Advanced Engineering Informatics xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Advanced Engineering Informatics

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

A domain-specific knowledge-based heuristic for the Blocks Relocation Problem

Christopher Expósito-Izquierdo, Belén Melián-Batista, J. Marcos Moreno-Vega *

Departamento de Estadística, IO y Computación, ETS de Ingeniería Informática, Universidad de La Laguna, Spain

ARTICLE INFO

Article history:
Received 11 December 2012
Received in revised form 28 February 2014
Accepted 30 March 2014
Available online xxxx

Keywords:
Blocks Relocation Problem
Maritime container terminal
Heuristics
Logistics

ABSTRACT

The Blocks Relocation Problem consists in minimizing the number of movements performed by a gantry crane in order to retrieve a subset of containers placed into a bay of a container yard according to a predefined order. A study on the mathematical formulations proposed in the related literature reveals that they are not suitable for its solution due to their high computational burden. Moreover, in this paper we show that, in some cases, they do not guarantee the optimality of the obtained solutions. In this regard, several optimization methods based on the well-known A^* search framework are introduced to tackle the problem from an exact point of view. Using our A^* algorithm we have corrected the optimal objective function value of 17 solutions out of 45 instances considered by Caserta et al. (2012) [4]. In addition, this work presents a domain-specific knowledge-based heuristic algorithm to find high-quality solutions by means of short computational times. It is based on finding the most promising positions into the bay where to relocate those containers that are currently located on the next one to be retrieved, in such a way that, they do not require any additional relocation operation in the future. The computational tests indicate the higher effectiveness and efficiency of the suggested heuristic when solving real-world scenarios in comparison with the most competitive approaches from the literature.

1. Introduction

The container yard is an outstanding functional area of maritime container terminals due to the fact that it brings together the main flows of containers; those established between the quayside and the landside [15]. Generally, container yards are split into several blocks, that is, sets of container bays arranged in parallel. A bay is a delimited two-dimensional stock disposed in the vertical direction, where containers can be placed over each other. The container stacking poses a great limitation for their handling because they are accessed by following the Last In First Out (LIFO) policy, in such a way that, only those containers placed at the top of the stacks are accessible directly.

A large variety of handling machinery has been proposed to manage containers. The most common are reach stackers, straddle carriers, Rubber-Tyred Gantry Cranes (RTGCs), Rail-Mounted Gantry Cranes (RMGCs) and Overhead Bridge Cranes (OBGs) [21]. Nowadays, the RMGCs are the most widely extended systems in modern terminals due to their high performance and potential for automation. The dimensions of RMGCs allow to use large-dimensioned container blocks, that is, blocks comprising 40–50 bays composed of 6–8 stacks and 5–6 tiers. This type of cranes can move along the blocks by means of a pair of rail tracks placed on both sides. The access to containers is carried out following a well-defined set of steps in succession [29]. Firstly, the crane is positioned over the bay where the container to retrieve is located. Later, its trolley is moved toward a position above the target stack and its spreader is lowered in order to achieve the target container. Once the container is hooked up, this can be picked up and move it outside the bay.

Accurate information about the containers (weight, destination, size, etc.) allows to reach a fluent exchange between the transportation modes brought at the terminal. Unfortunately, most of the times the available information is somehow inaccurate or incomplete due to the intrinsic features associated with the environment. Some sources of disturbances are traffic jams in which container trucks may be involved, changes in the arrival time of vessels, and so forth. In this regard, the container retrieval order is strongly influenced by unforeseen changes. In spite of conducting an exhaustive analysis of container handling along the yard, some containers can be placed below other ones when they have to be retrieved. This fact gives rise to relocate before those containers currently placed on the containers to retrieve. Implementing this type of movements adversely affects the performance of the

* Corresponding author. Tel.: +34 922318175.
E-mail addresses: cexposit@ull.es (C. Expósito-Izquierdo), mbmelian@ull.es (B. Melián-Batista), jmmoreno@ull.es (J. Marcos Moreno-Vega).

http://dx.doi.org/10.1016/j.aei.2014.03.003
1474-0346/

Please cite this article in press as: C. Expósito-Izquierdo et al., A domain-specific knowledge-based heuristic for the Blocks Relocation Problem, Adv. Eng. Informat. (2014), http://dx.doi.org/10.1016/j.aei.2014.03.003

terminal because they delay the delivery of goods being, therefore, considered as unproductive.

This paper addresses the Blocks Relocation Problem (BRP), whose main goal is to determine the sequence of movements performed by a gantry crane to retrieve a subset of containers placed into a bay of the yard according to a predefined order. The objective is to find the sequence of movements with the minimum number of relocation operations.

The major contributions of the present paper are the following:

i. Exhaustive description of the BRP. A complete analysis of the different variants already addressed in the literature (Restricted and Unrestricted BRP), a classification of the containers according to their priorities and the slots in which they are placed and several useful remarks to exploit the structure of the solutions in optimization techniques are provided.
ii. Study of the mathematical formulations for the BRP published in the related literature. Several examples in which the optimality of the reported solutions by Caserta et al. [4] is not guaranteed are showed.
iii. Development of two exact algorithms based on the A* search framework devoted to solve the Restricted and Unrestricted BRP. Several lower bounds and an algorithm aimed at finding an upper bound are proposed to prune the underlying trees. The computational results show that they can be successfully applied to small-sized practical scenarios. Additionally, using our A* algorithm we have corrected the optimal objective function value of 17 solutions out of 45 instances considered by Caserta et al. [4] for the Restricted BRP.
iv. Development of a heuristic algorithm aimed at reducing the high computational burden required by the exact approaches when solving the BRP in large scenarios. The computational results indicate that the performance of this solution approach is suitable for a large variety of practical scenarios and provides statistical significant differences with the most competitive algorithms from the related literature.

The remainder of this paper is organized as follows. Firstly, Section 2 presents a complete description of the BRP. Later, Section 3 overviews the most highlighted contributions of the works published in the literature. Section 4 discusses the applicability of the mathematical models proposed so far and provides two exact algorithms based on the well-known A* search framework that can be used as benchmark approaches. Section 5 describes a heuristic technique to solve the BRP from an approximate point of view. Section 6 analyses a broad set of computational tests conducted to assess the performance of the suggested approaches and their most influential components. Lastly, Section 7 depicts the summary and main conclusions extracted from the work.

2. Blocks Relocation Problem

The Blocks Relocation Problem (BRP) is a combinatorial optimization problem described as follows. Given a set of n uniform containers placed into a bay with S stacks and T tiers, the goal of the BRP is to find the sequence of movements with minimum length that should be performed by a gantry crane in order to retrieve k given containers one after the other in a predefined order, where $k \leqslant n$. A few different objective functions related to the working time of the handling equipment may be proposed. Nevertheless, minimizing the number of movements is the most widely extended in the literature (see Section 3). This is due to the fact that the time used to move the containers among stacks is usually negligible with respect to that required to perform the coupling and lifting/dropping tasks of containers [11].

The BRP is restricted by the dimensions of the bay. The capacity of the bay depends on the number of stacks and tiers, $C = S \times T$, that is, the number of slots where containers can be placed for their storage. From a general point of view, each slot could contain at most one container. Furthermore, each container has associated an exclusive priority or group representing its position within the retrieval order. Without loss of generality, it is assumed that the retrieval order is defined by following the container priorities, in such a way that, the container with the highest priority, 1, must be retrieved before container 2; container 2 must be retrieved before container 3; and so forth, until all the k containers with the highest priorities are retrieved. After the retrieval process finishes, the bay will only contain the $m = n - k$ containers with the lowest priorities. The BRP is already known to be $\mathcal{NP}$-hard (see Caserta et al. [4]) so that efficient optimization techniques must be developed to tackle medium and large scenarios.

Some notation must be introduced to ease the perusal of the remainder of this paper. Given a container c, let $s(c)$ and $t(c)$ be the stack and the tier in which c is currently placed, respectively. Also, let $p(c)$ be the priority of container c and c^* the next container to retrieve from the bay. The number of containers placed in the stack s is denoted by $h(s)$. The highest priority of a container in the stack s is denoted by $max(s)$ and defined as follows:

$$max(s) = \begin{cases} \min\{p(c) \mid s(c) = s\}, & \text{if } h(s) > 0 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

The retrieval of containers from the bay can be performed by following two general types of movements. Firstly, retrieval movements are those in which a container currently placed at the top of some stack is moved outside the bay. After implementing a retrieval movement the number of containers into the bay is decreased in one unit. This type of movements are considered as productive because they give answer to specific service requests. It is straightforward to check that each feasible solution for the BRP has exactly k retrieval movements. On the other hand, relocation movements are those in which a container placed at the top of some stack is moved toward the top of another one. The target stack must contain at least one empty slot. Unlike retrieval operations, implementing this type of movement does not decrease the number of containers into the bay. It is worth mentioning that the bay should have empty slots to store the relocated containers. In this regard, at least $H = T - 1$ empty slots must be available before the retrieval process starts. This value stems from the fact of accessing to a container placed at the lowest tier in a full stack, for which the $T - 1$ containers placed on it must be previously relocated to empty slots in the same bay [4].

Fig. 1 illustrates an example of the BRP in which there is a bay with 6 stacks, 4 tiers and 16 containers that must be retrieved, that is, $k = n$. As defined by the capacity of the bay, a maximum of 24 containers can be stored. The priority of each container is represented by a number indicating the order in which it must be retrieved from the bay.

A general classification of the containers according to their priorities and the slots where they are placed is proposed hereunder. Firstly, a non-located container is that placed at a higher tier than another one in the same stack with a higher priority. See striped containers in the example depicted in Fig. 1. Let $\Omega(s)$ be the set of non-located containers in stack s, defined as follows:

$$\Omega(s) = \{c \mid (s(c) = s) \wedge \exists\, c' : (s(c') = s) \wedge (t(c') < t(c)) \wedge (p(c') < p(c))\}, \quad \forall 0 \leqslant s < S. \quad (2)$$

The set of non-located containers into the bay is defined by