



ARTICLE

Optimal Scheduling of Multiple Rail Cranes in Rail Stations with Interference Crane Areas

Nguyen Vu Anh Duy, Nguyen Le Thai and Nguyen Huu Tho*

Faculty of Engineering and Technology, Nguyen Tat Thanh University, Ho Chi Minh City, 700000, Vietnam

*Corresponding Author: Nguyen Huu Tho. Email: nhtho@ntt.edu.vn

Received: 05 December 2022 Accepted: 12 April 2023 Published: 29 March 2024

ABSTRACT

In this paper, we consider a multi-crane scheduling problem in rail stations because their operations directly influence the throughput of the rail stations. In particular, the job is not only assigned to cranes but also the job sequencing is implemented for each crane to minimize the makespan of cranes. A dual cycle of cranes is used to minimize the number of working cycles of cranes. The rail crane scheduling problems in this study are based on the movement of containers. We consider not only the gantry moves, but also the trolley moves as well as the re-handle cases are also included. A mathematical model of multi-crane scheduling is developed. The traditional and parallel simulated annealing (SA) are adapted to determine the optimal scheduling solutions. Numerical examples are conducted to evaluate the applicability of the proposed algorithms. Verification of the proposed parallel SA is done by comparing it to existing previous works. Results of numerical computation highlighted that the parallel SA algorithm outperformed the SA and gave better solutions than other considered algorithms.

KEYWORDS

Multi-crane scheduling; logistics; containers; makespan; rail stations

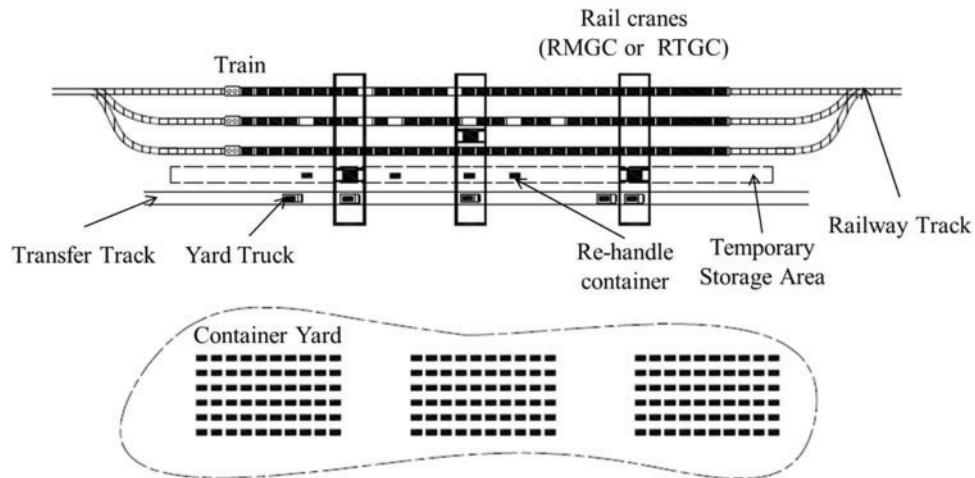
1 Introduction

Currently, air pollution is becoming more critical than ever. Rail transportation is an environmentally friendly solution because of low CO₂ emission, and will be further developed in the near future. In a rail transportation system, the main facility is the rail station. It is the place for trains to be discharged/loaded with outbound/inbound containers by rail cranes (RMGC, RTGC), forklifts and reach stackers. Fig. 1 shows the general layout of a rail station. In this paper, we consider the scheduling of the rail cranes because their operations directly influence the throughput of the rail stations. We not only assigned the job to cranes but also gave the job sequencing for each crane in order to minimize the makespan (the maximum completion time) of cranes. A dual cycle of crane is selected to minimize the number of crane working cycles.

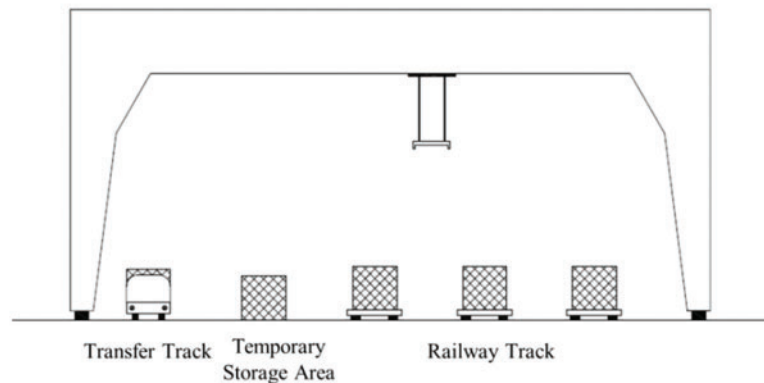
In a working cycle, a crane simultaneously loads and discharges inbound container and outbound container on the trains, respectively. First, the crane moves to the transfer track, picks up the inbound container from the truck and loads it on to the train. Then, the crane moves to the outbound container position, discharges it from the train and loads it on to the truck. The re-handling of container (double



handling) happened when an inbound container needs loading on the train but the outbound container at that position has not been unloaded. In that case, the inbound container needs to be loaded to a temporary storage area and be loaded on the train immediately after the outbound container is unloaded.



(a) Bird's eyes view



(b) Cross section

Figure 1: Layout of a rail station

One difficulty of this problem is re-handling. An inbound container is directly loaded onto a train if the wagon is empty; otherwise, the inbound container needs to be re-handled. In that instance, the inbound container is unloaded from a truck and stacked in a temporary storage area. The inbound container that is stacked in the temporary storage area will be loaded onto the train after the wagon is empty (or the outbound container is discharged). In that case, we say that the outbound needs re-handling. Obviously, the re-handling operations of containers (inbound containers or outbound containers) depend on the point of time that the containers are transferred by cranes. In the simplest case, the re-handling case is caused by the operation of a crane, or may unfortunately be caused by two cranes being operated. A crane loads an inbound container on a wagon but that wagon has yet to

be discharged, meaning the inbound container needs to be stored in a temporary storage area. Later, another crane unloads the wagon, and it should load the inbound container onto the wagon.

This paper is organized as follows. [Section 2](#) presents the literature review while [Section 3](#) explains the mathematical model. In [Section 4](#), the Simulated Annealing algorithms are developed to get near optimal solutions. [Section 5](#) presents the performance validation of the algorithms and finally, [Section 6](#) presents the conclusions of this study.

2 Literature Review

There are numerous studies related to the operations in rail terminals. Research related to the operation problems in a rail station can be classified into three sub-problems of layout design, load planning, and rail crane scheduling. There are some studies related to the layout design of a rail terminal. Kozan [1] determined the crane split and assignment of trains to rail way tracks by a simple heuristic decision rule and a simple dispatching rule. In addition, a simulation was applied based on these heuristic rules to compare different yard layouts. Ballis et al. [2] identified the main design parameters of a railroad transport terminal such as the length and utilization of trans-shipment tracks, the mean stacking height in a storage area, and the type and amount of handling equipment by an analysis tool consisting of three modules comprising of an expert system, a simulation model, and a cost calculation module. Abacoumkin et al. [3] have proposed a similar approach to Ballis et al. [2], where they developed an expert system to define the terminal parameters such as cargo volume, loading unit mixture, equipment, layout characteristic, and operation conditions. Kozan [4] used a simulation model to compare different combinations of handling equipment to determine the most efficient set of handling equipment which is balanced between investigation cost and performance. Benna et al. [5] presented a simulation tool to plan and to design for the infrastructure, capacity, layout, and strategies of railroad container terminals.

There are several papers on the load planning on trains. For instance, Feo et al. [6] optimized the assignments of highway trailer to wagon in a piggyback (trailer on flatcar) system. Bostel et al. [7] were interested in the loading of containers on trains at a rail transfer terminal. Their aim was to minimize the displacement of containers. In contrast to Bostel et al. [7], Corry et al. [8] considered a different terminal system, different objectives and constraints. Their study proposes several techniques for determining an appropriate balance between container handling and weight distribution at a railroad transfer terminal. In Bruns et al. [9], their objective was to assign containers to the wagon of a train to maximize the weight of load units and minimize the travelling cost of load units from their current storage location to the assigned location.

Similarly, there are several papers related to the rail crane scheduling problem. For instance, Aliche [10] delved into logistical issues in container transfer at a theoretical intermodal terminal “Mega Hub” based on constraint satisfaction modeling and heuristics to minimize the maximum lateness of all trains. Boysen et al. [11] investigated the static crane areas in rail—road transshipment yards. In contrast, Boysen et al. [12] focused on the optimization of static crane areas in rail—rail transshipment contexts, using dynamic programming for optimal work area allocation. However, the optimal job sequencing of rail cranes remained unaddressed. Interestingly, none of these studies considered the efficiency-enhancing dual-cycle operation for rail cranes. Jeong et al. [13] demonstrated the application of dual-cycle operations for sequencing rail crane jobs and truck positions, but limited their model to unidirectional crane movement. Pap et al. [14] proposed a branch-and-bound algorithm for optimized dual-cycle job sequencing of a rail cranes, but their work excluded container re-handling scenarios. Guo et al. [15] tackled container loading/unloading scheduling with multiple gantry crane

constraints, addressing safety clearance and non-crossing constraints. Their work, however, assumed one-dimensional crane travel and neglected dual-cycle operations or container re-handling. In contrast to these approaches, Nguyen et al. [16] studied rail crane job scheduling specifically in the context of container transfers between trains and trucks, incorporating both re-handling and dual-cycle operations into their model.

Scheduling problems of rail cranes in a rail station are similar to the scheduling problems of quay cranes at sea port container terminals. Many researchers have studied quay crane scheduling problems (QCSP) at port container terminals. In the quay crane scheduling problem, there are several factors which must be considered which are the bays, stacks, groups of containers and containers. However, the majority of studies have focused on the QCSP of container groups or vessel bays. Kim et al. [17] considered the sequence problem of discharging and loading groups of containers of quay cranes in order to minimize the completion time of a ship operation with the interference constraints of cranes. They proposed a branch-and-bound algorithm, and a greedy randomized adaptive search procedure (GRASP) to solve the problem. Moccia et al. [18] revised the formulation in Kim et al. [17], and obtained the optimal solution of the QCSP in small and medium sized instances. A branch-and-cut algorithm was developed to achieve the solutions of larger-sized instances. Moreover, some later studies including those of (Kaveshgar et al. [19]; Lee et al. [20]; Sammarra et al. [21]) tried to revise several issues in the formulation of Kim et al. [17] and applied novel approaches towards problem solving. Ng et al. [22] formulated the quay crane scheduling problem through integer programming. They developed an algorithm to obtain the lower bound of the problem based on the job scheduling problem for parallel machines with sequence independent processing time and LPT (largest processing time) rule. In order to find near optimal solutions, they developed a scheduling heuristic by decomposing the original problem into a number of independent quay crane scheduling sub-problems. The ship was partitioned into zones, with each crane operating within an individual zone. Dynamic programming was used to determine the optimal partition. Meisel [23] considered the QCSP with the inclusion of another aspect, in which a crane at a vessel is only available within certain time windows. A mixed integer programming model as well as a lower bound was developed. A heuristic approach was proposed to minimize the makespan of the cranes. Chen et al. [24] proposed a multi-commodity network flow model with two sets of flow trade-off constraints for multiple cranes and automated guide vehicles. Their constraints were used to solve inter-robot constraints to show the complicated interactions among agents accurately. Yang et al. [25] presented an integrated optimization method to manage the multi-devices integrated scheduling and storage space assignment issue in an energy-efficient manner. A bi-objective model is suggested to minimize the whole operating time and energy consumption, in which the transporting operations of imported and exported intermodal containers were solved simultaneously. Similarly, Li et al. [26] conducted a simulation-based solution for a multiple cranes scheduling issue in a steelmaking shop. Crane scheduling issues for a novel kind of automated container terminal system based on multi-storey frame bridges were explored by Zhen et al. [27]. Schulz et al. [28] depicted the planning situation encountered at a large rail-road terminal in Germany and presented a mixed-integer model capturing the core of the decision-making. They carried out a complicated analysis of issue's variants and developed a heuristic method in a simulated annealing framework based on these structural insights. Moreover, recent studies on railway container terminal have shown the ongoing interests of various researchers for allocation, planning and scheduling problems [29,30].

In summary, the rail crane scheduling problems in this study are based on the movement of containers. We consider not only the gantry moves, but also the trolley moves. Moreover, the re-handle cases are also considered. In contrast, many studies of QCSP have focused on the operation of quay

cranes based on the groups or bays of containers. The movements of containers are usually decided by the crane drivers' experiences, and hence, reports on QCSP usually only focus on the movement of the gantry. The multi-crane scheduling problem in rail stations is NP-hard, so it is meaningful to propose intelligent optimization algorithm to solve this problem based on parallel simulated annealing.

3 Mathematical Model

In this section, the mathematical model is developed.

Notation

T	Number of trains ($T + 1$ is the temporary storage area position; and $T + 2$ is the transfer track position)
W	Number of wagons per train
i^x, j^x	Wagon indexes
i^y, j^y	Train indexes
i, j	An outbound or an inbound container index, respectively, that is located at position $(i^x, i^y), (j^x, j^y)$
t_x	Travelling time of a crane between two adjacent wagons
t_y	Travelling time of a crane between two adjacent railway tracks
I	Set of inbound containers
O	Set of outbound containers
I'	Set of inbound containers that have the same position as outbound containers $I' = \{i \in I: \exists j \in O \text{ and } i^x = j^x \text{ and } i^y = j^y\}$
O'	Set of outbound containers that have the same position as inbound containers $O' = \{i \in O: \exists j \in I \text{ and } i^x = j^x \text{ and } i^y = j^y\}$
M	A large enough constant
$d_{x,y}^{x',y'}$	Travelling time of a crane from position (x', y') to position (x, y) , determined based on "Tchebychev distance" as follows. $d_{x,y}^{x',y'} = \text{Max} [x - x' t_x, y - y' t_y]$
D_{ij}	The time require to complete the loading or unloading job of container j if container j is performed after container i

Decision variable

X_{ij}^k	1, if container i is handled immediately before container j by crane k ; 0, otherwise
------------	---

Derived variable

C^k	Completion time of crane k
C_i	Completion time of the loading or unloading job of container i
C_{\max}	Makespan of all cranes
T_{Ii}^k	Additional handling time of crane k if inbound container i needs to be re-handled
T_{Oo}^k	Additional handling time of crane k if outbound container i needs to be re-handled
TR^k	Total re-handling time of crane k
Z_{ij}	1, if container j starts to be handled after container i is completely handled; 0, otherwise
R_i	1, if container i needs to be re-handled; 0, otherwise
p_i	The time required to perform container i

$$\text{Min}[C_{\max}] \tag{1}$$

Subject to

$$C_i - C^k \leq (1 - X_{iE_k}^k) M \forall k \in K, i \in I \tag{2}$$

$$C_i + R_i \times d_{i^x, T+2}^{i^x, i^y} - C^k \leq (1 - X_{iE_k}^k) M \forall k \in K, i \in O \tag{3}$$

$$C_{\max} \geq C^k + \sum_{i \in I} T_{li}^k + \sum_{i \in O} T_{oi}^k \forall k \in K \tag{4}$$

$$p_i = \begin{cases} t_y (Tr + 2 - i^y) + R_i t_y (Tr + 2 - i^y) \forall i \in O \\ R_i t_y + (1 - R_i) t_y (Tr + 2 - i^y) \forall i \in I \end{cases} \tag{5}$$

$$-MZ_{ij} \leq C_i - C_j + p_j < M(1 - Z_{ij}) \forall i, j \in I \cup O \tag{6}$$

$$\sum_{v=1}^k \sum_{u \in I \cup O} X_{uj}^v - \sum_{v=1}^k \sum_{u \in I \cup O} X_{ui}^v \leq M(Z_{ij} + Z_{ji}) \forall i, j \in I \cup O, i^x < j^x, k \in K \tag{7}$$

$$\sum_{k \in K} \sum_{i \in I \cup O} X_{ij}^k = 1 \forall j \in I \cup O \tag{8}$$

$$\sum_{i \in I \cup O} X_{ij}^k - \sum_{i \in I \cup O} X_{ji}^k = 0 \forall k \in K, j \in I \cup O \tag{9}$$

$$\sum_{i \in I \cup O} X_{0ki}^k = 1 \forall k \in K \tag{10}$$

$$\sum_{i \in I \cup O} X_{iE_k}^k = 1 \forall k \in K \tag{11}$$

$$C_j - C_i \leq R_i M + |x(i) - x(j)| M + |y(i) - y(j)| M \forall i \in I', j \in O' \tag{12}$$

$$|R_i - R_j| \leq |x(i) - x(j)| M + |y(i) - y(j)| M \forall i \in I', j \in O' \tag{13}$$

$$R_i = 0 \forall i \in I \setminus I' \tag{14}$$

$$D_{ij} + C_i - C_j \leq (1 - X_{ij}^k) M \forall k \in K \tag{15}$$

$$\sum_{j \in O} X_{ij}^k (d_{j^x, T+1}^{i^x, T+1} - d_{j^x, j^y}^{i^x, i^y} - d_{i^x, i^y}^{i^x, T+2} + d_{i^x, T+1}^{i^x, T+2})$$

$$+ \sum_{j \in I} X_{ij}^k (d_{j^x, T+2}^{i^x, T+1} - d_{j^x, T+2}^{i^x, i^y} - d_{i^x, i^y}^{i^x, T+2} + d_{i^x, T+1}^{i^x, T+2}) - T_{li}^k \leq (1 - R_i) M$$

$$\forall k \in K, i \in I \tag{16}$$

$$\begin{aligned} & \sum_{j \in O} X_{ij}^k (d_{j^x, j^y}^{i^x, i^y} - d_{j^x, j^y}^{i^x, T+2} + d_{i^x, T+2}^{i^x, i^y}) \\ & + \sum_{j \in I} X_{ij}^k (d_{j^x, T+2}^{i^x, i^y} - d_{j^x, T+2}^{i^x, T+2} + d_{i^x, T+2}^{i^x, i^y}) - T_{O_i}^k \leq (1 - R_i) M \\ \forall k \in K, i \in O \end{aligned} \quad (17)$$

$$T_{ii}^k \geq (0 - R_i) M \quad \forall k \in K, i \in I \quad (18)$$

$$T_{0i}^k \geq (0 - R_i) M \quad \forall k \in K, i \in O \quad (19)$$

$$d_{i^x, T+2}^{0^{kx}, 0^{ky}} + d_{i^x, i^y}^{i^x, T+2} - C_i < (1 - X_{0i}^k) M \quad \forall k \in K, i \in I \quad (20)$$

$$d_{i^x, i^y}^{0^{kx}, 0^{ky}} + d_{i^x, T+2}^{i^x, i^y} - C_i < (1 - X_{0i}^k) M \quad \forall k \in K, i \in O \quad (21)$$

$$X_{ij}^k, R_i, Z_{ij} = 0 \text{ or } 1 \quad \forall k \in K, i, j \in I \cup O \quad (22)$$

The mathematical model is defined in the objective function as Eq. (1). The makespan is the maximum completion time of all cranes, as described in Eq. (4). The completion time of a crane is the completion time of the last container in its container sequence, as in Eqs. (2) and (3). Eq. (5) is the processing time of a container. Eq. (6) defines the variable Z_{ij} if container j is handled after the operation of container i is completed, $Z_{ij} = 1$; otherwise, $Z_{ij} = 0$. Eq. (7) is noninterference constraint. Eq. (8) ensures that a container is assigned to only one crane. Eq. (9) is the flow balance constraint. Eqs. (10) and (11) select the initial and final containers of a crane, respectively. Eqs. (12) to (14) evaluate the re-handling cases. Eq. (12) denotes that if an inbound container $i \in I$ is loaded onto a wagon before the unloading of the outbound container j , $C_i \leq C_j$, $j \in O$ and $i^x = j^x$ and $i^y = j^y$, the container i needs to be loaded to the temporary storage area, $R_i = 1$. If the inbound container i and the outbound container have the same position on a train, $R_i = R_j$ (Eq. (13)). If the inbound container i is loaded into temporary storage area, after unloading the outbound container j , the crane should load the inbound container i onto the wagon. Eq. (14) guarantees that the inbound container, which is not occupied by an outbound container, is not loaded into the temporary storage area. Eq. (15) determines the finishing time of a container. Eqs. (16) to (19) define the additional time if a container needs to be re-handled. Eqs. (20) and (21) determine the completion time of the first container in the schedule of a crane. The value of D_{ij} is calculated by Eq. (23).

$$D_{ij} = \begin{cases} d_{j^x, T+2}^{i^x, i^y} + d_{j^x, j^y}^{i^x, T+2} \quad \forall i, j \in I \\ d_{j^x, j^y}^{i^x, T+2} + d_{j^x, T+2}^{i^x, i^y} \quad \forall i, j \in O \\ d_{j^x, j^y}^{i^x, i^y} + d_{j^x, T+2}^{i^x, j^y} \quad \forall i \in I, j \in O \\ d_{j^x, T+2}^{i^x, T+2} + d_{j^x, j^y}^{i^x, T+2} \quad \forall i \in O, j \in I \end{cases} \quad (23)$$

4 Simulated Annealing Algorithms

The meta-heuristic simulated annealing algorithm is used to solve the problem of multiple rail cranes scheduling.

4.1 The Traditional SA

4.1.1 Initial Solution

A solution to the scheduling problem is a permutation representation of the job sequence of each crane. The initial solution of the SA is obtained by assigning containers to cranes based on the balancing workload among cranes. The “workload” is the total of all the travelling times to unload outbound containers or to load inbound containers [11]. One time unit is applied for the travelling time between the two closest railway tracks. The temporary storage area serves as a buffer zone between the transfer track and the railway track. Table 1 presents examples of containers. The table also includes the total crane travel time required to load or unload these containers. In the table, the train and wagon columns show the positions of containers. The total workload requires 41 units of time. Hence, the first crane is assigned wagons 1 to 5. The second crane is assigned wagons 6 to 10. The first crane has a total workload of 20 units. The workload of the second crane is 21 units.

Table 1: Examples of a job scheduling problem

Container	Type	Train	Wagon	Workload	Container	Type	Train	Wagon	Workload
1	Inbound	1	1	3	9	Inbound	1	9	3
2	Inbound	2	1	2	10	Outbound	2	1	2
3	Inbound	1	2	3	11	Outbound	1	2	3
4	Inbound	2	3	2	12	Outbound	1	6	3
5	Inbound	1	5	3	13	Outbound	2	7	2
6	Inbound	2	5	2	14	Outbound	2	8	2
7	Inbound	1	6	3	15	Outbound	1	9	3
8	Inbound	1	7	3	16	Outbound	2	9	2

The job sequence comes from a simple heuristic, called “unidirectional” operating, in which a crane will handle its job in a single direction.

4.1.2 Neighborhood Solution

Two methods are used to generate the neighborhood solutions in this simulated annealing algorithm. The first method is pairwise interchange two containers that are assigned to the same crane. The second method is to exchange the assignment of a container. In the second method, two adjacent cranes are selected randomly, assume that two cranes k_n and k_{n+1} are selected. Crane k_n has a job sequence $\{s_1^n, s_2^n, \dots, s_m^n\}$ and crane k_{n+1} has a job sequence $\{s_1^{n+1}, s_2^{n+1}, \dots, s_{m'}^{n+1}\}$. There are two cases of the exchange of assignment of containers. Case 1: The last task in the job sequence of crane k_n is assigned to crane k_{n+1} as the first task, $\{s_m^n, s_1^{n+1}, s_2^{n+1}, \dots, s_{m'}^{n+1}\}$. Case 2: The first task in the job sequence of crane k_{n+1} is assigned to crane k_n as the last task, $\{s_1^n, s_2^n, \dots, s_m^n, s_1^{n+1}\}$. These two cases are applied with probability 0.5. Overall, the first method and the second method are selected with probability α , if $\text{rand}(0, 1) \leq \alpha$ the first method is used, otherwise the second method are applied, where $\text{rand}(0, 1)$ is random number between 0 and 1.

4.1.3 The Acceptance Criterion

The neighbor solution is accepted if $Q(R) \leq Q(C)$ or $\text{rand}(0, 1) < e^{\frac{Q(C)-Q(R)}{T_i}}$, where $Q(*)$ is the completion time of crane k in solution $(*)$, T_i is the current temperature of iteration i .

4.1.4 The Cooling Down Function

In this SA, the geometric update scheme of (Lundy et al. [31]) is referred.

$$T_i = T_{i-1} \times \pi \tag{24}$$

and the algorithm stops when the computation exceeds an expected time.

4.2 The Parallel SA

Parallel computing is a technique to increase computation speed of a computer. In this algorithm the searching neighborhood solutions is separated into many threads of the CPU, which search the solution simultaneously. Hence it can search more solution candidates than the traditional SA within a similar time frame. All operators in this algorithm are the same as those of the traditional SA. Fig. 2 shows the flowchart of the algorithm. In the parallel SA, the neighborhood solutions are generated from the same current solution by many threads simultaneously. If a thread can achieve a new current solution or a new best solution, it will update to the current solution or the global best solution. All the operators of the parallel SA are the same as the operators of the SA.

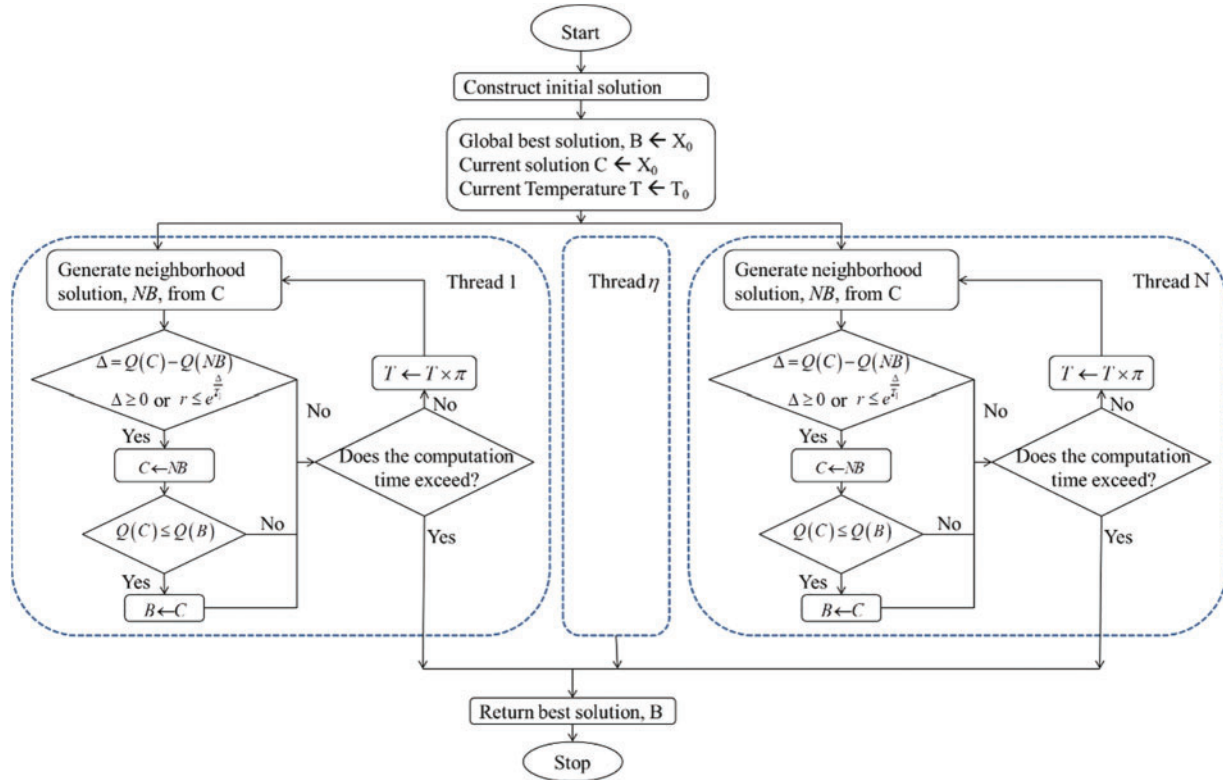


Figure 2: Parallel SA flowchart

5 Numerical Examples

It is very difficult to obtain the optimal solutions from the mathematical model. Fig. 3 shows the computation of CPLEX 12.6 on Windows 10 64 bits, Intel core i7, RAM 24 GB, SSD 512 GB to get the optimal solution from the mathematical model. From Fig. 3, the computation time increases rapidly when the number of containers increases. The computation time is very long because CPLEX used

an exact algorithm and the multi-crane scheduling problem in rail stations is NP-hard. Sometimes it is difficult to find the optimum solution with large problem size. For instance, when the number of containers is 10, we cannot achieve the optimal solution within 12 h. With the case of 10 containers and 2 cranes, it takes a very long computational time (more than 16 h) to find the optimal solution.

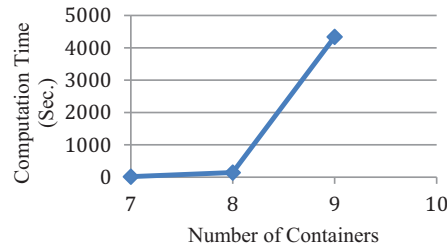


Figure 3: Computation time of mathematical model

5.1 Determine Parameter of the Simulated Annealing Algorithm

In this Section, we use the method of Ruiz et al. [32] to determine the parameters of these algorithms. The simulated annealing algorithms in this research have three parameters, α , T and π . We set four values for α , {0.2, 0.4, 0.6, 0.8}, seven values for T , {1, 5, 10, 20, 40, 80, 160} and four values for π , {0.9999000, 0.9999900, 0.9999990, 0.9999999}.

There are 112 combinations of three parameters. The algorithm is tested with 24 problems, which are generated randomly by Bernoulli distribution. The probabilities that a wagon carries a container are {0.25, 0.50, 0.75, 0.95}. The number of the trains are {1, 2, 3} and the number of the rail cranes are {2, 3}. A problem is generated twice for the inbound containers and outbound containers. The response variable is calculated by relative percentage deviation (RPD) (Ruiz et al. [32]).

$$RPD = \frac{M_{sol} - Best_{sol}}{Best_{sol}} \times 100 \quad (25)$$

where, M_{sol} is the objective value of a given problem obtained by a combination of the three parameters, $Best_{sol}$ is the best solution returned by all combinations of all parameters for the same problem. The algorithms are run by setting the computation time:

$$CpuTime = \text{NumberOfInbounds} \times \text{NumberOfOutbounds} \times \text{NumberOfCranes} \times 20 \text{ (Milliseonds)}$$

5.2 Determine Parameter of the SA

Table 2 shows the ANOVA table of RPD vs. all the parameters. The result in Table 2 shows that all parameters are very significant. In Fig. 4, the mean plot of the three parameters is presented. From this figure, the set of parameters giving the best performance is $\alpha = 0.2$, $T = 5$ and $\pi = 0.999999$.

Table 2: ANOVA: RPD vs. No., α , T and π

Source	DF	Seq SS	Adj SS	Adj MS	F	P
No.	23	9577.29	9577.29	416.40	225.18	0.000
α	3	179.08	179.08	59.69	32.28	0.000

(Continued)

Table 2 (continued)

Source	DF	Seq SS	Adj SS	Adj MS	F	P
T_o	6	4802.97	4802.97	800.50	432.89	0.000
π	3	135.69	135.69	45.23	24.46	0.000
$N_o \times \alpha$	69	457.92	457.92	6.64	3.59	0.000
$N_o \times T_o$	138	4856.63	4856.63	35.19	19.03	0.000
$N_o \times \pi$	69	343.01	343.01	4.97	2.69	0.000
$\alpha \times T_o$	18	183.29	183.29	10.18	5.51	0.000
$\alpha \times \pi$	9	53.61	53.61	5.96	3.22	0.001
$T_o \times \pi$	18	247.59	247.59	13.75	7.44	0.000
Error	2331	4310.43	4310.43	1.85		
Total	2687	25147.52				

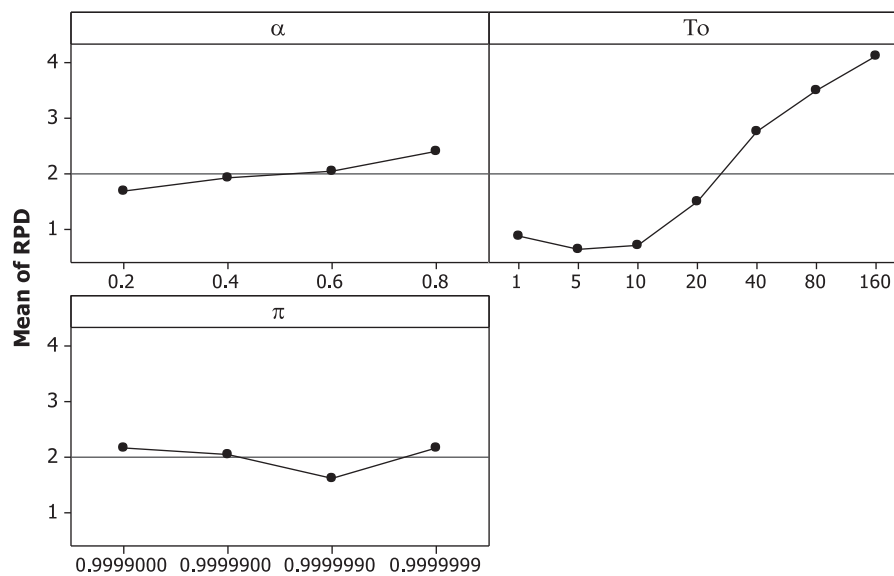


Figure 4: Mean of RPD of the SA

5.3 Determine Parameter of the PSA

Table 3 shows the ANOVA results of three parameters, showing all the factors are significant. From the Fig. 5, the set of parameters giving the best results is $\alpha = 0.2$, $T = 10$ and $\pi = 0.999999$.

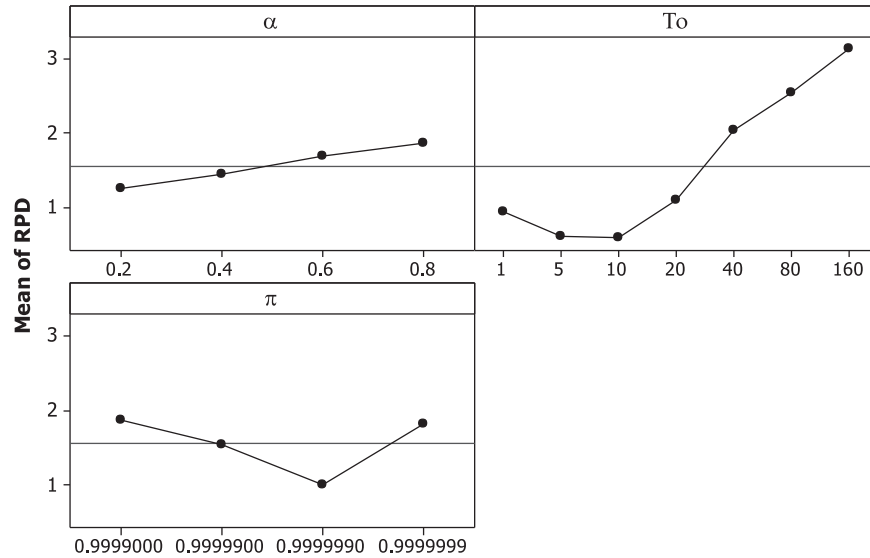
Table 3: ANOVA: RPD vs. No., α , T and π

Source	DF	Seq SS	Adj SS	Adj MS	F	P
No.	23	6773.15	6773.15	294.48	196.29	0.000
α	3	145.82	145.82	48.61	32.40	0.000

(Continued)

Table 3 (continued)

Source	DF	Seq SS	Adj SS	Adj MS	F	P
T_o	6	2376.76	2376.76	396.13	264.04	0.000
π	3	315.24	315.24	105.08	70.04	0.000
$N_o \times \alpha$	69	394.85	394.85	5.72	3.81	0.000
$N_o \times T_o$	138	2953.83	2953.83	21.40	14.27	0.000
$N_o \times \pi$	69	559.22	559.22	8.10	5.40	0.000
$\alpha \times T_o$	18	179.10	179.10	9.95	6.63	0.000
$\alpha \times \pi$	9	33.84	33.84	3.76	2.51	0.007
$T_o \times \pi$	18	397.27	397.27	22.07	14.71	0.000
Error	2331	3497.15	3497.15	1.50		
Total	2687	17626.23				

**Figure 5:** Mean of RPD of the PSA

5.4 Performances of the Algorithms

Table 4 shows the results of the SA and PSA. 24 problems are used to test the algorithms. Each problem is executed 10 times to obtain the mean and standard deviation. The Discrete Artificial Bee Colony algorithm (DABC) in Guo et al. [15] and the Greedy Randomized Adaptive Search Procedure (GRASP) in Kim et al. [17] are developed to compare to our proposed algorithms. All the algorithms are coded in JAVA programming language. From Table 4, the SA and PSA outperform the DABC and GRASP. In the small size problem, the gap between our algorithms and the other algorithms is small. When the sizes of the problems increase, the gaps among the algorithms are increased. The PSA outperforms the SA because the PSA can search for more solution candidates than the SA for a given time period.

Table 4: Experimental results of simulated annealing algorithms

No.	Number of Crane/ Trains/Inbound/ Outbound Containers	DABC (Guo et al. [15])			GRASP (Kim et al. [17])			Simulated annealing algorithm				
		Makespan		CPU time (s)	Makespan		CPU time (s)	Traditional SA		Parallel SA		
		Mean	(Std.)		Mean	(Std.)		Mean	(Std.)	Mean	(Std.)	
1	2/1/12/08	135.37	13.02	3.88	116.50	0.00	0.19	116.50	0.00	116.50	0.00	3.85
2	2/1/16/16	202.94	13.16	10.53	178.94	3.07	1.02	153.78	0.00	153.78	0.00	10.25
3	2/1/24/21	276.57	29.25	20.65	298.01	9.48	5.38	181.74	0.00	181.74	0.00	20.17
4	2/1/30/31	456.21	163.34	41.81	356.72	6.81	15.47	223.68	0.00	223.68	0.00	37.21
5	2/2/14/17	260.96	8.79	9.85	198.05	0.00	0.74	184.54	1.47	184.07	0.00	9.53
6	2/2/36/28	608.36	146.98	41.09	373.73	4.56	11.80	304.07	1.98	302.90	0.00	40.33
7	2/2/49/48	1335.09	205.47	97.65	573.18	6.77	57.32	400.76	2.46	396.10	0.00	94.09
8	2/2/64/61	1973.98	80.25	175.28	1072.73	5.94	420.61	504.45	12.63	484.87	2.56	156.17
9	2/3/25/23	394.00	50.39	24.37	378.39	7.86	5.16	283.33	2.50	279.60	0.00	23.01
10	2/3/51/46	1339.98	217.65	114.59	659.16	11.17	87.90	494.19	5.20	490.47	3.52	93.85
11	2/3/72/81	2683.23	61.08	347.88	983.96	3.81	452.43	739.08	3.07	720.20	3.72	233.29
12	2/3/97/97	3548.82	79.56	553.78	2970.75	0.00	3832.51	903.34	11.52	844.63	1.65	376.37
13	3/1/12/08	89.47	7.55	5.84	93.20	0.00	0.09	83.88	0.00	83.88	0.00	5.77
14	3/1/16/16	142.13	7.12	15.64	164.73	8.08	0.49	107.18	0.00	107.18	0.00	15.37
15	3/1/24/21	210.63	15.92	31.36	206.90	18.86	1.61	122.56	3.15	121.16	0.00	30.25
16	3/1/30/31	384.45	46.30	62.16	320.84	11.31	7.59	149.35	0.74	149.12	0.00	55.81
17	3/2/14/17	164.96	15.33	14.71	179.41	9.70	0.52	123.49	0.00	123.49	0.00	14.29
18	3/2/36/28	433.85	43.75	61.77	311.75	21.01	9.09	203.88	4.15	201.08	2.21	60.49
19	3/2/49/48	815.03	19.70	146.54	474.16	10.49	42.07	271.68	3.84	266.55	3.15	141.13
20	3/2/64/61	1135.64	15.77	266.95	616.28	11.84	122.25	338.78	6.33	325.27	1.63	234.25
21	3/3/25/23	317.58	31.24	35.91	239.99	10.59	3.07	194.09	2.47	191.06	0.00	34.51
22	3/3/51/46	857.44	25.19	147.93	529.84	9.96	48.45	329.00	0.98	327.83	1.57	140.77
23	3/3/72/81	1521.96	29.28	353.79	873.75	17.64	279.92	494.43	3.93	486.97	5.71	349.93
24	3/3/97/97	1982.36	64.99	841.78	993.98	3.15	612.57	604.17	8.22	564.09	0.74	564.55

Table 5 and Fig. 6 show the makespan obtained from the parallel SA with 24 problems which are generated with unbalanced rate of inbound and outbound containers. If the problems have similar total number of containers, the problem which have more outbound containers than inbound containers will lead to higher makespan value. The balance rate of inbound and outbound containers gives smaller makespan than the unbalance case.

Table 5: Effect of the average percentage of wagons with In/Outbound containers on makespan

No.	No. of Cranes/Train- s/In/Outbound containers	Average % of a Wagon with an inbound container	Average % of a wagon with an outbound container	Parallel SA		
				Makespan		CPU time (s)
				Mean	(Std.)	
25	2/1/12/32	0.25	0.95	181.74	(0.00)	15.37
26	2/1/19/25	0.50	0.75	181.74	(0.00)	19.01
27	2/1/26/19	0.75	0.50	181.74	(0.00)	19.77
28	2/1/32/06	0.95	0.25	163.10	(0.00)	7.69
29	2/2/16/62	0.25	0.95	384.68	(0.74)	39.69
30	2/2/35/51	0.50	0.75	371.87	(1.96)	71.41
31	2/2/45/29	0.75	0.50	323.87	(0.00)	52.21
32	2/2/59/14	0.95	0.25	358.35	(0.98)	33.05
33	2/3/24/94	0.25	0.95	687.35	(3.47)	90.25
34	2/3/54/76	0.50	0.75	631.66	(3.00)	164.18
35	2/3/78/56	0.75	0.50	644.01	(2.73)	174.73
36	2/3/94/31	0.95	0.25	671.97	(3.15)	116.57
37	3/1/12/32	0.25	0.95	121.16	(0.00)	23.05
38	3/1/19/25	0.50	0.75	121.16	(0.00)	28.50
39	3/1/26/19	0.75	0.50	125.82	(0.00)	29.65
40	3/1/32/6	0.95	0.25	107.18	(0.00)	11.53
41	3/2/16/62	0.25	0.95	256.53	(1.72)	59.53
42	3/2/35/51	0.50	0.75	250.24	(1.63)	107.11
43	3/2/45/29	0.75	0.50	214.59	(1.32)	78.31
44	3/2/59/14	0.95	0.25	236.96	(1.13)	49.57
45	3/3/24/94	0.25	0.95	459.24	(2.56)	135.37
46	3/3/54/76	0.50	0.75	425.46	(2.50)	246.25
47	3/3/78/56	0.75	0.50	436.18	(3.07)	262.09
48	3/3/94/31	0.95	0.25	452.72	(3.48)	174.85

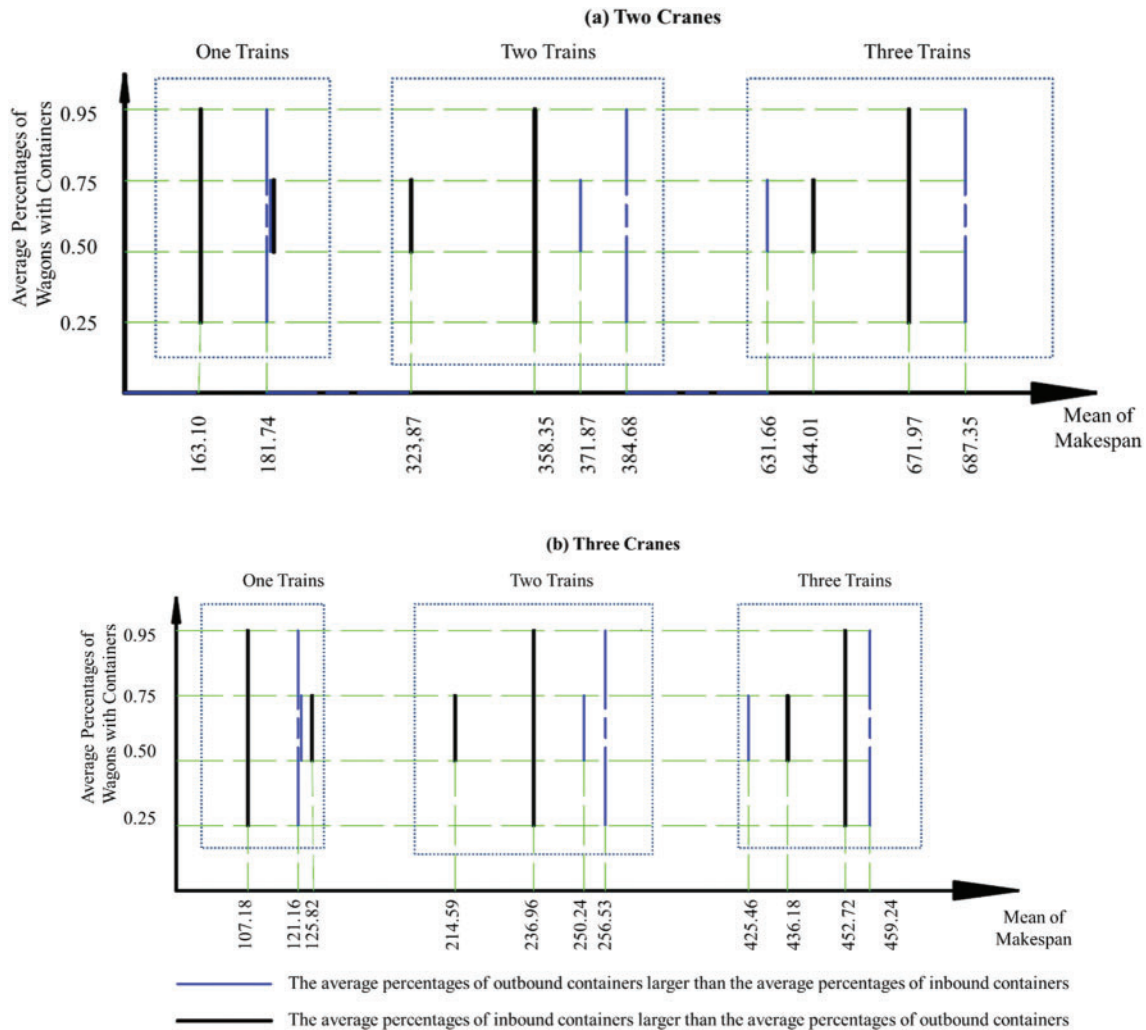


Figure 6: The average of percentages of wagon with containers vs. the mean of makespan

6 Conclusions

In this paper, we considered a multi-crane scheduling problem in rail stations. We not only assigned the job tasks to cranes in the dynamic boundary of cranes but also determined the sequence scheduling of each crane. The makespan of cranes was used as an optimization criterion and a Simulated Annealing (SA) and a parallel SA algorithm were proposed to find the solutions. We investigated the performance of algorithms by numerical examples. In general, the parallel SA algorithm outperformed the SA. Furthermore, the algorithms proposed in this paper gave better solutions than other considered algorithms. Moreover, we considered problems with balanced and unbalanced rates of inbound and outbound containers. The problems which had a balance rate of inbound and outbound containers gave smaller makespan, because cranes can perform more dual-cycle operations. When the rate of the outbound container is bigger than the rate of the inbound container, it usually leads to a higher makespan, because of the occurrence of re-handling cases.

Acknowledgement: None.

Funding Statement: The authors received no funding for this study.

Author Contributions: The authors confirm contribution to the paper as follows: Study conception and design: NVAD, NHT; data collection: NVAD, NLT; analysis and interpretation of results: NVAD, NLT, NHT; draft manuscript preparation: NVAD, NHT. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: Not applicable.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] E. Kozan, "Increasing the operational efficiency of container terminals in Australia," *J. Oper Res Soc*, vol. 48, no. 2, pp. 151–161, 1997.
- [2] A. Ballis and J. Golias, "Comparative evaluation of existing and innovative rail-road freight transport terminals," *Transp. Res. Part A: Policy Pract.*, vol. 36, no. 7, pp. 593–611, 2002.
- [3] C. Abacoumkin and A. Ballis, "Development of an expert system for the evaluation of conventional and innovative technologies in the intermodal transport area," *Eur. J. Oper. Res.*, vol. 152, no. 2, pp. 410–419, 2004.
- [4] E. Kozan, "Optimum capacity for intermodal container terminals," *Transp. Plan. Technol.*, vol. 29, no. 6, pp. 471–482, 2006.
- [5] T. Benna and M. Gronalt, "Generic simulation for rail-road container terminals," in *Proc. 40th Winter Simul. Conf. 2008*, Miami, FL, USA, Dec. 6–12, 2008, pp. 2656–2660.
- [6] T. A. Feo and J. L. González-Velarde, "The intermodal trailer assignment problem," *Transport. Sci.*, vol. 29, no. 4, pp. 330–341, 1995.
- [7] N. Bostel and P. Dejax, "Models and algorithms for container allocation problems on trains in a rapid transshipment shunting yard," *Transport. Sci.*, vol. 32, no. 4, pp. 370–379, 1998.
- [8] P. Corry and E. Kozan, "An assignment model for dynamic load planning of intermodal trains," *Comput. Oper. Res.*, vol. 33, no. 1, pp. 1–17, 2006.
- [9] F. Bruns and S. Knust, "Optimized load planning of trains in intermodal transportation," *OR Spectrum*, vol. 34, no. 3, pp. 511–533, 2012.
- [10] K. Alicke, "Modeling and optimization of the intermodal terminal Mega Hub," in H. O. Günther, K. H. Kim (Eds.), *Container Terminals and Automated Transport Systems: Logistics Control Issues and Quantitative Decision Support*, Berlin Heidelberg: Springer Berlin Heidelberg, 2005, pp. 307–323.
- [11] N. Boysen and M. Fliedner, "Determining crane areas in intermodal transshipment yards: The yard partition problem," *Eur. J. Oper. Res.*, vol. 204, no. 2, pp. 336–342, 2010.
- [12] N. Boysen, M. Fliedner, and M. Kellner, "Determining fixed crane areas in rail-rail transshipment yards," *Transp. Res. Part E: Logist. Transp. Rev.*, vol. 46, no. 6, pp. 1005–1016, 2010.
- [13] B. J. Jeong and K. H. Kim, "Scheduling operations of a rail crane and container deliveries between rail and port terminals," *Eng. Optimiz.*, vol. 43, no. 6, pp. 597–613, 2011.
- [14] E. Pap, G. Bojanic, N. Ralevic, M. Georgijevic, and V. Bojanic, "Crane scheduling method for train reloading at inland intermodal container terminal," in *Proc. 2012 IEEE 10th Jubilee Int. Symp. Intell. Syst. Inf.*, 2012, pp. 189–192.
- [15] P. Guo, W. Cheng, Z. Zhang, M. Zhang, and J. Liang, "Gantry crane scheduling with interference constraints in railway container terminals," *Int. J. Comput. Int. Syst.*, vol. 6, no. 2, pp. 244–260, 2013.

- [16] V. A. D. Nguyen and W. Y. Yun, "Optimal job scheduling of a rail crane in rail terminals," *Int. J. Ind. Eng.*, vol. 21, no. 3, pp. 129–140, 2014.
- [17] K. H. Kim and Y. M. Park, "A crane scheduling method for port container terminals," *Eur J. Oper. Res.*, vol. 156, no. 3, pp. 752–768, 2004.
- [18] L. Moccia, J. F. Cordeau, M. Gaudioso, and G. Laporte, "A branch-and-cut algorithm for the quay crane scheduling problem in a container terminal," *Nav. Res. Log.*, vol. 53, no. 1, pp. 45–59, 2006.
- [19] N. Kaveshgar, N. Huynh, and S. K. Rahimian, "An efficient genetic algorithm for solving the quay crane scheduling problem," *Expert Syst. Appl.*, vol. 39, no. 18, pp. 13108–13117, 2012.
- [20] D. H. Lee and J. H. Chen, "An improved approach for quay crane scheduling with non-crossing constraints," *Eng. Optimiz.*, vol. 42, no. 1, pp. 1–15, 2010.
- [21] M. Sammarra, J. F. Cordeau, G. Laporte, and M. F. Monaco, "A tabu search heuristic for the quay crane scheduling problem," *J. Sched.*, vol. 10, no. 4–5, pp. 327–336, 2007.
- [22] W. C. Ng and K. L. Mak, "Quay crane scheduling in container terminals," *Eng. Optimiz.*, vol. 38, no. 6, pp. 723–737, 2006.
- [23] F. Meisel, "The quay crane scheduling problem with time windows," *Nav. Res. Logist.*, vol. 58, no. 7, pp. 619–636, 2011.
- [24] X. Chen, S. He, Y. Zhang, L. Tong, P. Shang and X. Zhou, "Yard crane and AGV scheduling in automated container terminal: A multi-robot task allocation framework," *Transp. Res. Part C: Emerg. Technol.*, vol. 114, pp. 241–271, 2020.
- [25] Y. Yang, X. Zhu, and A. Haghani, "Multiple equipment integrated scheduling and storage space allocation in Rail-Water intermodal container terminals considering energy efficiency," *Transp. Res. Record.*, vol. 2673, pp. 199–209, 2019.
- [26] J. Li, A. Xu, and X. Zang, "Simulation-based solution for a dynamic multi-crane-scheduling problem in a steelmaking shop," *Int. J. Prod. Res.*, vol. 58, pp. 6970–6984, 2020.
- [27] L. Zhen, H. Hu, W. Wang, X. Shi, and C. Ma, "Cranes scheduling in frame bridges based automated container terminals," *Transp. Res. Part C: Emerg. Technol.*, vol. 97, pp. 369–384, 2018.
- [28] A. Schulz, M. Fliedner, B. Fiedrich, and C. Pfeiffer, "Levelling crane workload in multi-yard rail-road container terminals," *Eur. J. Oper. Res.*, vol. 293, pp. 941–954, 2021.
- [29] G. Ren, X. Wang, J. Cai, and S. Guo, "Allocation and scheduling of handling resources in the railway container terminal based on crossing crane area," *Sustain.*, vol. 13, no. 3, pp. 1190, 2021. doi: [10.3390/su13031190](https://doi.org/10.3390/su13031190).
- [30] J. Li *et al.*, "A flexible scheduling for twin yard cranes at container terminals considering dynamic cut-off time," *J. Mar. Sci. Eng.*, vol. 10, no. 5, pp. 675, 2022. doi: [10.3390/jmse10050675](https://doi.org/10.3390/jmse10050675).
- [31] M. Lundy and A. Mees, "Convergence of an annealing algorithm," *Math. Program.*, vol. 34, pp. 111–124, 1986.
- [32] R. Ruiz and T. Stützle, "A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem," *Eur. J. Oper. Res.*, vol. 177, pp. 2033–2049, 2007.