

Modeling of integrated quay cranes, yard trucks and yard cranes scheduling problem for outbound containers

Yujie Xiao^{1, a}, Yan Zheng^{2, b} and Peng Li^{2, c}

¹Jiangsu Key Laboratory of Modern Logistics, School of Marketing and Logistics Management, Nanjing University of Finance and Economics, Nanjing 210023, China;

²College of Automobile and Traffic Engineering, Nanjing Forestry University, Nanjing 210037, China.

^ayujie Xiao@njue.edu.cn, ^byzheng_x@163.com, ^cdawan007@njfu.edu.cn

Abstract. In container terminals, quay cranes, yard trucks and yard cranes are three main handling equipment and responsible for transporting inbound and outbound containers. To lower ship turnaround time, it is crucial that operations by these three handling equipment are well coordinated. Most previous studies concerned on one type of equipment independently of other handling equipment, which cannot guarantee the global optimal solution. It is necessary to develop and solve an integrated model where the interdependency of three handling equipment are highly considered. This paper studies an integrated model for scheduling quay cranes, yard trucks and yard cranes jointly. Real world constraints, such as blocking and precedence relationships between containers, are also taken into account. An iterated greedy based algorithm combined with two local search methods are designed to deal with this optimization model. The computational results show that the integrated model is solvable within the reasonable time by using the proposed solution method.

Keywords: integrated scheduling; genetic algorithm; container terminals; handling equipment.

1. Introduction

The international trade facilitated the worldwide logistics and transportation systems, especially the maritime container terminals (CTs) in the last decade. In most CTs, containers are boxes in a standard size of twenty-foot-equivalent unit and the world trade containers have grown by 7.1% in 2011[1]. At some point, the efficiency of CTs is reflected by one critical indicator—the ship turnaround time (i.e. the total time spent by a ship at berth) [2]. To lower the ship turnaround time and improve the service level of CTs, close cooperation of different handling equipment in CTs is required.

Fig. 1 shows a typical layout of CTs, which include two parts: one is a ship operation area with berths for ships to dock, and the other is the yard area for storing containers. Three types of handling equipment are deployed to transport containers. They are quay cranes (QCs), yard truck (YT) and yard cranes (YCs). Containers are sequentially handled by QCs, YTs and YCs. QCs are located along the seaside and responsible for loading/unloading containers onto/from the ship. YCs serve in the storage yard to lift/place containers from/onto YTs and stack or retrieving containers in the block. YTs are used to transport containers between QCs and YCs. For example, the process of completing the loading operation of an outbound container consists of three stages. First, a set of YCs moves inside the storage area to pick up the required containers from blocks and load them onto YTs. Then, YTs are dispatched to transport the containers to QCs. Finally, QCs load the containers onto the ships. Inbound containers are handled in the reverse order. Any delay in each stage would have an adverse influence on the overall efficiency of CTs. Therefore, it is necessary to simultaneously schedule the processes of three handling equipment in an integrated model.

Most previous studies focused on the optimization problems of single handling equipment independently of one another, which are QC scheduling problems [3-5], YT scheduling problems [6, 7] and YC scheduling problems[8-10]. A few studies concentrated on integrating QC-YT scheduling problems [2, 11, 12, 13, 14]. Traditional models didn't reflect the practical characteristic of CTs and the global optimal solution cannot be guaranteed. This paper studies an integrated scheduling problem of QCs, YTs and YCs, considering the YTs and YCs assignment, processing sequences of QCs, YTs

and YCs, precedence constraints between containers, blocking occurrence. To solve the integrated model, an iterated greedy (IG) based algorithm is developed, within which two local search methods are applied to diversify the solutions.

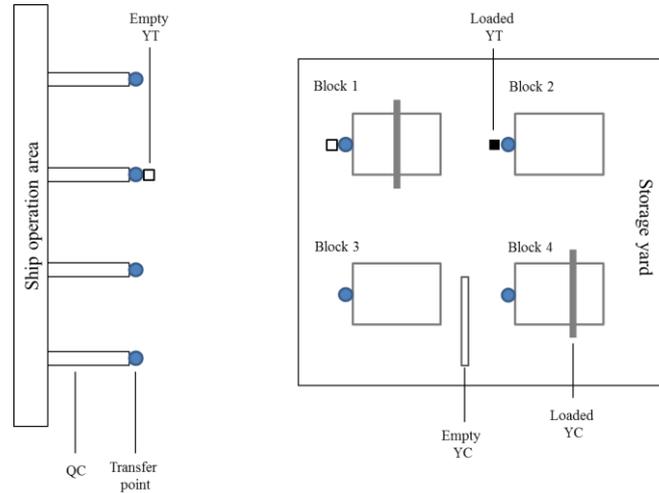


Fig. 1 A typical layout of CTs

2. Literature review

A brief review on the work of QC scheduling, YT scheduling or YC scheduling problems as well as those simultaneously considering the schedule of different handling equipment, are provided as below.

Lee et al. [3] proposed the QC scheduling problem with crane interference and developed an MILP model as well as a GA to solve the problem. The authors showed that QC scheduling problem was NP-complete. Tavakkoli-Moghaddam et al. [4] jointly studied QC scheduling and assignment problem. Meisel & Bierwirth [5] proposed a unified approach to evaluate diverse QC scheduling models and solution procedures. Bish et al. [6] studied the problem of dispatching vehicles to containers for a single ship with the objective of minimizing the makespan. Ng et al. [7] made a study on a set of trucks scheduling problem with sequence-dependent processing times and different ready times. The objective is to minimize the makespan. Ng & Mak [8] addressed the scheduling problem for a single YC so as to minimize the sum of job waiting times. Loading and unloading operations with different ready times are both considered. Jung & Kim [9] extended a single YC scheduling problem to multiple YCs scheduling problem. Loading operations are only considered with the objective of minimizing the makespan. He et al. [10] employed a hybrid GA for solving YC scheduling problem. They addressed that YCs needed to move among different blocks to fully utilize YCs.

A single scheduling problem separately from other interrelated aspects in CTs often results in poor overall performance of CTs. Integrated problems considering interrelated decision making processes in CTs have been studied by some researchers. Kaveshgar & Huynh [11] developed an integrated quay cranes and yard truck model for inbound containers by applying hybrid flow shop scheduling technique. A genetic algorithm based method is used to solve the model. In the study by Cao et al. [12], an integrated scheduling problem of YTs and YCs for loading operations of outbound containers was proposed. An MILP model and Bender's decomposition method were developed to solve the integrated model. Li & Yip [13] presented a joint planning model in CTs which simultaneously considers home berths allocation and yard planning problems. A two-stage approach consisting of two related integer programming models was developed to find good solutions. In the work of Zeng & Yang [14], an integrated simulation and optimization method for simultaneously scheduling QCs, YTs and YCs for loading operations was introduced. Chen et al. [2] presented a tabu search algorithm to deal with the integrated scheduling problem of QCs, yard vehicles and YCs. They assumed that YCs were not shared between different blocks.

3. Problem description

The proposed integrated model simultaneously considers the schedule of QCs, YTs and YCs in CTs for outbound containers, which aims to find the assignment of containers to YTs and YCs, the processing sequences of QCs, YTs and YCs in such way that the makespan is minimized, while satisfying the precedence relationships and considering the blocking situation. For outbound containers, the precedence relationships between containers require that the containers located on the hold must be handled before the ones on the deck (above the hold) when these containers share the same bay. Since there is no buffer space between cranes (QCs and YCs) and YTs, blocking situation may occur in CTs. In other words, an YT has to deliver a container to an empty truck before it continues its work. An YT can only continue to work when its container is lifted by a QC.

The proposed integrated model is different from previous studies in that YCs are shared by various blocks to increase the utilization of YCs, i.e. YCs movement is considered. The locations of QCs, QCs assignment for each container, processing time for containers in each operation, the locations of containers in terms of bay number and storage block and the moving speed of YTs/YCs are predetermined. Control problems such as traffic congestion and conflicts of YTs and YCs are not in the consideration of this study. The following assumptions are adopted when modeling the integrated model: 1) identical YTs/YCs are used in CTs. Each container can be transported by any YT/YC. Loaded YTs/YCs and empty YTs/YCs are assumed to move at the same speed; 2) YTs have the unit capacity, i.e. each YT can transfer only one twenty-foot-equivalent container at a time; 3) The handling time of QCs/YCs for containers is assumed to be the average handling time. The transfer time between cranes (QCs, YCs) and YTs is included in the handling time of cranes; 4) Special containers (e.g. containers for dangerous goods, animals or frozen items) are not included. All containers can be stacked on the top of each other.

4. Solution method

IG algorithm is a simple and very efficient meta-heuristic algorithm which was first proposed by Jacobs & Brusco [15]. The original IG algorithm starts with an initial solution S_0 and iteratively searches better solutions through a main loop which includes two phases called destruction and construction. In the destruction phase, a partial solution S_p is created by removing d elements from the current solution S , and then in the construction phase, a complete solution S_c is reconstructed by a greedy heuristic. Once S_c is completed, an acceptance criterion is used to determine whether the current solution S is replaced by S_c . Application of the local search is optional, which is helpful to the diversification of the search. This loop proceeds until the termination condition is satisfied.

The integrated scheduling problem simultaneously deals with three sub-problems: the loading sequence of each QC, YTs dispatching problem, YCs dispatching problem. Based on this problem structure, a three-dimension solution representation is designed, which links three sub-solutions via considering the interrelation of three handling equipment. To enhance the efficiency of IG in finding better solutions, two local searches (i.e. Local_search_1 & Local_search_2) are applied to YTs dispatching problem and YCs dispatching problem, respectively.

The flowchart of IG algorithm is given in Fig.2. Based on the input data, the initial solution is randomly created and evaluated. In each iteration, destruction and construction are implemented successively, after which two local search methods are applied. IG searches better solutions iteratively and stops when the termination condition is satisfied.

4.1 Solution representation.

Each solution is represented as $3 \times N$ (N is the total number of containers) array, denoted as π , which has three fixed-length dimensions: the loading sequence of jobs, named $\pi(1)$; the assignment of YTs to jobs in $\pi(1)$, named $\pi(2)$; the assignment of YCs to jobs in $\pi(1)$, named $\pi(3)$. $\pi(a)$ is defined as $\pi(a) = \{\pi(a)(1), \pi(a)(2), \dots, \pi(a)(N)\}$ ($a = 1, 2, 3$). Based on the loading sequence in $\pi(1)$, the sequences of jobs on YTs and YCs are obtained from $\pi(2)$ and $\pi(3)$, respectively. In each π , the elements in each

column demonstrate YT and YC assignments to each job and the elements in each row give the processing sequence of jobs on QCs, YTs and YCs.

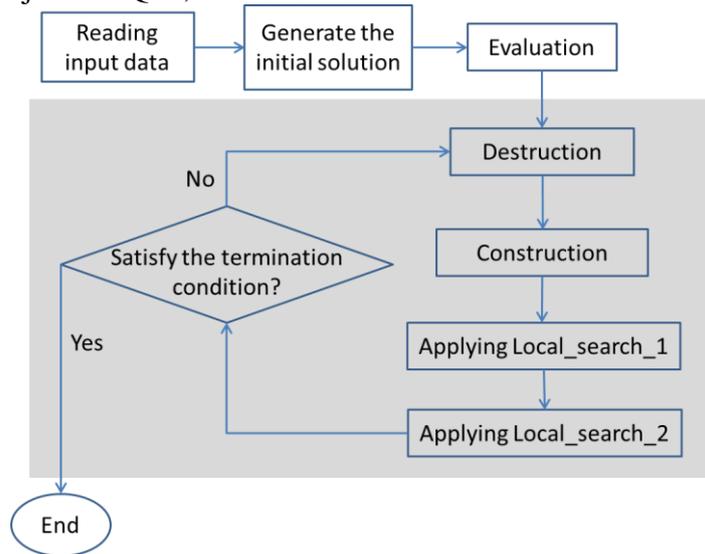


Fig. 2 The flow chart of the proposed IG algorithm

4.2 Procedure of IG algorithm

- Step 1: Initialization. An initial solution $\pi_{initial}$ is created randomly. Set the current solution $\pi = \pi_{initial}$ and the best solution $\pi_{best} = \pi_{initial}$.
- Step 2: Destruction and Construction. Select d containers from $\pi(1)$ in π . A partial solution π_p is generated by deleting the selected d containers together with its assigned YT and YC in $\pi(2)$ and $\pi(3)$ from π . Then add the d columns to πR - a list of the deleted columns, in the order they are removed. Reinsert the removed columns in πR , one by one, to all possible and feasible positions of π_p . Each time one deleted column is reinserted into the partial solution, a set of candidate partial solutions is generated. Only those satisfying the precedence relationships between containers are feasible, among which the one with minimum makespan C_{max} is selected as the new partial solution π_p . After inserting all d deleted columns, a new complete solution π_{new} is then obtained. Check if $C_{max}(\pi_{new}) \leq C_{max}(\pi)$, then $\pi = \pi_{new}$ and $\pi_{best} = \pi_{new}$; else if $C_{max}(\pi_{new}) > C_{max}(\pi)$ and $r \leq P$, then $\pi = \pi_{new}$, where r is a generated random number between $[0, 1]$, P is a parameter defined by users.
- Step 3: Local_search_1. Keep $\pi(1)$ and $\pi(3)$ of π unchanged and create a set of $N \times (V - 1)$ (V is the total number of YTs) candidate solutions by modifying YT assignment of each container one by one to any of other YTs. Among the candidate solutions, the one π_{local} with the minimum makespan C_{max} is selected. Check if $C_{max}(\pi_{local}) \leq C_{max}(\pi)$, then $\pi = \pi_{local}$ and $\pi_{best} = \pi_{local}$; else if $C_{max}(\pi_{local}) > C_{max}(\pi)$ and $r \leq P$, then $\pi = \pi_{local}$.
- Step 4: Local_search_2. Keep $\pi(1)$ and $\pi(2)$ of π unchanged and create a set of $N \times (C - 1)$ (C is the total number of YCs) candidate solutions by modifying YC assignment of each container one by one to any of other YCs. Among the candidate solutions, the one π_{local} with the minimum makespan C_{max} is selected. Check if $C_{max}(\pi_{local}) \leq C_{max}(\pi)$, then $\pi = \pi_{local}$ and $\pi_{best} = \pi_{local}$; else if $C_{max}(\pi_{local}) > C_{max}(\pi)$ and $r \leq P$, then $\pi = \pi_{local}$.
- Step 5: IG algorithm iterates through Step 2 to Step 4 until the termination condition is satisfied. When the counter of iteration n_{iter} reaches the maximum allowed number of iterations N_{iter} , IG algorithm stops.

5. Numerical experiment

To evaluate the performance of the proposed IG, seven problem instances (e1-e7) are generated based on the data from the published literature [12, 16]. In the layout of CT, there are six positions of QCs and 20 blocks in the storage yard. The assigned QC and block for each container as well as the

precedence relationship between containers is randomly generated. The initial positions of YTs and YCs are created in a random way. The moving speed of YTs and YCs is assumed to be 4m/s and 3m/s, respectively. The distance between any two locations of QCs and blocks is assumed to be along the shortest path. The average handling time of QCs and YCs are 60 seconds and 100 seconds, respectively.

The proposed IG was implemented using C language. All computational examples were tested on a PC with Inter Core 2, 3.30 GHz CPU and 4 GB RAM. Based on the preliminary experiments, the parameters selected for the computational experiments are $d = 6$, $P = 0.4$ and $N_{iter} = 500$. For each problem instance, IG is run 10 times and the best makespan together with the CPU time is reported as the final result. As shown in Table 1, the test problems are solvable by using the proposed IG efficiently.

Table 1 Computational results of IG

Problem instances	$N/Q/V/C$	IG results	
		C_{max}	CPU time (seconds)
e1	6/1/2/2	1371.33	0.02
e2	8/1/2/2	1382	0.03
e3	8/2/3/2	1460.33	0.05
e4	20/2/3/2	1565.67	0.23
e5	20/3/4/3	2166	0.36
e6	30/3/6/3	1642	1.00
e7	30/3/8/4	2097.33	1.45

N – total number of containers; Q – total number of QCs; V – total number of Yts; C – total number of YCs

6. Conclusion

This paper develops an integrated model for simultaneously scheduling QCs, YTs and YCs in CTs, which is close to the real world operation condition in CTs by considering practical constraints and the interdependency of different handling equipment. Specially, to fully utilize the limited resources YCs and overcome the workload imbalance among blocks, YCs move between different blocks. To solve this model, IG algorithm with a three dimension solution representation is developed. The experiment results show that the model is solvable using the integrated method. In future research, more practical constraints, such as the interference of handling equipment, storage allocation, the ready time of containers, will be involved into the integrated scheduling model. Moreover, the solution method will be improved by incorporating problem-matching strategies. A heuristic algorithm will be designed to generate the initial solution considering the workload of the YTs and YCs instead of in a random way. Other meta-heuristic algorithms and more large-size experiments will be developed in order to evaluate the proposed IG.

Acknowledgements

The research is supported by the National Natural Science Foundation of China (Grant No. 71501090, 61403204), the Natural Science Foundation of Jiangsu Higher Education Institutions of China (Grant No. 14KJD410001).

References

- [1] UNCTAD. Review of Maritime Transport. New York and Geneva: United Nations. 2012.
- [2] Chen, L., Bostel, N., Dejax, P., Cai, J. and Xi, L. A tabu search algorithm for the integrated scheduling problem of container handling systems in a maritime terminal. European Journal of Operational Research. Vol. 181 (2007) No.1, p. 40-58.

- [3] Lee, D.H., Wang, H.Q., Miao, L. Quay crane scheduling with non-interference constraints in port container terminals. *Transportation Research Part E: Logistics and Transportation Review*. Vol. 44(2008) No. 1, p. 124-135.
- [4] Tacakkoli-Moghaddam, R., Makui, A., Salahi, S., Bazzazi, M. and Taheri, F. An efficient algorithm for solving a new mathematical model for a quay crane scheduling problem in container ports. *Computers & Industrial Engineering*, Vol. 56(2009) No. 1, p. 241-248.
- [5] Meisel, F. and Bierwirth, C. A unified approach for the evaluation of quay crane scheduling models and algorithms. *Computers & Operations Research*. Vol. 38(2011), p. 683-693.
- [6] Bish, E.K., Chen, F.Y., Leong, Y.T., Nelson, B.L., Ng, J.W.C., Simchi-Levi, D. Dispatching vehicles in a mega container terminal. *OR Spectrum*. Vol. 27(2005), p. 491-506.
- [7] Ng, W.C., Mak, K.L., Zhang, Y.X. Scheduling trucks in container terminals using a genetic algorithm. *Engineering Optimization*. Vol. 39(2007) No. 1, p. 33-47.
- [8] Ng, W.C., Mak, K.L. Yard crane scheduling in port container terminals. *Applied Mathematical Modeling*. Vol. 29(2005), p. 263-276.
- [9] Jung, S.H., Kim, K.H. Loading scheduling for multiple quay cranes in port container terminals. *Journal of Intelligent Manufacturing*. Vol. 17(2006), p. 479-492.
- [10] He, J., Chang, D., Mi, W., Yan, W. A hybrid parallel genetic algorithm for yard crane scheduling. *Transportation Research Part E: Logistics and Transportation Review*. Vol. 46(2010) No.1, p. 136-155.
- [11] Kaveshgar, N., Huynh, N. Integrated quay crane and yard truck scheduling for unloading inbound containers. *International Journal of Production Economics*. Vol. 159 (2015), p. 168-177.
- [12] Cao, J.X., Lee, D-H., Chen, J.H., Shi, Q. The integrated yard truck and yard crane scheduling problem: Benders' decomposition-based methods. *Transportation Research Part E: Logistics and Transportation Review*. Vol. 46(2010) No.3, p. 344-353.
- [13] Li, M.K., Yip, T.L. Joint planning for yard storage space and home berths in container terminals. *International Journal of Production Research*. Vol. 51(2013) No.10, p. 3143-3155.
- [14] Zeng, Q., Yang, Z. Integrating simulation and optimization to schedule loading operations in container terminals. *Computers & Operations Research*. Vol. 36(2009), p. 1935-1944.
- [15] Jacobs, L.W., Brusco, M.J. A local search heuristic for large set covering problems. *Naval Research Logistics*. Vol. 42(1995) No.7, p. 1129-1140.
- [16] Lau, H.Y.K., Zhao, Y. Integrated scheduling of handling equipment at automated container terminals. *Annals of Operations Research*. Vol. 159(2008), p. 373-394.