An optimal algorithm for on-line palletizing at delivery industry

(extended abstract)

J. Rethmann E. Wanke

University of Düsseldorf, Dept. of Computer Science, D-40225 Düsseldorf, Germany
{rethmann,wanke}@cs.uni-duesseldorf.de

Abstract. We consider the combinatorial stack-up problem which is to process a given sequence q of bins by removing step by step bins from the first s positions of the sequence onto p stack-up places. We prove that the Most-Frequently algorithm has best worst-case performance of all *on-line* stack-up algorithms and is, additionally, the best polynomial time approximation algorithm for the stack-up problem known up to now, although it is a simple *on-line* algorithm.

1 Introduction

In this paper, we consider the combinatorial problem of stacking up bins from a conveyer onto pallets. This problem basically appears in so-called *stack-up systems* which play an important role in delivery industry and warehouses. Customers order a large amount of articles, which have to be packed into bins for delivery. This work is usually done in so-called *pick-to-belt orderpicking systems*, see [dK94,LLKS93]. Typically, not all articles ordered by one customer fit into a single bin, so each order is divided into several bins. For delivery, it is necessary that all bins belonging to one order are placed onto the same pallet, and bins for different orders are placed onto different pallets. Bins arrive the stack-up system on the main conveyer of the orderpicking system. At the end of the main conveyer they enter a *storage conveyer*. The bins are picked-up by stacker cranes from the storage conveyer and moved onto pallets, which are located at so-called *stack-up places*. Automatic driven vehicles take full pallets from stack-up places, put them onto trucks, and bring new empty pallets to the stack-up places, see also [RW97c].

Logistic experience over 10 years lead to such high flexible conveyer systems in delivery industry. So we do not intend to modify the architecture of existing systems, but try to develop a model that captures important parameters of stack-up systems, and to develop efficient and provable good algorithms to control them. Today, real stack-up systems are controlled by fuzzy logic. Since almost all decisions have to be done at a time where only a part of the complete sequence is known, we focus our work on on-line algorithms.

From a theoretical point of view, we are given a sequence of pairwise distinct bins $q = (b_1, \dots, b_n)$ and two positive integers s and p. Each bin b_i of q is destined for one pallet, where the number of bins for a pallet is arbitrary. The bins have to be removed step by step from q. If a bin is removed from the sequence, then all bins to the right are shifted one position to the left. The position of a removed bin must not be greater than s, and after each removal of a bin there must be at most p pallets open. A pallet t is called open, if at least one bin for t is already removed from sequence q, but at least one bin for t is still contained in the current sequence. Integer s corresponds to the capacity of the storage conveyer, and integer p corresponds to the number of stack-up places, because each open pallet occupies one stack-up place. A sequence q is called an (s, p)-sequence if it can be processed with a storage capacity of s bins and p stack-up places.

The following results are already known about the stack-up problem. In [RW97c] it is shown that the stack-up decision problem is NP-complete [GJ79], but belongs to NL if the storage capacity s or the number of stack-up places p is fixed. In [RW97b] the performances of several *on-line* algorithms are compared with optimal off-line solutions by a competitive analysis [MMS88]. The Most-Frequently algorithm – or MF algorithm for short – processes each (s, p)-sequence with a storage capacity of $(p + 1)s - p$ bins and p stack-up places, or with a storage capacity of s bins and at most $p \cdot (\log_2(s) + 2)$ stack-up places, respectively. In [RW97a], a polynomial time *off-line* approximation algorithm for the processing of (s, p)-sequences with a storage capacity of $s \cdot \lceil \log_2(p+1) \rceil$ and $p + 1$ stack-up places is introduced.

In contrast to the performance analysis given in [RW97b], we show in this paper that the MF algorithm is even an *optimal* on-line stack-up algorithm. Furthermore, we do not only measure the performance in terms of one objective, but we also relax both resources, the storage capacity as well as the number of stack-up places. We look at on-line stack-up algorithms that take a storage capacity of $s + c$ bins to process (s, p)-sequences, where c is any nonnegative integer. We give a lower bound on the number of stack-up places each such algorithm takes, and prove an upper bound of $p + p \cdot \log_2(\frac{s}{c+1} + 1) + 1$ the MF algorithm takes. Furthermore, we show that the MF algorithm processes each (s, p)-sequence q with a storage capacity of $2s$ bins and $2p$ stack-up places. Therefore, the MF algorithm is the best polynomial time approximation algorithm for the stack-up problem known up to now that approximates both objectives within a small factor. We emphasize that this approximation can be achieved by a very simple *on-line* algorithm. So Most-Frequently seems to be the right algorithm to put into practice.

References

[dK94] R. de Koster. Performance approximation of pick-to-belt orderpicking systems. *European Journal of Operational Research*, 92:558–573, 1994.

[GJ79] M.R. Garey and D.S. Johnson. *Computers and Intractability.* W.H. Freeman and Company, San Francisco, 1979.

[LLKS93] E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, and D.B. Shmoys. Sequencing and Scheduling: Algorithms and Complexity. In S.C. Graves, A.H.G. Rinnooy Kan, and P.H. Zipkin, editors, *Handbooks in Operations Research and*

Management Science, vol. 4, Logistics of Production and Inventory, pages 445–522. North-Holland, Amsterdam, 1993.

[MMS88] M.S. Manasse, L.A. McGeoch, and D.D. Sleator. Competitive algorithms for on-line problems. In *Proceedings of the Annual ACM Symposium on Theory of Computing*, pages 322–333. ACM, 1988.

[RW97a] J. Rethmann and E. Wanke. An approximation algorithm for stacking up bins from a conveyer onto pallets. In *Proceedings of the 5th Annual Workshop on Algorithms and Data Structures*, volume 1272 of *Lecture Notes in Computer Science*, pages 440–449. Springer-Verlag, 1997.

[RW97b] J. Rethmann and E. Wanke. Competitive analysis of on-line stack-up algorithms. In *Proceedings of the 5th Annual European Symposium on Algorithms*, volume 1284 of *Lecture Notes in Computer Science*, pages 402–415. Springer-Verlag, 1997.

[RW97c] J. Rethmann and E. Wanke. Storage Controlled Pile-Up Systems. *European Journal of Operational Research*, 103(3):515–530, 1997.