MINIMIZING THE MAKESPAN OF CONTAINER STORAGE AND RETRIEVAL REQUESTS

Amir Gharehgozli MSc[1], Dr. Yugang Yu[1], Prof. dr. Rene de Koster[1], Prof. dr. Jan Udding[2]

[1]Rotterdam School of Management, Erasmus University Rotterdam, the Netherlands
[2]Faculty of Mechanical Engineering, Department of Systems Engineering, Eindhoven University of Technology, the Netherlands

ABSTRACT

We sequence multiple container storage and retrieval requests in a single block located in the stack area of an automated container yard by minimizing the makespan of a single automated stacking crane (ASC) operating above the block. A number of input/output (I/O) points are located at both the seaside and the landside of the block. Every I/O point has a list of storage containers each with a given storage location. Every retrieval container can be delivered to any I/O point either at the landside or at the seaside depending on whether it is to be transported by ship or truck/train. The problem is modeled as an asymmetric travel salesman problem. We develop a two-phase solution method to optimally solve the problem.

KEYWORDS

Container storage and retrieval, automated staking crane, multiple I/O points, makespan

INTRODUCTION

Figure 1-a shows a typical layout of a container yard with several blocks in the stack area where automated stacking cranes (ASCs) store and retrieve containers. Each crane can handle containers in one block of the stack consisting of multiple rows, tiers, and bays as shown in Figure 1-b. Within one block, containers are densely stacked in piles. A number of input/output (I/O) points are located at the seaside and landside of the block. For a retrieval request, the ASC picks up a container from its location in the block and drops it in an I/O point whereas for a storage request it picks up a container from an I/O point and stores it in a location in the block.

Container terminals daily deal with a huge number of containers needed to be stored or retrieved. It is trivial that a proper method to sequence this enormous number of container storage and retrieval requests helps them to enhance their operations and decrease the

response time. On the other hand, considering its importance, the few number of surveys carried out on this topic doesn't investigate all its aspects.
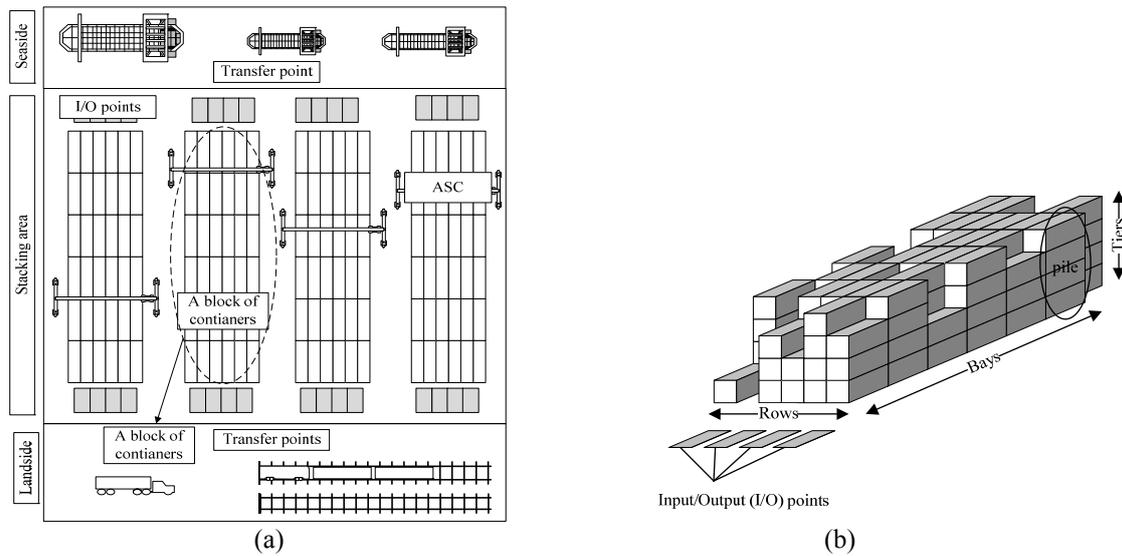

Figure 1: (a) A container yard layout, (b) a block of containers

We consider the problem of sequencing storage and retrieval requests in a single block of containers given that a single ASC operates above it. We assume a given number of N containers (export, import and transshipment) of any size (i.e. 20 or 40 feet) have to be stored and retrieved from the block. A storage container has to be picked up from a given I/O point and stored in a given storage location. A retrieval container has to be picked up from its retrieval location and delivered to either a seaside or a landside I/O point depending on whether it is to be transported by ship or truck/train. All moves are non-preemptive. Containers cannot be dropped off in a temporary location as this is time-consuming. Furthermore, all containers have the same priority to be stored or retrieved. The objective function is to minimize the makespan of the ASC to perform all requests.

The literature on this problem can be reviewed from two perspectives. Some papers directly discuss sequencing storage and retrieval requests in the storage area of a container yard or a warehouse. On the other hand, sequencing storage and retrieval requests can be discussed under a more general type of problems denoted pickup and delivery problems (PDPs). Next, we briefly discuss both streams of study.

Literature on sequencing storage and retrieval requests

Vis and Roodbergen (2009) consider scheduling storage and retrieval requests in a block of containers. They propose an exact polynomial solution method based on dynamic programming. They assume container yard uses straddle carriers (SCs) to store and retrieve containers in rows of containers with a single I/O point at each end. When a SC enters a row, all requests have to be done before it leaves the row, since changing rows is time-consuming.

Van den Berg and Gademann (1999) consider sequencing storages and retrievals in an automated storage/retrieval system (AS/RS). In an AS/RS system, an automated storage/retrieval (S/R) machine stores and retrieves unit loads. They formulate the problem as a transportation problem and claim each feasible solution of the transportation problem corresponds to a retrieval and storage sequence of the main problem. They assume a single command input point and a single command output point

Zhang et al. (2010) work on a similar problem in an AS/RS system with two I/O points. However, retrievals and storages can be performed in both I/O points. They propose an exact polynomial solution method. The method is based on the assignment problem (AP) relaxation and a subtour patching scheme.

In all the surveys mentioned above, the number of I/O point is limited. However, when the number of I/O points increases, the complexity of the problem increases exponentially, because the problem is NP-hard. Therefore, the solution method previously proposed fail.

Literature on pickup and delivery problems

Our problem can be formulated as both the asymmetric travelling salesman problem (ATSP) and the Stacker crane problem (SCP). Note that our problem has special properties that make it different than all the following studies. In our problem, every request is connected to an I/O point and retrieval requests have flexible I/O points.

The ATSP, most simply stated, is a combinatorial problem in which from a given list of cities with pairwise distances, a tour has to be determined passing every city exactly once in such a way that the total distance is minimized. Pairwise distances need not to be equal. Several algorithms are proposed to solve the ATSP. Among the most efficient ones is the depth-first branch and bound (B&B) algorithm based on the subtour elimination approach proposed by Carpaneto and Toth (1980). In general, the ATSP problem is *NP*-hard.

The SCP, in its simplest form, can be described as follows: a vehicle must start from an initial location, perform a specified set of moves, and return to its initial location. The moves are defined as carrying moves from a specified pickup point to a delivery point. The objective is to serve all required carrying moves with the shortest empty distance. This is done by adding non-carrying (empty) moves.

The SCP was originally proposed by Fredrickson et al. (1978). They prove that the SCP is *NP*-hard when the underlying network is a complete graph. Fredrickson and Guan (1993) prove that the SCP on a tree when the movements are non-preemptive is also *NP*-hard. They prove it by reducing the Steiner tree problem in bipartite graphs to the non-preemptive SCP.

PROBLEM FORMULATION

Our objective is to minimize the total makespan of the ASC for a group of storage and retrieval containers in a single block with multiple I/O points. This problem can be formulated as the ATSP. The model consists of an AP model plus a subtour elimination constraint.

The computation time of the problem mainly depends on the number of I/O points. With the increase of the number of I/O points, the computation time of the problem increases exponentially. This argument is supported by reducing the SCP with non-preemptive movements on a general graph to the problem of sequencing storage and retrieval requests.

Consider our problem in a simplified case with the following assumptions: (1) every retrieval request has to be delivered to a specific I/O point, (2) the number of storage and retrieval requests is equal to the number of I/O points, and (3) every I/O point is connected to only one retrieval or storage request. In this setting, we have a problem with some carrying moves and we need to add non-carrying moves in order to get a complete tour with the shortest empty distance. This implies that the problem is an SCP that is *NP*-hard.

SOLUTION METHOD

We propose a two-phase solution method. In the first phase of the solution, we take advantage of the properties of the problem to develop a merging algorithm to merge subtours of the AP

relaxation of the ATSP. The most important advantage of the problem is that the ASC has to deliver or pick up the container of each request to or from an I/O point. Therefore, I/O points can be used as a connection point to merge subtours. The merging algorithm runs within a polynomial time. In few cases we have to enter the second phase which is an improved version of the B&B algorithm to find the optimal solution of the problem.

NUMERICAL EXAMPLE

The preliminary results of the simulation study for different scenarios are presented in table 1. In each scenario, 100 samples of N storage and retrieval requests with their corresponding locations and I/O points are randomly generated by Monte Carlo simulation. The study is performed on a Notebook with Intel(R) Core(TM) i5 CPU, M 430 @ 2.27 GHz, 4.00 GB of RAM and the programming language is MATLAB.

Table 1: Results of the simulation study

			Number of requests	Number of retrievals	Number of storages	B&B algorithm						B&B + Merging algorithm					
						Computation time			number of nodes			Computation time			number of nodes		
Tiers	Rows	Bays				ave	max	min	ave	max	min	ave	max	min	ave	max	min
4	5	40	20	12	8	0.03	0.063	0.003	8.05	14	1	0.01	0.029	0.003	1.18	4	1
4	5	40	50	30	20	0.26	0.895	0.017	14.8	55	1	0.029	0.088	0.016	1.06	4	1
4	5	40	100	60	40	1.544	4.049	0.205	27.5	76	3	0.087	0.353	0.056	1.1	4	1
4	5	40	150	90	60	4.077	12.32	0.147	35.2	116	1	0.161	0.231	0.11	1	1	1
4	5	40	200	120	80	9.236	23.06	0.252	45.2	165	1	0.304	1.079	0.206	1.04	1	1

CONCLUSIONS

Using the proposed solution method developed based on the properties of the problem of sequencing storage and retrieval requests in a single block with a single ASC, we can efficiently solve small-size as well as large-size problems. This study can be extended by considering reshuffling or finding the storage locations of storage containers.

REFERENCES

Carpeneto, G., P. Toth (1980) Some New Branching and Bounding Criteria for the Asymmetric Travelling Salesman Problem, in: *Management Science*, 26(7), pp. 736-743.

Frederickson, G.N., D.J. Guan (1993) Nonpreemptive ensemble motion planning on a tree, in: *J. Algorithms*, 15(1), pp. 29-60.

Frederickson, G.N., M.S. Hecht, C.E. Kim (1978) Approximation Algorithms for Some Routing Problems, in: *ACM Journal of Computing*, 7(2), pp. 178-193.

van den Berg, J.P., A.J.R.M. Gademann (1999) Optimal routing in an automated storage/retrieval system with dedicated storage, in: *IIE Transactions*, 31(5), pp. 407-415.

Vis, I.F.A., K.J. Roodbergen (2009) Scheduling of Container Storage and Retrieval, in: *Operations Research*, 57(2), pp. 456-467.

Zhang, X., Y. Yu, S. van de Velde, R. de Koster (2010) Minimizing the makespan of storage and retrievals in a two-depot multi-aisle AS/RS: a solvable ATSP problem, *working paper*.