ILS2016 Information Systems Logistics and Supply Chain
6th International Conference

Synchronizing Quay Cranes and Automated Vehicles for Ship Operations by Using a Bidding Procedure

Ivan Kristianto Singgih, Hans-Otto Guenther, and Kap Hwan Kim

Department of Industrial Engineering
Pusan National University, Jangjeon-dong, Geumjeong-gu, Busan, 609-735, Korea
{ivanksinggih@pusan.ac.kr, guenther@pusan.ac.kr, kapkim@pusan.ac.kr}

Abstract. This paper proposes a method for dispatching automated vehicles in container terminals. A bidding concept is developed which particularly considers the urgency and required travel time of a transportation task. The objectives are to minimize the task delay and the required travel time to perform all tasks. Essential part of the bidding concept is its suitability for real-time application, i.e. the dispatching results are continuously being updated based on the current state of the terminal. The proposed bidding procedure is intended for practical application in an automated container terminal. Numerical experiments show that the bidding algorithm performs well compared to a greedy heuristic, which was adapted from a standard procedure known from the literature.

Keywords: dispatching, bidding, scheduling, automated container terminals.

1. Introduction

In an automated container terminal, vehicles travel between the quay side and the storage area to transport containers which are discharged from or to be loaded onto a berthed vessel. Related to the use of automated vehicles, various issues are studied in literature including flow path design, vehicle dispatching, routing, and traffic management together with the prevention and resolution of collisions and deadlocks. This paper deals with a dynamic vehicle dispatching problem in which dispatching is triggered when a new transportation task is released by the overall terminal control system or when a vehicle becomes idle. The dispatching results can be updated considering the current system state of the terminal such as quay crane (QC) task times or empty travel times by vehicles to execute the tasks.

Many researches about dynamic dispatching of vehicles have been conducted. In [1] a dynamic task-assignment algorithm for truck-load transportation is proposed. A real-time vehicle-dispatching problem, in which vehicles can be diverted from their current destination in response to a new customer request, is addressed in [2]. Procedures to optimally allocate vehicles to delivery tasks are proposed in [3], [4], and [5]. Specifically, the dispatching problem for trucks has been intensively studied in [6], [7], and [8]. In [9] and [10] an auction-based algorithm for solving the assignment problem in a distributed manner is developed. A number of studies on automated guided vehicle (AGV) dispatching are reported in [11], [12], [13], [14], [15], [16], [17], [18], [19], and [20]. Several of these papers consider the AGV dispatching problem in flexible manufacturing systems, while [13], [14], [15], [16], [17] and [20] specifically address the AGV dispatching problem arising in seaport container terminals. Collaborative strategies and simulation-optimization based methods for vehicle dispatching are studied in [21] and [22].

From the published academic, literature we consider the inventory-based dispatching algorithm proposed in [16] as most suitable for real-time application in an automated container terminal. In their publication, the authors develop a greedy heuristic which is modified in our study and used as a benchmark procedure in the numerical experimentation in (see Section 5). In [16] it is shown that, in a highly stochastic application environment, priority rule based dispatching algorithms outperform static optimization methods for a number of different performance criteria. This result is mainly due to the capability of the priority rule based concept to effectively deal with the highly stochastic and dynamic changes of the system state caused by factors like QC operation times, traffic congestion etc. Therefore, in this paper, we propose a bidding type procedure which allows quick update of the vehicle-task assignments in response to the frequent changes in the real state of the container terminal.

The remainder of this paper is organized as follows. In the next section we give a brief overlook of the control of ship and vehicle operations in a container terminal. In Section 3, a mathematical model formulation of the dispatching problem at hand is presented. Section 4 presents the proposed bidding algorithm. This is followed by some preliminary numerical experimentation. Finally, conclusions and an outlook of future research are given.

2. Ship Operation and Operation Control for Vehicles

Before a ship arrives at a container terminal, all information regarding inbound and outbound containers is sent to the terminal by the shipping agent. Based on this information, the sequence of discharging and loading operations of individual containers is determined. When the ship actually arrives, ship operations are usually performed on the basis of the discharging and loading sequence list [15].

In container terminals, vehicles are used to transport the containers between the quay side and the storage area. In order to avoid delays of the QCs, dispatching of vehicles is a critical task. To adapt to a changing environment, a dispatching decision must be made in real time whenever a relevant event occurs. The sequence of discharging and loading operations for each QC developed by ship operation planners is a major input for vehicle dispatching (see LIST A in Figure 1). Whenever a task in LIST A enters the planning horizon (planning time window) for dispatching, it will be moved to LIST B. The tasks in LIST B are candidates for assignment to vehicles. As soon as a vehicle commences its travel to pick up a container for a task, the task is removed from LIST B.

The dispatching algorithm is triggered when a vehicle becomes idle or when a new task enters LIST B. When a vehicle completes a delivery task, the vehicle reports the completion of the task to the vehicle supervisory system (VSS). The VSS will then trigger the dispatching algorithm for assigning a new task. For conducting a delivery task, the vehicle will travel to the pickup position of the assigned task. Otherwise, it will move to the parking area to await the next assignment.

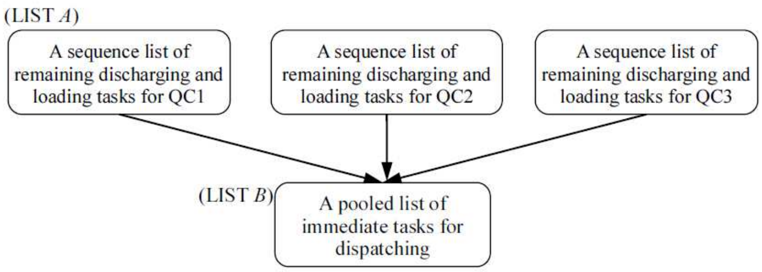

Figure 1: Lists of tasks used in for vehicle dispatching [15]

3. Mathematical Model for Dispatching

Ship operations are usually performed on the basis of the discharging and loading sequence list. The vehicle dispatching problem arising at one instance of the terminal operations can be formulated as a mixed-integer programming (MIP) model [14]. This kind of model assumes deterministic handling and travel times for the equipment. It is presented here as a formal description of the underlying static optimization problem. Below we first introduce the formulation of the dispatching problem and develop the heuristic bidding based on [14] in the subsequent section.

A loading operation cycle of a QC begins with the pickup of a container from a vehicle, while a discharging operation cycle ends with the release of a container onto a vehicle. For a QC operation to be performed without delay, a vehicle must be ready at a specified location beneath the associated QC before the transfer of a container commences. Let e_i^k be an event representing the moment at which a vehicle transfers the i^{th} container of QC k (the i^{th} operation of QC k). When the i^{th} operation of QC k is a loading operation, event e_i^k corresponds to the beginning of the pickup of a container from a vehicle. When the i^{th} operation of QC k is a discharging operation, it corresponds to the beginning of the release of a container onto a vehicle. The time of event e_i^k is denoted Y_i^k. A delay of an operation occurs when the corresponding vehicle does not arrive at the requested moment. The earliest possible event time, i.e. the time of the event with no delays of QC operations, is denoted by s_i^k.

Three types of events are faced by vehicles during a ship operation: the initial event, which represents the current state of each vehicle; the event when a vehicle begins to receive a container from a QC or when a vehicle begins to transfer a container to a QC; and the stopping event, when a vehicle completes all of its assigned tasks.

The notations related to ship operations are summarized as follows:

V	=	The set of vehicles.
K	=	The set of QCs.
e_v^O	=	The initial event of vehicle *v*, $v \in V$.
e_v^F	=	The stopping event of a vehicle *v*, $v \in V$. Note that, although the number of stopping events of vehicles is the same as the number of vehicles, stopping events with different subscripts do not need to be distinguished from each other.
e_i^k	=	The event that corresponds to the beginning of a pickup (or release) of a container from (onto) a vehicle for the task related to the *i*th operation of QC *k*. Assume that there exist m_k tasks for QC *k*.
T	=	The set of e_i^k for $i = 1,2,\cdots,m_k$ and $k \in K$.
$l(e_i^k)$	=	The location at which event e_i^k occurs. $l(e_v^O)$ represents the initial location of vehicle *v*. Here, $l(e_i^k)$ represents the position at which the *i*th container of QC *k* will be transferred. The location at which a vehicle completes its final delivery task is denoted $l(e_v^F)$.
t_{ki}^{lj}	=	The pure travel time from $l(e_i^k)$ to $l(e_j^l)$.
C_{ki}^{lj}	=	The time required for a vehicle to be ready for e_j^l after undergoing e_i^k, which is a random variable. For example, if both e_i^k and e_j^l are related to loading operations, then the starting moment (event) for evaluating C_{ki}^{lj} is the pickup of the *i*th container of QC *k* by QC *k*. Included in C_{ki}^{lj} are the travel time from the apron to the location of the next container (the *j*th container of QC *l*) in the marshalling yard, the release time of the container by a YC, and the travel time of the vehicle to QC *l*.

Let *S* and *D* be the sets of e_v^O and e_v^F, $v \in V$, respectively. A feasible dispatching decision is then a one-to-one assignment between all the events in $S \cup T$ and those in $D \cup T$. Let $K' = \{O\} \cup K$, $K'' = \{F\} \cup K$, and x_{ki}^{lj} be a decision variable that becomes 1 if e_i^k is assigned to e_j^l, for $k \in K'$ and $k \in K$. For $k, l \in K$, the assignment of e_i^k to e_j^l implies that the vehicle that has just delivered the *i*th container of QC *k* is scheduled to deliver the *j*th container of QC *l*.
Let α be the travel cost per unit time of a vehicle, and β be the penalty cost per unit time for a delay in the completion time. It is assumed that $\alpha << \beta$. Further, let m_O and m_F equal $|V|$. The dispatching problem can then be formulated as follows:

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

subject to

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

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

$$P\left\{C_{ki}^{lj} \le Y_j^l - Y_i^k\right\} - \theta \ge M\left(x_{ki}^{lj} - 1\right), \quad \text{for } \forall k \in K', l \in K,\ i = 1,\cdots,m_k, \text{ and } j = 1,\cdots,m_l \quad (4)$$

$$Y_v^O = 0, \quad \text{for } \forall v = 1,\cdots,|V| \quad (5)$$

$$Y_{i+1}^k - Y_i^k \ge s_{i+1}^k - s_i^k, \quad \text{for } \forall k \in K \text{ and } i = 1,\cdots,m_k - 1 \quad (6)$$

$$Y_i^k \ge s_i^k, \quad \text{for } \forall k \in K' \text{ and } i = 1,\cdots,m_k \quad (7)$$

$$x_{ki}^{lj} = 0 \text{ or } 1, \quad \text{for } \forall k \in K',\ l \in K'',\ i = 1,\cdots,m_k, \text{ and } j = 1,\cdots,m_l \quad (8)$$

Because $\alpha << \beta$, the sum of the delays of QC operations will be minimized first. With second priority, the total travel distance of the vehicles is minimized. Constraints (2) and (3) enforce the one-to-one assignment between all events in $S \cup T$ and those in $D \cup T$. Constraint (4) implies that two events that are served consecutively by the same vehicle must be set apart by at least the time required for the vehicle to travel and transfer a load between the two events. Constraint (6) implies that two events that are served by the same QC must be set apart by at least the time required for the QC to perform all the movements between the two events. Constraint (7) signifies that the actual event time is always greater than or equal to the earliest possible event time.
Problem (1)-(8) can be solved by systematically enumerating the values of Y. Initially, values of Y are set based on the required time between the event time of tasks of the same QC and updated using the event time delay procedure, which is presented later in Section 4. Suppose that a value of *Y* is given. Then, constraint (4) will restrict the feasible value of x_{ki}^{lj}. Because C_{ki}^{lj} is a random variable, by setting a_{ki}^{lj}=1 if $P\{y_j^l - (y_i^k + C_{ki}^{lj}) \geq 0\} \geq \theta$, i.e. $P\{C_{ki}^{lj} \leq y_j^l - y_i^k\} \geq \theta$; 0, otherwise. $P\{C_{ki}^{lj} \leq y_j^l - y_i^k\}$ is denoted as P_{ij}, which is the probability that vehicle *i* has enough time to reach task *j*, obtained by comparing the time difference between the current event time of task *j* and the event time of the task, which is being performed by vehicle *i*, with the required travel time between the end position of vehicle *i* after performing the current task and the start (pickup) position of task *j*. A reasonable value of the threshold θ is defined by the user. Using high θ values enables the user to deal with greater travel time uncertainty, but tends to require more iterations during the dispatching process because less dispatching links are created. Then the problem becomes:

Minimize $$\sum_{k \in K'} \sum_{i=1}^{m_k} \sum_{l \in K''} \sum_{j=1}^{m_l} t_{ki}^{lj} \cdot x_{ki}^{lj} \quad \textbf{(9)}$$

subject to

$$\sum_{l \in K''} \sum_{j=1}^{m_l} a_{ki}^{lj} x_{ki}^{lj} = 1, \quad \text{for } \forall k \in K' \text{ and } i = 1, \cdots, m_k \quad \textbf{(10)}$$

$$\sum_{k \in K'} \sum_{i=1}^{m_k} a_{ki}^{lj} x_{ki}^{lj} = 1, \quad \text{for } \forall l \in K'' \text{ and } j = 1, \cdots, m_l \quad \textbf{(11)}$$

$$x_{ki}^{lj} = 0 \text{ or } 1, \quad \text{for } \forall k \in K',\ l \in K'',\ i = 1, \cdots, m_k, \text{ and } j = 1, \cdots, m_l \quad \textbf{(12)}$$

4 Heuristic Procedure

4.1 Main Procedure of the Heuristic Algorithm

Let $c_{ki}^{lj} = E(C_{ki}^{lj})$. The overall procedure including ASSIGN-TASK-TO-A-NEW-VEHICLE (ATNV) or ASSIGN-VEHICLE-TO-A-TASK (AVNT) can be described as follows:

Step 1: *Update arcs* to (from) the new task (vehicle). Note that arcs from node *i* to *j* are linked in the graph only if $P_{ij} \geq \theta$.

Step 2. *Perform bidding based dispatching method (BDM).* Perform *BDM*, which is an algorithm for solving the assignment problem.

Step 3. *Delaying Event Time.* Let the selected unassigned task be task ξ. Delay the event time of the task in order to allow arc creation between the task and any unconnected vehicles. Denote y_λ^γ as the event time of event (*ξ*). Then, update $y_j^\gamma = y_j^\gamma + \pi_{i^* \xi}$, for $j \geq \lambda$. Go to Step 1.

Delaying Event Time: To satisfy the revised constraint subset, one or more additional x_{ij} must be allowed to become 1 by relaxing constraint (12). In other words, the time for event *ξ* is delayed so that at least one $x_{i\xi}$, for $i < \xi$, becomes 1, denoted $\pi_{i\xi}$. The process is repeated until the current constraint subset becomes feasible.

4.2 Bidding Based Dispatching Method

The Bidding Based Dispatching Method (BDM) assumes that the price of each delivery is determined through a bidding process. During the bidding process, each vehicle selects the delivery task that maximizes its own margin which is the price of the task minus the cost of performing the task, while each task chooses the vehicle that requests the least compensation which is the transportation cost required by the vehicle for performing the task plus the minimum margin requested by the vehicle.

For a given assignment of vehicles to delivery tasks, a set of prices and margins is said to be at "equilibrium" if a vehicle cannot increase its margin by changing its currently assigned task and a delivery task cannot decrease the compensation by changing its currently assigned vehicle [13].

In the dynamic situation assumed in this study, all loaded or idle vehicles are candidates for dispatching. A vehicle may have one task being performed and one assigned task. Tasks in list B (see Figure 1) are considered to be candidates for dispatching. A dispatching decision process is initiated whenever a vehicle is loaded or a new delivery order enters list B. When a vehicle finishes delivery, the next task is fixed from the updated dispatching result and the ASSIGN-TASK-TO-A-NEW-VEHICLE procedure is initiated to secure the next delivery task. In case a new delivery order enters list B, the ASSIGN-VEHICLE-TO-A-NEW-TASK procedure is initiated. Once either of the two procedures is initiated, a feasible assignment and prices almost at equilibrium are obtained [13]. A dispatching decision on a vehicle or a delivery task is fixed when a vehicle starts its travel for conducting the assigned delivery task, in which moment both the vehicle and the task will be excluded from the candidate list for dispatching.

During the ASSIGN-TASK-TO-A-NEW-VEHICLE procedure, prices of tasks and margins of vehicles decrease. However, during the ASSIGN-VEHICLE-TO-A-NEW-TASK procedure, prices of tasks and margins of vehicles increase. The prices of tasks are limited by a pre-specified upper bound of the price from the above, while, above zero, margins of vehicles change.

In the following, two procedures (ASSIGN-TASK-TO-A-NEW-VEHICLE, ASSIGN-VEHICLE-TO-A-NEW-TASK) are introduced for matching vehicles with delivery tasks. Note that the number of available vehicles (n) may not be the same as the number of available delivery tasks (m). Assume that $m>n$ without loss of generality. Then, even after an assignment is determined, one or more tasks may not be assigned ("unassigned and inactive": UI) to a vehicle. Both the number of vehicles and the number of tasks in the "assigned (A)" state are n. However, when a vehicle (task) becomes a new candidate for an assignment, it is "unassigned" but has a potential to be assigned. The vehicle (task) is said to be "unassigned but activated (UA)." Also, during the assignment procedure, a less competitive vehicle (task) may have its assigned task (vehicle) taken away by another more competitive vehicle (task). Then, the former vehicle (task) becomes "unassigned but activated (UA)," while the latter vehicle (task) becomes "assigned (A)."

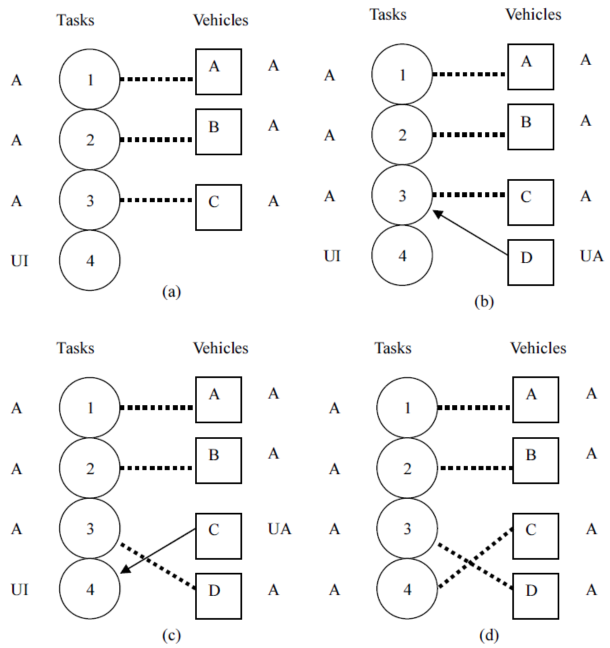

Figure 2: An illustration of the ASSIGN-TASK-TO-A-NEW-VEHICLE [13]

The bidding process (ASSIGN-TASK-TO-A-NEW-VEHICLE) for the case of the vehicle initiation (when a new vehicle becomes idle) is illustrated in Figure 2. Before the new vehicle becomes idle, three vehicles are assigned to three tasks. Thus, they are in state "A" except one task that is not assigned to any vehicle and so is in state "UI" (see Figure 2-(a)). When a vehicle (vehicle D) becomes idle, its initial state is "UA", and it seeks a delivery task to perform (see Figure 2-(b)). Among waiting tasks, vehicle D selects a task (task 3) giving the maximum margin at the current price. Then, vehicle D submits a bid with a price lower than the current price of task 3. Then, task 3 accepts the bid because the suggested price is lower than the current price, and informs the cancellation of assignment to the vehicle (vehicle C) that task 3 was previously assigned to. The state of vehicle D is changed from "UA" to "A" and the state of vehicle C is changed from "A" to "UA" (see Figure 2-(c)). Then, vehicle C whose state became "UA" searches for the task that gives the highest margin at the current price. In Figures 2-(c), it is task 4. Because task 4 was in state of "UI," no vehicle turns to "UA" and so the bidding process is terminated (see Figure 2-(d)). Note that tasks 1 and 2 and vehicles A and B were not involved in the entire dispatching process. That is, the effect of changes was confined only to related tasks and vehicles. And also note that no central controller was involved in the bidding process.

A similar process (ASSIGN-VEHICLE-TO-A-NEW-TASK) will be followed when a new delivery task appears. Figure 3 illustrates the bidding process. The new task selects the vehicle (vehicle 1) with the lowest price that is the sum of the travel cost for vehicle to perform the new task and the current margin of vehicle 1 for performing the currently assigned task (task 2 in this example). The entering task submits a bid – which suggests a margin higher than the current margin of vehicle 1 – to vehicle 1. Then, vehicle 1 sends a cancellation notice to the currently assigned task (task 2). Then, task 2 begins the same procedure as what the new entering task did.

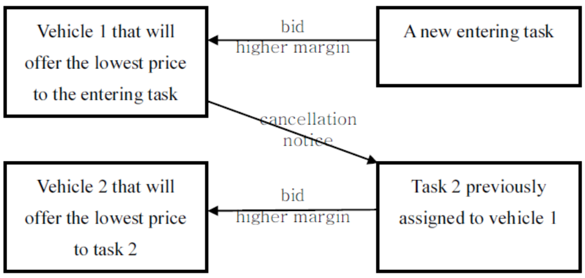

Figure 3: Procedure of ASSIGN-VEHICLE-TO-A-NEW-TASK [13]

A feasible assignment and a set of prices and margins that are at *almost equilibrium* are obtained through two bidding processes: ASSIGN-TASK-TO-A-NEW-VEHICLE and ASSIGN-VEHICLE-TO-A-NEW-TASK, which is explained in [13]. It was proved that for a given set of candidate vehicles and tasks, procedures of ASSIGN-TASK-TO-A-NEW-VEHICLE and ASSIGN-VEHICLE-TO-A-NEW-TASK enable a feasible assignment to (P1), and a set of prices and margins that are almost at equilibrium in a finite number of iterations.

In this paper, a_{ij} is evaluated by using the empty travel time to pick up task j after performing task i. Another alternative for evaluating a_{ij} is to use the response time - which is the time from the current location of the vehicle performing task i to the pick-up location of task j – for minimizing the total response time of vehicles, which is a possible objective in material handling systems. There may be cases where other objective terms need to be considered. Then, those objectives may be integrated into the objective function and used as the value of a_{ij}.

Initially, all tasks have the same price and all vehicles have the same margin. In the ASSIGN-VEHICLE-TO-A-NEW-TASK procedure, each task selects the best vehicle with the least required compensation, while each vehicle aims to obtain a higher margin. Both values are calculated basically using the a_{ij} value. Every time a task bids for a vehicle, the amount of the bid must cause the vehicle to have an increased margin, otherwise the task cannot bid for the vehicle. Every time the task is dispatched to a vehicle, its price is reduced. In the bidding procedure, less compensation and higher vehicle margins can be obtained if a pair of tasks and vehicles with lower a_{ij} value is dispatched. Therefore, the amount of the bid is automatically determined as the task selects its best vehicle, which is used later to determine the updated task price.

5 Numerical Experiment

The proposed bidding-based dispatching algorithm is compared with a greedy method using the Plant Simulation 11 software. In each problem, discharging operations are performed from one vessel served by 3-5 QCs. In the greedy method, which is an adaptation of the inventory-based dispatching method proposed in [16], after a vehicle completes performing an unloading task, the next target QC is selected, based on the weighted estimated horizontal distance and the current number of assigned vehicles to the QC. The comparison of the greedy and the bidding-based dispatching method is given in Table 1. It is shown that the bidding-based dispatching algorithm outperforms the greedy dispatching method for some objectives in some cases, which are written in bold at Table 1. The best assignments of vehicles and tasks are always being maintained in the bidding-based dispatching method by the dynamic dispatching function. In the proposed dispatching algorithm, if the states of tasks or vehicles change, only the affected tasks and vehicles are included in the updating process. Therefore, the update requires only short computational time. Hence, the bidding-based approach is well suited for real-time application in practice.

Table 1: Comparison between bidding-based and greedy dispatching methods.

Vessel ID	Number of discharging tasks	QC productivity (moves per hour)		Vessel berthing time		Total vehicle operation times		Average vehicle waiting time	
		Greedy	Bidding-based	Greedy	Bidding-based	Greedy	Bidding-based	Greedy	Bidding-based
1	1227	25.17	23.58	**9:53:15**	**9:34:18**	5:02:47:40	5:15:47:01	2:27	2:57
2	688	25.91	25.37	**7:24:17**	**7:06:44**	2:19:14:44	2:21:56:27	**2:19**	**2:07**
3	225	**27.33**	**27.48**	4:34:15	4:36:14	21:19:48	22:21:00	1:27	1:45
4	379	26.82	26.63	**5:54:08**	**5:52:44**	1:11:48:06	1:13:22:52	1:22	1:44
5	611	27.31	26.08	**6:52:46**	**6:30:33**	2:08:39:53	2:12:08:14	**1:53**	**1:44**

*Time is measured in days:hours:minutes:seconds

6 Conclusions

A bidding-based dispatching algorithm which is used to dispatch vehicles and tasks in an automated container terminal was proposed. In this dispatching algorithm, the assignments are dynamically updated in order to maintain the best assignments, which minimize task delays and the required travel times by the vehicles. The proposed bidding-based dispatching algorithm outperformed a greedy method which is an adaptation of the inventory-based dispatching method proposed in [16]. In future studies, a method for determining values of various parameters for the dispatching procedure needs to be developed.

Acknowledgement

This work was supported by the Korean Federation of Science and Technology Societies (KOFST), funded by the Korean government (MSIP: Ministry of Science, ICT and Future Planning), and Technological Development of Low-carbon Automated Container Terminals Project funded by the Ministry of Oceans and Fisheries, Korea (Project Number: 201309550004).

References

1. Powell, W.B., Gittoes, D.: An Approximate Labeling Algorithm for the Dynamic Assignment Problem, in: L. Bianco, P. Toth, M. Bielli, Eds., Advanced Methods in Transportation Analysis (Springer, New York) (1996)
2. Ichoua, S., Gendreau, M., Potvin, J.-Y.: Diversion Issues in Real-Time Vehicle Dispatching. Transportation Science 34(4), 426--438 (2000)
3. Frantzeskakis, L.F., Powell, W.B.: A Successive Linear Approximation Procedure for Stochastic, Dynamic Vehicle Allocation Problems. Transportation Science 24(1), 40--57 (1990)
4. Powell, W.B.: A Stochastic Formulation of the Dynamic Assignment Problem with An Application to Truckload Motor Carriers. Transportation Science 30(3), 195--219 (1996)

5. Carvalho, T.A., Powell, W.B.: A Multiplier Adjustment Method for Dynamic Resource Allocation Problems. Transportation Science 34(2), 150—164 (2000)
6. Gronalt, M., Hartl, R.F, Reimann, M.: New Savings Based Algorithms for Time Constrained Pickup and Delivery of Full Truckloads. European Journal of Operational Research 151(3), 520--535 (2003)
7. Gendreau, M., Guertin, F., Potvin, J.-Y., Seguin, R.: Neighborhood Search Heuristics for a Dynamic Dispatching Problem with Pick-ups and Deliveries. Transportation Research Part C 14, 157--174 (2006)
8. Imai, A., Nishimura, E., Current, J.: A Lagrangian Relaxation-Based Heuristic for the Vehicle Routing with Full Container Load. European Journal of Operational Research 176(1), 87--105 (2007)
9. Bertsekas, D.P.: A New Algorithm for The Assignment Problem. Mathematical Programming 21, 152--171 (1981)
10. Bertsekas, D.P.: The Auction Algorithm for Assignment and Other Network Flow Problems: A Tutorial. Interfaces 20(4), 133--149 (1990)
11. Egbelu, P.J., Tanchoco, J.M.A.: Characterization of Automatic Vehicle Dispatching Rules. International Journal of Production Research 22, 359--374 (1984)
12. Sabuncuoglu, I., Hommertzheim, D.L.: Dynamic Dispatching Algorithm for Scheduling Machines and Automated Guided Vehicles in a Flexible Manufacturing System. International Journal of Production Research 30, 1059--1079 (1992)
13. Lim, J.K., Kim, K. H., Yoshimoto, K., Lee, J.H., Takahashi, T.: A Dispatching Method for Automated Guided Vehicles by Using a Bidding Concept. OR Spectrum 25(1), 25--44 (2003)
14. Kim, K.H., Bae, J.W.: A Look-Ahead Dispatching Method for Automated Guided Vehicles in Automated Port Container Terminals. Transportation Science 38(2), 224--234 (2004)
15. Nguyen, V.D., Kim, K.H.: Dispatching Vehicles Considering Uncertain Handling Times at Port Container Terminals. Progress in Material Handling Research: Editors: Kimberly Ellis, Kevin Gue, Rene de Koster, Russell Meller, Benoit Montreuil, Mike Ogle. Material Handling Institute (2010)
16. Briskorn, D., Drexl, A., Hartmann, S.: Inventory-based Dispatching of Automated Guided Vehicles on Container Terminals. OR Spectrum 28, 611--630 (2006)
17. Choe, R., Kim, J., Ryu, K.R.: Online Preference Learning for Adaptive Dispatching of AGVs in an Automated Container Terminal. Applied Soft Computing 38, 647--660 (2016)
18. Carida, V.F., Morandin Jr, O., Tuma, C.C.M.: Approached of Fuzzy Systems Applied to an AGV Dispatching System in a FMS. International Journal of Advanced Manufacturing Technology 79, 615--625 (2015)
19. Miyamoto, T., Inoue, K.: Local and Random Searches for Dispatch and Conflict-free Routing Problem of Capacitated AGV Systems. Computers & Industrial Engineering 91, 1--9 (2016)
20. Co, C.G., Tanchoco, J.M.A.: A Review of Research on AGVS Vehicle Management. Engineering Costs and Production Economics 21, 35--42 (1991)
21. Gelareh, S., Merzouki, R., McGinley, K., Murray, R.: Scheduling of Intelligent and Autonomous Vehicles under Pairing/Unpairing Collaboration Strategy in Container Terminals. Transportation Research Part C 33, 1--21 (2013)
22. Tau, J., Qiu, Y.: A Simulation Optimization Method for Vehicles Dispatching Among Multiple Container Terminals. Expert Systems with Applications 42, 3742--3750 (2015)