European Journal of Operational Research 239 (2014) 297–312

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Invited Review

Loading, unloading and premarshalling of stacks in storage areas: Survey and classification

Jana Lehnfeld, Sigrid Knust *

Institute of Computer Science, University of Osnabrück, Germany

ARTICLE INFO

Article history:
Received 26 July 2013
Accepted 7 March 2014
Available online 20 March 2014

Keywords:
Loading
Unloading
Premarshalling
Storage
Stack
Classification scheme

ABSTRACT

Problems of loading, unloading and premarshalling of stacks as well as combinations thereof appear in several practical applications, e.g. container terminals, container ship stowage planning, tram depots or steel industry. Although these problems seem to be different at first sight, they hold plenty of similarities. To precisely unite all aspects, we suggest a classification scheme and show how problems existing in literature can be described with it. Furthermore, we give an overview of known complexity results and solution approaches.

1. Introduction

The interim storage of items in a storage area leads to different kinds of optimization problems. On the one hand, if incoming items arrive at a storage area, they need to be assigned to positions which causes *loading problems*. On the other hand, *unloading problems* arise if outgoing items need to be retrieved from the storage area and one has to decide which items will leave the storage in which order and which relocations are performed. *Premarshalling* occurs if items have to be sorted inside the storage area such that all items can be retrieved without any further reshuffle afterwards. If incoming items need to be stored while outgoing items need to be retrieved, *combined loading/unloading problems* appear.

In this paper, we concentrate on storage problems where the storage area is organized in stacks and items are put on top of each other in these stacks. Each stack may have a limited height and its own fixed position in a two-dimensional area. The items are assumed to be cuboids, e.g. containers or wooden/steel plates. Problems dealing with the storage of round items require another layout and other stacking conditions (cf. Tang, Zhao, and Liu (2012) and Zäpfel and Wasner (2006)).

A storage area can either be a yard, a warehouse or a tram/bus depot, but also a container ship where containers are stored in the bunt. For a comparative survey of different stacking policies in container terminals see Dekker, Voogd, and van Asperen (2006). An overview of warehouse management problems and different warehousing models can be found in van den Berg and Zijm (1999). De Koster, Le-Duc, and Roodbergen (2007) provide a survey of how to retrieve items in high rack warehouses where items are stored in racks and accessible from the side. High rack warehouses with movable shelves are also called "puzzle-based storage systems" (cf. Gue and Kim (2007) and Alfieri, Cantamessa, Monchiero, and Montagna (2012)) since the problem of moving a shelf to the exit is similar to the so-called 15-puzzle (a classical children's game).

In practice, all operations to move and get items are executed by cranes located above the stacks so that direct access is possible only to the topmost item of any stack, i.e. the items are arranged in LIFO-order (last in, first out). This implies that if an item stacked below has to be retrieved, so-called *reshuffling* (or *relocation*) is necessary. We assume that once a required item is the topmost item of a stack, it can be retrieved instantaneously. An example with two stacks and four items is given in Fig. 1. If the gray item at the bottom of the left stack has to be retrieved, two reshuffling operations are needed to free this item. Since reshuffling operations are usually very time-consuming, they should be avoided as often as possible.

Loading problems deal with the storage of incoming items. Each item reaching the storage area has to be assigned to a feasible location. In a pure loading problem, it is assumed that no outgoing item is retrieved during the loading phase. Loading problems typically

* Corresponding author. Tel.: +49 541 969 2483.
E-mail addresses: jana.lehnfeld@uni-osnabrueck.de (J. Lehnfeld), sknust@uos.de (S. Knust).

http://dx.doi.org/10.1016/j.ejor.2014.03.011
0377-2217/

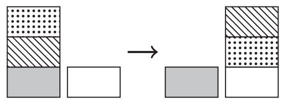

Fig. 1. An example for reshuffling.

arise in container terminals where containers arrive by truck, train or vessel and have to be stored until they are required to be retrieved. Since the individual retrieval times of the items are usually unknown at the time of loading, the objective often aims at minimizing the number of expected reshufflings (cf. Jang, Kim, and Kim (2013) and Kim, Park, and Ryu (2000)).

Unloading problems deal with the retrieval of items from a storage area. Depending on the type of problem one has either to define the order in which items are retrieved or decide which items are to retrieve in order to fulfil certain requests. Additionally, the way, how relocations are performed, has to be determined. In Fig. 2, an example for an unloading problem with 8 items can be found. Given an area with four stacks, each limited to hold three items at the most, the items have to be retrieved in the order $1, 2, \ldots, 8$. In order to do this, 4 reshuffles are needed in total. This is the minimum number of reshuffles since item 4 needs to be reshuffled to get access to item 2, item 6 needs to be reshuffled to get access to item 3, item 8 needs to be reshuffled to get access to item 1 and no matter which stack item 8 will be reshuffled to, at least one more reshuffle becomes necessary.

In a pure unloading problem, it is assumed that there are no incoming items during the unloading phase. Many storage unloading problems are inspired by retrieving containers from container terminals to store them onto vessels, trains or trucks (for an overview see e.g. Caserta, Schwarze, and Voß (2011a)). Other practical applications for storage unloading occur in steel industry or tram/bus depots. In the former, steel slabs are stored in stacks and have to be retrieved in a given order to a hot rolling mill. In the latter, items are stored on sidings with dead ends, so they can be modeled as (vertical) stacks, organized in LIFO-order. The waggons stored on the sidings have to be retrieved to join several waggons to one tram. Since access is only possible from one end of the siding, shunting might be necessary to retrieve a required waggon. For example, Blasum et al. (1999), Gallo and Di Miele (2001) and Gatto, Maue, Mihalák, and Widmayer (2009) deal with shunting problems in tram/bus depots. In both, steel industry and tram depots, items are often grouped into families. Then, a retrieval request does not ask for a specific item, but for any item of a specific family (cf. for example Tang, Liu, Rong, and Yang (2002)).

Premarshalling problems occur if items inside a storage area have to be sorted such that they can be retrieved without any additional reshuffle afterwards. The difference to an unloading problem is that during premarshalling no item leaves the storage area, i.e. the area always contains the same set of items (cf. for example Lee and Hsu (2007), Caserta and Voß (2009a) and Bortfeldt and Forster (2012)). In Fig. 3, one can find an example for a premarshalling problem with 8 items. The initial layout is the same as in Fig. 2 with an area of four stacks each limited to hold at most three items. The items have to be sorted such that they can be retrieved in the order $1, 2, \ldots, 8$. This can again be done with 4 reshuffles in total (which is the minimum number of reshuffles, see the explanation above).

Combined problems mostly occur as a combination of loading and unloading, which means that new items arrive and need to be stored simultaneously to the retrieval process. Predominantly, combined problems can be found in container stowage planning where vessels are used to transport containers and call sequentially at several ports (see e.g. Aslidis (1989) and Avriel, Penn, Shpirer, and Witteboon (1998)). The containers' departure and destination ports are known in advance determining the loading and unloading processes, respectively. Other combined problems occur as a combination of loading and premarshalling (see Malucelli, Pallottino, and Pretolani (2008)). In this case, several sets of incoming items have to be stored sequentially. During the loading process, premarshalling may be performed with the goal that at the end of the loading process all items can be retrieved without any additional reshuffle.

Since storage problems originate from a wide range of practical applications, a lot of scientific literature exists dealing with problems motivated from practice. Although these problems are described in different terms and notions, they share several similar features. For example, both in warehouses and in container yards, items are piled up on each other accessible from the top only. A main difference is that warehouses have higher stacks (30–40 items may be stacked in contrast to approximately 4–5 containers in a container yard) and handle much more items. Furthermore, the item flow of retrieving and storing is done in parallel while in container yards these operations are done one after another (cf. Caserta et al. (2011a)).

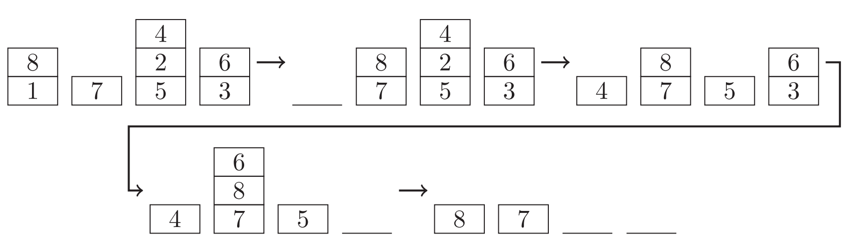

Fig. 2. An example for unloading.

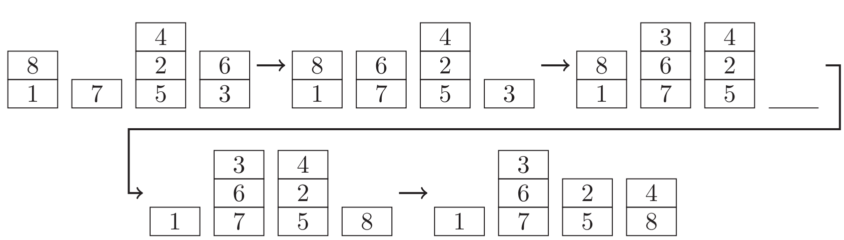

Fig. 3. An example for premarshalling.