An approximation algorithm for stacking up bins from a conveyer onto pallets

J. Rethmann and E. Wanke

University of Düsseldorf, Department of Computer Science, D-40225 Düsseldorf, Germany

Abstract. Given a sequence of bins $q = (b_1, \ldots, b_n)$ and a positive integer p. Each bin is destined for a pallet. We consider the problem to remove step by step all bins from q such that the positions of the bins removed from q are as less as possible and after each removal there are at most p open pallets. (A pallet t is called open if the first bin for t is already removed from q but the last bin for t is still contained in q. If a bin b is removed from q then all bins to the right of b are shifted one position to the left.)
The maximal position of the removed bins and the maximal number of open pallets are called the storage capacity and the number of stack-up places, respectively. We introduce an $O(n \cdot \log(p))$ time approximation algorithm that processes each sequence q with a storage capacity of at most $s_{\min}(q,p) \cdot \lceil \log_2(p+1) \rceil$ bins and $p+1$ stack-up places, where $s_{\min}(q,p)$ is the minimum storage capacity necessary to process q with p stack-up places.

1 Introduction

From a practical point of view, a stack-up system consists of one or more stacker cranes picking up bins from a roller conveyer. It is usually located at the end of a pick-to-belt orderpicking system; see [dK94, LLKS93] for a description of orderpicking systems. A costumer order consists of several bins that arrive the stack-up system on a conveyer. At the end of the conveyer the bins enter a so-called storage conveyer from that they are moved by stacker cranes onto pallets. The pallets are build on so-called stack-up places. Vehicles take full pallets from stack-up places, put them onto trucks, and bring new empty pallets to the stack-up places. A more detailed description of stack-up systems which are sometimes also called pile-up systems is given in [RW97b].

All bins of a costumer order have to be placed onto the same pallet. Unfortunately, the bins arrive the stack-up system not in a succession such that they can be placed one after the other onto pallets. If all available places are occupied by pallets and the storage conveyer is completely filled with bins not destined for the pallets currently stacked up on the places, then the stack-up system is blocked. Controlling a stack-up system means to make the right decision whenever a new pallet has to be opened such that no blocking situation will happen.

From a theoretical point of view, the stack-up problem can be defined as follows. Given a sequence of bins $q = (b_1, \ldots, b_n)$ and two integers s and p. Each bin is destined for some pallet t. A pallet t is called open if the first bin for t is already removed from q but the last bin for t is still contained in q. The stack-up problem is the question of whether all bins can be removed step by step from q such that the positions of the bins removed from q are always not greater than s and after each removal there are at most p open pallets. If a bin b is removed from q then all bins to the right of b are shifted one position to the left. The integers s and p correspond to the capacity of the storage conveyer and the number of available stack-up places, respectively.

In [RW97b], it is shown by a reduction from 3-SAT that the stack-up problem is in general NP-complete [GJ79] but can be solved in polynomial time if the storage capacity or the number of stack-up places is bounded. It is also shown that it can be solved in polynomial time whether or not it is possible to block the stack-up process by making wrong decisions.

In [RW97a], the worst-case behavior of on-line stack-up algorithms is analyzed. Such on-line algorithms make a decision without knowing the complete sequence, but only the next s bins. On-line algorithms are very interesting from a practical point of view. Let $s_{\min}(q,p)$ be the minimum storage capacity necessary to process q with p stack-up places and let $p_{\min}(q,s)$ be the minimum number of stack-up places necessary to process q with storage capacity s. In [RW97a] the performances of on-line algorithms are compared with optimal off-line solutions by competitive analysis [MMS88]. It is shown that there exist on-line algorithms having a logarithmic bound of $O(p_{\min}(q,p) \cdot \log(s))$ on the used number of stack-up places if the storage capacity is bounded by s, but only a linear bound of $(p+1) \cdot s_{\min}(q,p) - p$ on the storage capacity necessary to process q with p stack-up places.

The stack-up problem seems to be not much investigated up to know by other authors, although it really has important practical applications. In this paper, we introduce a polynomial time approximation algorithm for the processing of sequences q with a storage capacity of $\lceil s_{\min}(q,p) \cdot \log_2(p+1) \rceil$ bins and $p+1$ stack-up places. This is the first polynomial time approximation algorithm for the stack-up storage minimization problem with a logarithmic bound on the storage capacity when using only $p+1$ stack-up places. The existence of a polynomial time approximation algorithm that processes each sequence q with a storage capacity of less than $s_{min}(q,p) \cdot \epsilon_1$ bins and $p + \epsilon_2$ stack-up places for some $\epsilon_1, \epsilon_2 > 0$ is still open.

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 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. Storage Controlled Pile-Up Systems. *European Journal of Operational Research*, 103(3):515–530, 1997.