Computers & Industrial Engineering 101 (2016) 352–360

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

Heuristic algorithm for retrieving containers

Yongmin Kim, Taeho Kim, HongChul Lee *

School of Industrial Management Engineering, Korea University, Anam-dong 5-ga, Seongbuk-gu, Seoul, Republic of Korea

ARTICLE INFO

Article history:
Received 4 August 2015
Received in revised form 31 July 2016
Accepted 26 August 2016
Available online 30 August 2016

Keywords:
Container retrieval
Heuristic
Container terminal

ABSTRACT

This study proposes a heuristic algorithm to achieve efficient container retrieval given a specified order, by minimizing the number of container movements and the working time of the crane. We evaluated our proposed algorithm by solving 70 sample problems. The results confirmed the validity and efficiency of the algorithm compared to methods used in previous research. Our algorithm succeeds in reducing the number of container movements and the working time of the crane. Furthermore, the heuristic algorithm is computationally efficient and greatly reduces the amount of time required to obtain a solution. Therefore, we expect our heuristic to be useful in the real-world industry where rapid decision-making is required.

1. Introduction

The global increase in the number of multinational corporations has led to a concomitant increase in the demand for international trade. Accordingly, container terminals, which form the touch point between conveyance by land and by water, have become equally important. Moreover, in the last five decades the container as an essential part of the unit load-concept has achieved undoubted importance in international sea freight transportation (Steenken, Voß, & Stahlbock, 2004). The cost resulting from a container passing through a container terminal accounts for nearly one-third of total container transportation cost. As a result, an improvement in the operational efficiency of a container terminal is required to ensure that customer demands are met and costs are reduced. Reflecting these recent trends, this study aims to suggest an efficient container retrieval method to improve the operational efficiency of a container terminal.

2. Background

Many researchers have devoted studies to solving efficiency problems relating to container terminals and achieved varying results. Chung, Randhawa, and McDowell (1988) proposed a methodology involving the use of a buffer space as a method to increase the utilization of the material handling equipment and reduce the total container loading time. They developed a simulation model using a graphics simulation system to compare their proposed methodology with the then practice at the Port of Portland. This was an early study to attempt to optimize the retrieval process in a container yard. Kim and Kim (1997) studied the routing problem for a single transfer crane to load export containers onto a containership. They formulated the routing problem as an integer programming problem to minimize the total container handling time of the transfer crane including the set-up time at each yard-bay and the travel time between consecutive yard-bays. As an extension thereof, Kim and Kim (1999) focused on how to optimally route transfer cranes in a container yard during loading operations of export containers at port terminals. Wang and Zhu (2014) presented a rail-mounted gantry crane (RMGC) scheduling optimization model, whose objective was to determine an optimization handling sequence in order to minimize the RMGC idle load time in handling tasks.

Avriel, Penn, Shpirer, and Witteboon (1998) formulated an integer programming model for a stowage plan of containers in a container ship. The objective of their model was to minimize the number of containers shifted and to minimize the number of rehandling operations. Wang, Zhu, and Xie (2014) formulated a storage space allocation problem in a railway container terminal as a two-stage optimization model, whose objectives were balancing the workload of inbound containers and reducing the amount of overlap.

Kim (1997) proposed a methodology to estimate the expected number of rehandlings for any arbitrary container and the total number of rehandlings for all the containers in a bay for a given initial stacking configuration. They provide simple tables and equations to facilitate the estimation of the number of rehandlings. Both Kim and Hong (2006) and Wan, Liu, and Tsai (2009) developed heuristics to substitute the time-consuming optimization

---

* Corresponding author.
E-mail addresses: iamkym@korea.ac.kr (Y. Kim), airth@korea.ac.kr (T. Kim), hclee@korea.ac.kr (H. Lee).

http://dx.doi.org/10.1016/j.cie.2016.08.022
0360-8352/

procedures. Kim and Hong (2006) suggested a heuristic rule to minimize the number of relocations during pickup operations affecting a single bay. Their heuristic used a decision rule for determining the locations of relocated blocks. The heuristic rule proposed used an estimator for an expected number of additional relocations for a stack. They compared the performance of their heuristic with the performance of a branch-and-bound algorithm. Wan et al. (2009) minimized the number of reshuffles in assigning storage locations for incoming and reshuffled export containers. They developed the first optimization-based method to capture the interaction of stacking containers in a dynamic setting with continual container arrivals and retrievals.

Lee and Lee (2010) presented a three-phase heuristic to solve for an optimized working plan for a crane to retrieve all containers from a given yard according to a given order. They solved 70 examples to demonstrate the performance of their heuristic, and the results showed that the heuristic was able to solve instances involving more than 700 containers. Unluyurt and Aydin (2012) solved the problem of optimizing the retrieval of containers using a branch and bound-based algorithm and alternative heuristic. Their algorithm produced optimal solutions and their alternative heuristic produced results that closely approximated the optimal solution. Lin, Lee, and Lee (2015) developed a heuristic to solve the container retrieval problem by minimizing both the number of container movements and the working time of the rail-mounted gantry cranes (RMGC). Moreover, they demonstrated that using multiple spreaders can reduce both the RMGC working time and the number of container movements.

The research efforts cited above confirm that studies attempting to address efficient container handling have been carried out actively. Meanwhile, the purpose of this study is to suggest efficient container retrieval by using a heuristic algorithm. The purpose of this study is similar to that of Lee and Lee, the result of which is noteworthy in terms of quantitative performance; however, the gap between their heuristic solutions and the lower bounds remains. Furthermore, the solving time in Lee and Lee's study is too long in spite of the use of a heuristic. This led us to suggest a distinct heuristic that aims to reduce the solving time and to obtain solutions closer to the lower bounds.

3. Problem description

The main task in a container terminal is to retrieve and transfer containers from the container yard to load onto a container truck for shipping, according to a given order. A container yard is divided into bays, each of which accommodates many stacks of containers. The most ideal situation for container retrieval from a yard is to retrieve the container at the top of each stack by priority. If a target container is not located at the top of a stack, there is a delay caused by moving all containers above the target container to another stack. We refer to this additional work as rehandling, and it is important to consider which stack is selected for relocation because naïve reallocation may cause subsequent rehandling.

Fig. 1 shows an image of a container yard where a container is being retrieved, similarly, Fig. 2 represents a simple container yard with containers stacked in random order. The retrieval of a container is based on the following four assumptions.

(i) All containers are retrieved according to a given numerical order.
(ii) A crane can only lift the container located at the top of each stack.
(iii) Decreasing the total number of container movements reduces the working time of the crane.
(iv) The height of a stack is limited.

Fig. 1. A yard where a container is being retrieved.

Applying these assumptions to the container retrieval situation of Fig. 2, container 1 is retrieved without additional rehandling because it is located at the top of stack 6. Thus, the crane lifts container 1 from stack 6 onto a container truck, which is stack 0, and we designate this work by using the triplet (1:6,0). The triplet (1:6,0) means that move item 1 from stack 6 to stack 0. Contrary to container 1, container 2 needs additional rehandling because it is not located at the top of stack 1. Therefore, containers 6 and 8 above container 2 have to be relocated before container 2 can be retrieved. If stack 5 is selected for the relocation of containers 6 and 8, the generated movements are (6:1,5) and (8:1,5). As a preview of our algorithm, we discuss how these movements (6:1,5) and (8:1,5) could be improved. If we decide to perform rehandling works (6:1,5) and (8:1,5) for retrieving container 2, the entire process for retrieving all containers is as follows; (1:6,0), (6:1,5), (8:1,5), (2:1,0), (3:4,0), (9:3,4), (4:3,0), (13:6,3), (5:6,0), (8:5,4), (6:5,0), (11:2,3), (7:2,0), (8:4,0), (9:4,0), (10:5,0), (11:3,0), (12:4,0), (13:3,0). Thus, total number of container movements is 19. On the other hand, if we move containers as follows; (1:6,0), (6:1,2), (9:3,5), (8:3,5), (2:1,0), (3:4,0), (4:3,0), (13:6,3), (5:6,0), (6:2,0), (11:2,4), (7:2,0), (8:5,0), (9:5,0), (10:5,0), (11:4,0), (12:4,0), (13:3,0), total number of container movements is 18. Through it, we can deduce that efficient algorithm for container retrieval could reduce the movements of containers.

4. Description of the heuristic algorithm

We introduce newly defined terms for the sake of convenient algorithm description before we describe the proposed heuristic.

- Ideal stack: Empty stack or a stack for which the specified order of containers is in top-bottom ascending order.
- Unideal stack: A stack for which the specified order of containers is not in top-bottom ascending order.
- Critical stack: Unideal stack in which a container exists that has to be retrieved prior to containers above it.
- TN(Top Number): Specified order of the container located at the top of a stack.
- TNC(Top Number of Critical stack): TN of critical stack.
- c_{min}: Container which has minimum order in a yard.
- $tnc_{c_{min}}$: TNC of the stack where c_{min} is located.
- TN_{Ideal}: TN of a set of ideal stacks.
- $TNC_{Critical}$: TNC of a set of critical stacks.
- Candidate stack: The stack of which TN is in $A = \{x | x \in TN_{Ideal} \text{ and } x > tnc_{c_{min}}\}$.

In Fig. 2 stacks 4 and 5 are ideal stacks that do not require rehandling. Meanwhile, stacks 1, 2, 3, and 6 are unideal stacks. Among these stacks, stacks 1, 2, and 3 are critical stacks. The TNC