An Approximation Algorithm for the Stack-Up Problem*

J. Rethmann E. Wanke

University of Düsseldorf, Department of Computer Science
Universitätsstraße 1, D-40225 Düsseldorf, Germany
Email: {rethmann,wanke}@cs.uni-duesseldorf.de

Abstract

We consider the combinatorial stack-up problem motivated by stacking up bins from a conveyor onto pallets. The stack-up problem is to decide whether a given list q of labeled objects can be processed by removing step by step one of the first s objects of q so that the following holds. After each removal there are at most p labels for which the first object is already removed from q and the last object is still contained in q. We give some NP-completeness results and we introduce and analyze a polynomial time approximation algorithm for the stack-up problem.

Key words approximability, discrete algorithms, problem complexity, computational analysis

1 Introduction

Let us first motivate our research on stack-up systems. The problem of stacking up bins from a conveyor onto pallets basically appears in so-called *stack-up systems* which play an important role at delivery industry and warehouses. The customers are companies that order a large amount of articles. The articles have to be put into bins for delivery which is usually done by human workers in so-called *pick-to-belt order-picking systems*, see [4, 17] for more details. In general, each order consists of several bins. During the packing process the bins are mixed up between different orders. However, for delivery reasons, it is absolutely necessary to place all bins belonging to one costumer order onto the same pallet. Therefore, some bins are temporarily stored before they are moved onto their pallets.

In practice, the bins arrive the stack-up system on a conveyor from an order-picking system. At the end of the conveyor they enter a *cyclic storage system*, where they are moving around in a cycle. From the storage the bins are pushed out into *buffers*, where they are queued. From the end of the buffers the bins are picked-up by stacker cranes and

*A short abstract of section 5 is already published in the proceedings of WADS '97 [15].

moved onto pallets, which are located at so-called *stack-up places*. There is one buffer for each stack-up place. 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 [17].

Many details of the architecture are unimportant to compute efficiently an order in which the bins can be palletized using the available number of pallets. We model the buffers and the cyclic storage system by one random access *storage region* from which the bins can be picked-up and moved onto pallets. In real live, the buffers and the cyclic storage system are necessary to enable a smooth stack-up process irrespective of the real speed the cranes and conveyors are moving. Logistic experiences over 10 years lead to such high flexible conveyor systems in delivery industry. So we do not intend to modify the architecture of existing systems, but we try to develop efficient algorithms to control them. Figure 1.1 shows a sketch of a simplified stack-up system.

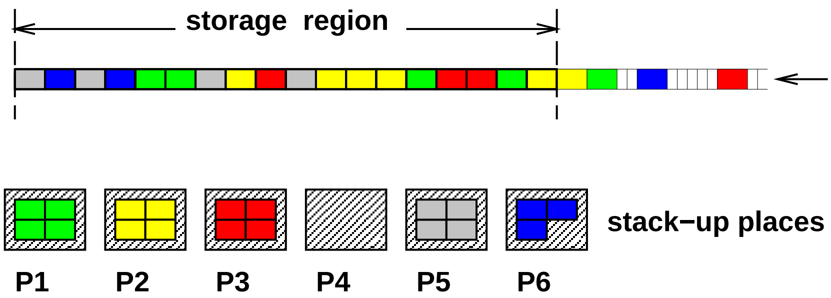

Figure 1.1: A sketch of a simplified stack-up system.

Our aim in controlling stack-up systems is to avoid blocking situations. The system is *blocked*, if all stack-up places are occupied, and the storage region is completely filled with bins not destined for the pallets on the stack-up places. To unblock the system, bins have to be picked-up manually and brought to pallets by human workers.

Our model is the first attempt to develop discrete algorithms and to capture important parameters necessary for an efficient and provable good controlling of stack-up systems. The stack-up system that has initiated our research is located at Bertelsmann Distribution GmbH in Gütersloh, Germany. On certain days, several thousands of bins are stacked-up using a storage conveyor with a capacity of approximately 60 bins, and 24 stack-up places, while at most 32 bins are destined for each pallet.

From a theoretical point of view, we are given a sequence of bins $q = (b_1, \ldots, b_n)$, and two integers s and p. Each bin b_i of sequence q is destined for exactly one pallet. The bins have to be removed step by step from the first s positions, such that after each removal at most p pallets are open. A pallet is called *open*, if not all but at least one bin for the pallet is already removed from q. If a bin is removed then all bins to the right are shifted one position to the left. Integer s represents the capacity of the storage region from which the bins can be picked-up by the stacker cranes. Integer p represents the number of available stack-up places. 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.

In recent years, scheduling of order-picking systems has received much attention [2, 8, 13]. Although the combinatorial stack-up problem has important practical applications, it seems to be not investigated by other authors up to now. However, the following facts are already known. In [17] it is shown that the stack-up decision problem is NP-complete [6], but can be solved efficiently if the storage capacity s, or the number of stack-up places p is fixed. In [16] the performances of simple stack-up algorithms are compared with optimal off-line solutions by competitive analysis [3, 5, 10]. An algorithm called *Most-Frequently* is introduced which 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 [18] it is shown that the performance of the Most-Frequently algorithm is the best that can be achieved by any deterministic *on-line* stack-up algorithm. The algorithm takes at most $2p$ stack-up places to process any (s,p)-sequence with a storage capacity of $2s$ bins. On-line stack-up algorithms do not see the whole sequence but the first s bins. Additionally, they know whether any bin they see is the last one for its destination. The Most-Frequently 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. In contrast to the previous work, we give in this paper some approximability results, and a polynomial time off-line approximation algorithm for the stack-up problem.

The paper is organized as follows. In the next section we introduce the preliminary notations for processing sequences of bins. Afterwards, we show that the stack-up decision problem remains NP-complete, even if we restrict the sequences to contain at most 9 bins for each pallet. Furthermore, we show that for any positive integers c and d there is no polynomial time approximation algorithm that yields an $(s+c, p+d)$-processing of any (s,p)-sequence, unless P = NP. In section 4 we show that the stack-up problem can be solved in polynomial time if the sequences contain at most 3 bins for each pallet. Finally, we introduce an $O(n\cdot\log(p))$-time approximation algorithm that takes at most $p+1$ stack-up places and a storage capacity of $s\cdot(1+\ln(p+1))$ bins to process (s,p)-sequences. The algorithm is not an approximation algorithm in the classical sense [7, 12] since it violates both feasibility and optimality. Nevertheless, such relaxed or multi-criteria approximation algorithms are also considered by other authors [9, 14, 19].

Acknowledgments We are very grateful to Bertelsmann Distribution GmbH in Gütersloh, Germany. They provided real data instances and made it possible to get an insight in the real problematic nature of stack-up systems.

References

[1] A.V. Aho, J.E. Hopcroft, and J.D. Ullman. *The Design and Analysis of Computer Algorithms*. Addison-Wesley Publishing Company, 1974.

[2] N. Ascheuer, M. Grötschel, and A. Abdel-Aziz Abdel-Hamid. Orderpicking in an Automatic Warehouse: Solving Online Asymetric TSPs. Technical Report 35/98, Konrad-Zuse-Zentrum für Informationstechnik, D-14195 Berlin, Germany, 1998.

[3] Alan Borodin. *On-line Computation and Competitive Analysis*. Cambridge University Press, 1998.

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

[5] A. Fiat and G.J. Woeginger, editors. *Online Algorithms: The state of the art*, volume 1442 of *Lecture Notes in Computer Science*. Springer-Verlag, 1998.

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

[7] D.S. Hochbaum. *Approximation algorithms for NP-hard problems*. PWS Publishing Company, 20 Park Plaza, Boston, MA 02116-4324, 1997.

[8] 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.

[9] J.H. Lin and J.S. Vitter. ϵ-Approximations with Minimum Packing Constraint Violation. In *Proceedings of the Annual ACM Symposium on Theory of Computing*, pages 771–781. ACM, 1992.

[10] 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.

[11] K. Mehlhorn. *Data Structures and Algorithms I: Sorting and Searching*. EATCS Monographs on Theoretical Computer Science. Springer-Verlag, 1984.

[12] C.H. Papadimitriou. *Computational Complexity*. Addison-Wesley Publishing Company, New York, 1994.

[13] H.D. Ratcliff and A.S. Rosenthal. Orderpicking in a Rectangular Warehouse: A Solvable Case of the Traveling Salesman Problem. *Operations Research*, 31:507–521, 1983.

[14] R. Ravi, M.V. Marathe, S.S. Ravi, D.J. Rosenkrantz, and H.B. Hunt. Many birds with one stone: Multi-objective approximation algorithms. In *Proceedings of the Annual ACM Symposium on Theory of Computing*, pages 438–447. ACM, 1993.

[15] 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.

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

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

[18] J. Rethmann and E. Wanke. An optimal algorithm for on-line palletizing at delivery industry. In *Proceedings of the International Symposium on Algorithms and Computation*, volume 1533 of *Lecture Notes in Computer Science*, pages 109–118. Springer-Verlag, 1998.

[19] A. Warburton. Approximation of pareto optima in multiple-objective shortest path problems. *Operations Research*, 35(1):70–79, 1987.