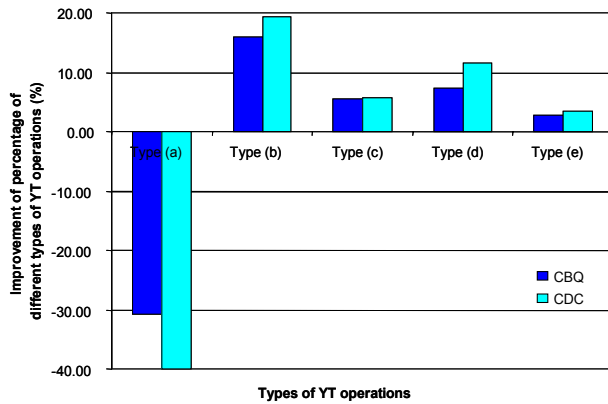
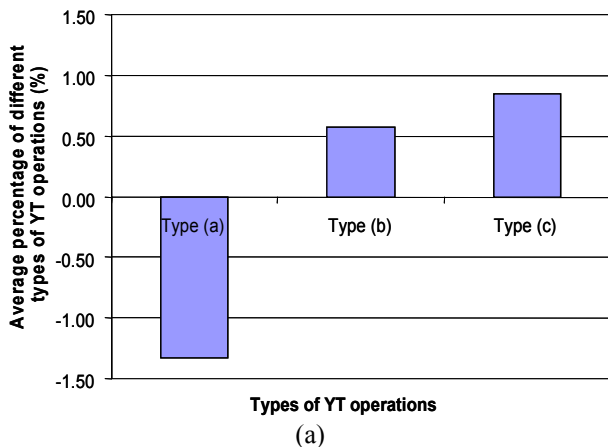


Figure 17. Changes in the percentage of different types of YT operations for CBQ and CDC compared with those of LAD.



(a)

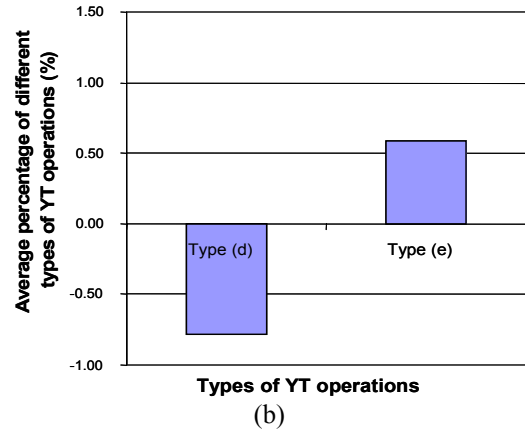


Figure 18. Changes in percentage of operation type of YTs from type (a) to types (b) and (c); and type (b) from type (d) to type (e) by postponing storage assignments.

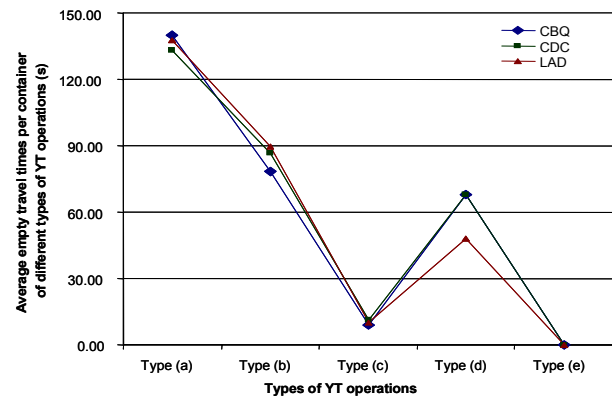


Figure 19. Average empty trip times per container of different types of YT operations with different data sets.

5. CONCLUSIONS

This paper addressed the YT dispatching problem considering storage locations in port container terminals. Dual cycle operations of quay cranes, yard trucks, and yard cranes were discussed. A modified mixed integer programming model and a heuristic algorithm were proposed to solve the problem.

A number of problems were generated to evaluate the optimal solutions from mathematical formulations compared to the heuristic algorithm. The results showed that the ratio of the objective values in the heuristic algorithm to the optimal objective values in the ILOG CPLEX[®] was between 1.000 and 1.223 and had an average value of 1.018. The ratio of the computational time in the heuristic algorithm to that of ILOG CPLEX[®] ranged from 0.00001 to 0.0465 and had an average value of 0.0077.

By performing simulation experiments, two dis-

patching methods with pre-assignments and postponed assignments of storage locations were compared with each other and evaluated for three different types of QC operations with different numbers of dual cycle operations. From the results of the experiments, it could be concluded that the YT dispatching heuristic algorithm with postponed assignment of storage locations had a better performance than that with pre-assigned storage locations. In addition, it was shown that the operation type of the quay cranes had a significant impact on the efficiency of the yard truck operations.

This study addressed only the dispatching of vehicles (YTs); however, the integrated scheduling of itineraries for vehicles, operations of yard cranes, and operations of QCs may be addressed in a further research. In addition, the storage locations of loading containers should be considered when dispatching YTs.

ACKNOWLEDGEMENTS

This study was supported by the Korean Ministry of Education & Human Resources Development through the Research Center of Logistics Information Technologies (LIT).

REFERENCES

- Bish, E. K., Leong, T. Y., Li, C. L., Ng, J. W. C., and David, S. L. (2001), Analysis of a new vehicle scheduling and location problem, *Naval Research Logistics*, **48**, 363-385.
- Briskorn, D., Drexl, A., and Hartmann, S. (2006), Inventory-based dispatching of automated guided vehicles on container terminals, *OR Spectrum*, **28**, 611-630.
- Evans, J. R. and Minieka, E. (1992), *Optimization Algorithms for Networks and Graphs*, New York, Marcel Dekker.
- Goodchild, A. V. and Daganzo, C. F. (2006), Double-cycling strategies for container ships and their effect on ship loading and discharging operations. *Transportation Science*, **40**, 473-483.
- Goodchild, A. V. and Daganzo, C. F. (2007), Crane double cycling in container ports: Planning methods and evaluation. *Transportation Research Part B*, **41**, 875-891.
- Grunow, M., Günther, H. O., and Lehmann, M. (2006), Strategies for dispatching AGVs at automated seaport container terminals, *OR Spectrum*, **28**, 587-610.
- Kim, K. H. and Bae, J. W. (2004), A look-ahead dispatching method for automated guided vehicles in automated port container terminals, *Transportation Science*, **38**, 224-234.
- Lee, D. H., Cao, J. X., Shi, Q., and Chen, J. H. (2009), A heuristic algorithm for yard truck scheduling and storage allocation problems, *Transportation Research Part E*, **45**, 810-820.
- Lee, S. (2007), Locating idle vehicles in tandem-loop automated guided vehicle systems to minimize the maximum response time, *Industrial Engineering & Management Systems*, **6**, 125-135.
- Nguyen, V. D. and Kim, K. H. (2009), A dispatching method for automated lifting vehicles in automated port container terminals, *Computers and Industrial Engineering*, **56**, 1002-1020.
- Zhang, H. P. and Kim, K. H. (2009), Maximizing the number of dual-cycle operations of quay cranes in container terminals, *Computers and Industrial Engineering*, **56**, 979-992.