

Received April 28, 2022, accepted May 31, 2022, date of publication June 8, 2022, date of current version June 23, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3180752

Heuristic/Metaheuristic-Based Simulation Optimization Approaches for Integrated Scheduling of Yard Crane, Yard Truck, and Quay Crane Considering Import and Export Containers

HSIEN-PIN HSU¹, CHIEN-CHANG CHOU², AND CHIA-NAN WANG³

¹National Kaohsiung University of Science and Technology, Nanzih, Kaohsiung 81157, Taiwan

²National Kaohsiung University of Science and Technology, Qijin, Kaohsiung 80543, Taiwan

³National Kaohsiung University of Science and Technology, Sanmin, Kaohsiung 807618, Taiwan

Corresponding author: Chia-Nan Wang (cn.wang@nkust.edu.tw)

This work was supported by the Ministry of Science and Technology (MOST) Taiwan under Grant MOST 109-2410-H-992-020.

ABSTRACT Yard cranes (YCs), yard trucks (YTs), and quay cranes (QCs) are the main material handling equipment (MHE) used to fulfill Yard Storage Plan (YSP) and Vessel Stowage plan (VSP) in a traditional container terminal. However, previous studies have often considered part of these MHE, with either import or export containers being neglected simultaneously. Such studies at most can only achieve local optimality which in a worst-case scenario can cause bottleneck(s) in the downstream operations. In addition, it is noted that the YSP, VSP, and their constraints have been often neglected in previous studies. Such negligence can cause operational problems. To best operate a container terminal, it needs to address the above issues. Our literature review shows that exact methods have difficulty solving a problem of big size to optimality due to NP-hard, thus alternative approaches are still required. Simulation-based optimization approaches are found with the potential to deal with container terminal problems. However, software for the integration of simulation and optimization effectively is found unavailable. In this research, different simulation-based optimization approaches, based on a simulation-based optimization framework (HSBOF), have been developed. These approaches help to schedule YCs, YTs, and QCs in handling both import and export containers for a ship in a traditional container terminal, taking YSP, VSP, and their constraints into consideration. Various heuristics/metaheuristics have been employed in the HSBOF as a sequencing method to arrange alternative operational sequences of containers. The results show that the Simulation(ISFLA) has the best performance.

INDEX TERMS Quay crane, yard crane, yard truck, terminal, metaheuristic, simulation-based optimization approach.

I. INTRODUCTION

Recently, port congestion and supply chain disruption have introduced problems to the world, which highlights the importance of maritime transport. Maritime container transport is found especially important as it takes a large share of maritime transport. This kind of transport has an essential part of the global transportation system. The use of standard containers leads to the advantage of the ease of loading,

The associate editor coordinating the review of this manuscript and approving it for publication was Muhammad Zakarya¹.

unloading, and transportation in the supply chain. The increasing global container shipments highlight the need for continuous improvement in the capacity and efficiency of a container terminal [1].

Improving the terminal capacity, such as by increasing the total number of container terminals, is a hardware approach that can have a significant effect. However, it belongs to a long-term plan that is time-consuming and costly. To achieve an immediate effect, it needs to improve the operational efficiency of the material handling equipment (MHE) used in a container, which is considered one software approach

focused in this research. In a conventional container terminal, yard cranes (YCs), yard trucks (YTs), and quay cranes (QCs) are the three essential MHE for handling containers. They affect the overall performance of a container terminal considerably [3], [4]. Coordinating the operations of these MHEs is critical.

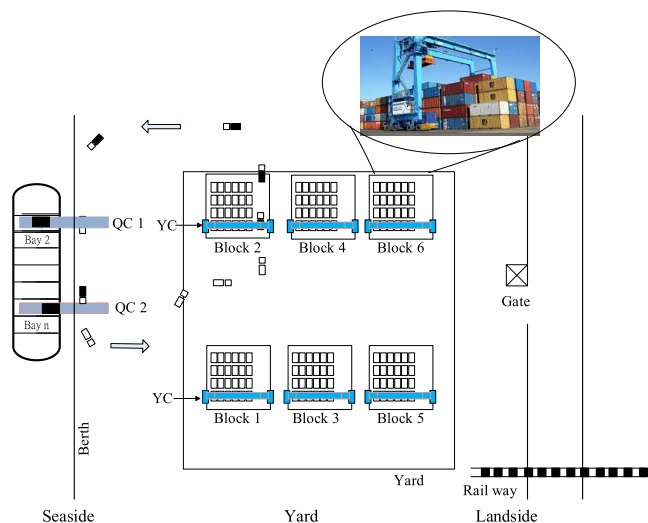


FIGURE 1. A common layout of a seaport container terminal.

Fig. 1 shows the layout of a traditional container terminal which is separated into the seaside, yard, and landside areas with different MHE [5]–[7]. For example, the seaside uses QCs to load/unload containers; the yard uses YCs to store/retrieve containers; the YTs transport containers between the QCs and YCs. These MHE are necessary to handle both import and export containers for a ship in a container terminal. The import containers are unloaded firstly, followed by the export containers. Containers can be treated as groups or individuals when planning. This research treats containers as individuals.

A container terminal faces many operational problems, such as berth allocation problem (BAP), quay crane assignment problem (QCAP), quay crane scheduling problem (QCSP), yard crane assignment problem (YCAP), yard crane scheduling problem (YCSP), yard truck assignment problem (YTAP), yard truck scheduling problem (YTSP), YT routing and traffic control problem, storage planning, space planning problem, and pre-reshuffling problem, etc. [8]. These problems are belonging to different areas. Solving these problems in one study is impossible, thus this study focuses on the YCSP, YTSP, and QCSP simultaneously.

The container terminal is a complex system that is hard to optimize its operations. This complexity comes from many factors such as different types of containers, storage positions, storage/stowage plans, and MHE. For instance, a container can have various storage positions such as one in the yard arranged by Yard Storage Plan (YSP) and another in the vessel arranged by Vessel Stowage Plan (VSP). The purpose of the activities in the container terminal can be regarded as to

fulfill the two plans through the use of MHE, including YCs, YTs, and QCs.

The [9] reviewed some studies focusing on the seaside and yard side areas. That study helps identify the following research gaps: (1) the only focus on one MHE; (2) the negligence of YSP and VSP and their constraints; and (3) the lack of applying simulation approaches to deal with container terminal problems [9]. Another study [10] reviewed some maritime logistics problems and it pointed out that software for the integration of simulation and optimization effectively is unavailable. This is another research gap that is required to be addressed.

Different kinds of approaches such as exact approaches, evolutionary heuristics, heuristics, and simulation have been applied to deal with container terminal problems [9]. As kinds of exact approaches, Mixed Integer Linear Programming (MILP), Mixed Integer Programming (MIP), Dynamic Programming (DP), Branch and Bound (B&B), and Column Generation (CG) have been often used to find the optimal solution. However, these exact approaches tend to face the computationally intractable problem in dealing with a big instance so Rules, genetic algorithms (GAs), local search-based algorithms, greedy algorithms, particle swarm optimization (PSO), etc. have been used to find an optimal/near-optimal solution [9].

In this research, a heuristic/metaheuristic-based simulation optimization framework is proposed for developing simulation-based optimization approaches in which heuristics and metaheuristics serve as a sequencing method. The developed approaches are used to plan the collaborative operations of YCs, YTs, and QCs in handling both import and export containers, taking YSP and VSP and their constraints into consideration. A simulation model is used in the framework to simulate the planning results. An iterative procedure is used to improve and find the best planning result.

In this research, one heuristic (Random) and some metaheuristics, including GA, PSO, firefly algorithm (FA), shuffled frog-leaping algorithm (SFLA), and an improved SFLA (ISFLA), have been used as alternative sequencing methods respectively used in the framework. The standard SFLA, proposed by Eusuff and Lansey [11], was originally used to design the water distribution network which is not with a discrete solution domain. The standard SFLA is found with the following weaknesses; (1) lack of global strategy for grouping frogs, (2) lack of capability to handle the problem with the discrete domain, and (3) lack of adaptive jump to approach the target frog adequately. The ISFLA improves these weaknesses of the standard SFLA by introducing new features. Then, extensive experiments have been conducted to investigate their effectiveness. The experimental results show that the Simulation(ISFLA) outperforms the others in terms of the total cost consisting of sub-costs of makespan and penalty of constraint violation. The statistical *t*-test has been used to validate the robustness of the results.

The rest of this paper is organized as follows. Section 2 reviews background knowledge and relevant

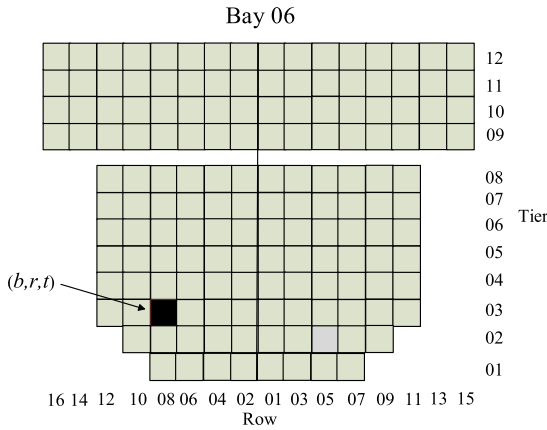


FIGURE 2. Stowage positions of containers in a vessel.

studies. Section 3 defines the problems of QCSP, YTSP, and YCSP. Section 4 proposes a hierarchical simulation-based optimization framework for developing simulation-based optimization approaches. Section 5 shows and discusses the experimental results. Finally, Section 6 concludes and suggests future research directions.

II. BACKGROUND AND RELEVANT STUDIES

A. VESSEL STOWAGE PLAN

A container ship contains multiple bays. Each bay is composed of many stowage positions for storing containers [12]. The positions of containers in a vessel are planned by the so-called Vessel Stowage Plan (VSP). Fig. 2 shows the example of Bay 6 with stowage positions in a vessel. Each position in the VSP is denoted as (b, r, t) , with the b , r , and t indicating a bay, row, and tier numbers, respectively.

B. YARD STORAGE PLAN

A container yard consists of multiple blocks [13]. Fig. 3 illustrates a block of Asia type with containers stacked one on one. The assignment of containers to storage positions in a block is specified by the so-called Yard Storage Plan (YSP). Each storage position is denoted as (x, y, z) , with the x , y , and z indicating bay, row, and tier number, respectively. Usually, import and export containers are respectively stored in different blocks to avoid problems. These blocks provide temporary storage service for containers before loading/unloading into/from a ship.

C. RELATED STUDIES

1) STUDIES FOCUSING ON ONE SINGLE PROBLEM

Some studies focused on QC [14]–[18]. The [14] proposed a coordinated optimization model to deal with the QCSP and QCAP (quay crane assignment problem). The Red Deer Algorithm (RDA) was at its first time applied to solve the two problems simultaneously. Differently, the [15] first decomposed the QCSP into a workload-assignment master problem and operation-sequence slave sub-problem, which were formulated as mathematical models, considering

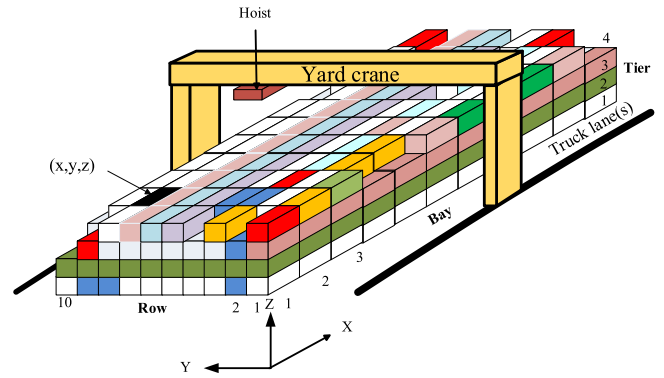


FIGURE 3. Storage positions of containers in a block.

non-crossing constraints. This model was based on logic-based Bender decomposition. The Logic-based cuts were proposed to ensure the convergence of this approach. The [16] used a two-level dynamic programming (DP) algorithm to solve the QCSP. The solution found by the DP algorithm was compared to a lower bound. The [17] dealt with the QCSP with the uncertainty of loading/unloading times of containers and travel time of quay crane taking into consideration. A simulation-optimization approach was used to find good-quality solutions. An Ant Colony Optimization (ACO) metaheuristic hybridized with a Variable Neighborhood Descent (VND) local search was used to assign tasks to QCs and determine the operational sequences of containers on each QC. The simulation was used to generate scenarios. The [18] was also devoted to solving the QCSP as quay cranes are considered the most important MHE in port container terminals. A MIP mathematical model was formulated. In addition, a B&B method was developed to find the optimal solution. To overcome the computational difficulty of the B&B method, a heuristic search algorithm called the greedy randomized adaptive search procedure (GRASP) was proposed.

Others focused on the YT [19]–[23]. The [19] formulated a mathematical model for reducing emissions generated from YTs idling. The total truck waiting time in the yard was estimated by using discrete event simulation. To solve this model, a GA was proposed and a case study in a Singaporean port was analyzed. The [20] proposed a Look-Ahead Dispatching Method for automated guided vehicles (AGVs) in Automated Port Container Terminals. This study discussed how to dispatch AGVs by utilizing information about locations and times of future tasks. The [21] studied the optimization of dispatching YTs to containers for minimizing the total service time required for a ship. The total time includes the unloading time of import containers as well as the loading time of export containers. Some easy heuristic algorithms were developed for dispatching YTs. The [22] used GA to address the YTSP. A MIP mathematical model was firstly formulated for the YTSP. However, due to NP-hard, a GA was used to solve the problem. The proposed GA has been compared to other GAs with six popular crossover operators. The [23] presented a

new MIP model considering two strategies, outsourcing, and collaboration, simultaneously. In literature, the two strategies have been used as alternatives to keep a large number of YTs to alleviate the possible arising of bottlenecks during peak hours. The outsourcing strategy means renting additional YTs for a container terminal while the collaborative strategy means the sharing of YTs among multiple container terminals with adjacent locations. However, the two strategies have been dealt with separately in the literature. The proposed model is solved using a rolling horizon heuristic.

Still, others focused on the YC [24]–[29]. The [24] first formulated a MIP mathematical model for the YCSP with non-interference constraints taken into consideration. However, due to NP-hard, a hybrid GA was further used to obtain near-optimal solutions. The [25] formulated a MIP for the QCSP, considering non-interference constraints. However, due to NP-hard, the authors further proposed an alternative approach. The [26] first formulated an IP mathematical model for the YCSP. The objective was to minimize the total task delay at blocks. However, due to NP-hard, a novel dynamic rolling-horizon decision strategy, along with a simulation model, was then proposed for solving the YCSP. The [27] was devoted to solving the YCSP with the trade-off between efficiency and energy consumption considered. The YCSP was regarded as a vehicle routing problem with soft time windows (VRPSTW) and formulated into a MIP. The objectives aimed at minimizing the total completion delay of all task groups and the total energy consumption of all YCs. Subsequently, an integrated simulation optimization method is developed for solving the problem. The simulation is used to evaluate alternative solutions. On the other hand, an optimization algorithm that integrates the genetic algorithm (GA) and the particle swarm optimization (PSO) algorithm was used to explore alternative solutions. The [28] conducted a comparative study of two modeling approaches, centralized and decentralized, for solving the YCSP. The centralized approach was found to outperform the decentralized approach by an average of 16.5 %, due to complete and accurate information about future truck arrivals. Nevertheless, the decentralized approach can better adapt to real-time truck arrivals, making it better suited for real-life operations. Finally, it is suggested to integrate the two approaches as they have complementary features. The [29] studied the YCSP for a given set of loading/unloading jobs with different ready times. The objective was to minimize the total waiting time for jobs. A B&B algorithm was proposed to solve the scheduling problem to optimality. In addition, efficient and effective algorithms are proposed to find lower bounds and upper bounds.

2) INTEGRATED STUDIES OF MULTIPLE PROBLEMS

Other studies have been devoted to solving integrated problems. The [30] regarded the integrated scheduling problem of QC, YT, and YC as a hybrid flow shop scheduling problem. The authors highlight that good coordination among the MHE is essential to minimizing the service times of ships.

In addition to formulating a mathematical model, a Tabu search (TS) was proposed for dealing with a big instance within an acceptable time. The [31] dealt with the QCSP and YCSP simultaneously with the storage positions in a yard as well as a vessel being considered. A heuristic was proposed as a scheduling means for an automated container handling system with twin 40' cranes. However, that study neglected the YTs. The [32] formulated a MIP model to deal with the YCSP and YTSP simultaneously. However, the mathematical model was found hard to solve a big instance. Two methods, based on Benders' decomposition, were thus further proposed. However, this study neglected the QCs. The [33] formulated a Constraint Programming (CP) model to deal with the QCSP, YCSP, and YTSP simultaneously. The YTs can serve multiple vessels simultaneously. The objective was to minimize empty travel times of the QC, YC, and YT. However, it is found that the CP has difficulty dealing with a big instance and even a small instance. A three-stage approach, based on heuristic and disjunctive graph, was thus proposed. This approach can deal with a big problem of up to 500 containers. However, that study neglected the VSP and YSP constraints. The [34] proposed an integrated model including YC and AGV, considering yard storage position. The objective was to optimize the yard operation. This study considered the loading operation as short-term planning and the YSP as long-term planning. In addition, it assumed that the processing sequences of containers for QCs were known. A MIP model was formulated to minimize the total berthing time for ships. However, a GA was another proposed to solve a big instance. The [35] studied an integrated problem including yard storage allocation, QCSP, and YTSP. A MIP was formulated, which minimizes two weighted objectives. The first objective is makespan and the second objective is the total transportation distance of YTs. However, this study neglected the non-crossing constraint required for QCs. The yard storage assignment considered block, instead of container slots assignments. A two-stage heuristic algorithm was proposed. The first stage uses ant colony optimization (ACO) for yard storage allocation. The second stage uses a greedy algorithm and a local search algorithm to solve the YCSP and QCSP simultaneously. The [36] studied the integrated problem of QCSP, YTSP, and YCSP. A MIP was firstly formulated to minimize the overall departure delay of ships and energy consumption of tasks. Then, a simulation-based optimization approach, which integrates GA with PSO, was proposed to solve the integrated problem. The GA performs a global search while the PSO performs a local search. The results showed the optimal tradeoff between time-saving and energy-saving. The [37] studied the YTSP together with the yard storage problem for import containers in an automated container terminal. A MIP model was formulated for dealing with small instances while a GA was further used to deal with big instances. However, this study neglected QC interference and export containers. The [12] proposed a framework to deal with the integrated problem of QCSP and 3D stowage planning problem. Simulation and GA were used to solve the

problems with 30 ports and 1500 TEUs of ships as well as 15 ports and 22,000 TEUs of ships, under the use of 2 QCs. The additional consideration of QCSP increased about 45.82% of loading/unloading time for the solution to the 3D stowage planning problem. The [38] formulated a MIP model and a CP model to optimize the assignment and scheduling of QC and YC. The containers were treated as groups (or shipments). Each group has a port destination and belongs to a customer. Then, QC and YC are assigned to handle these shipments. Containers of the same group are stored in the same vessel bay and storage block. The CP model was found more efficient than the MIP model. The [39] studied integrated scheduling of QCs, YCs, and AGVs as well as the routing problems. Both import and export containers were considered. A bi-level optimization model was proposed to minimize makespan. The top level considers the scheduling problem and the second level handles the AGV routing problem. A GA-based rule was proposed to prevent congestion. The authors highlighted the impotence to solve these problems at the same time. The [40] considered the QCSP, YCSP, and AGV simultaneously as a hybrid flow shop problem and proposed a formulation to process both import and export containers. In addition, this study allows a QC to handle two containers at the same time. SA was used as the main approach. The [41] presented a planning model for the integration of BAP, QCSP, and YTSP. The developed model contributes to the existing literature by including energy-saving goals and some realistic factors such as shortages of internal trucks and handling time estimations. A Lagrangian relaxation-based method was proposed in that study. The [42] integrated the assignment of QCs in container terminals and internal truck sharing assignments among them. A bi-objective optimization model was proposed. This model includes multiple assignment phases, including the assignments of the vessel to container terminals, cranes to terminals, cranes to vessels, and trucks to cranes. The model also seeks to improve the efficiency and effectiveness of YTs by sharing them among multiple terminals. Two meta-heuristic multi-objective algorithms, including modified non-dominated sorting genetic algorithm-II (MNSGA-II) and modified multi-objective particle swarm optimization (MMOPSO), are presented. The [43] presented a mathematical model for solving the QCAP, the specific QCAP, and the assignment of YTs to each QC simultaneously. The proposed model considers important practical aspects such as the limited availability and operational cost of the YTs. A Lagrangian relaxation and subgradient optimization procedure-based heuristic were proposed. The [13] developed integrated scheduling for YC and YT in the handle of export containers collaboratively in the yard side area of a container terminal. However, this study neglected QCs and the VSP constraints in the seaside area. The [44] extended the [13] to further consider the QC operations as a whole, in addition to taking the VSP constrain into consideration. However, this study and many of the past studies, including the [13], have neglected either import or export containers whose operations are found can also affect

the whole operations in a container terminal. Both operations of inbound and outbound containers should be equally esteemed.

3) RESEARCH GAP

The above literature review shows that many past studies have been devoted to one single MHE problem, which at most can only achieve local optimality. The local optimality in the worst case can introduce bottleneck(s) to the downstream operations if the whole operations in the container terminal are not well synchronized. Coordinating the whole operations in a container terminal is essential. A global view requests coordination of all of the operations in the container terminal, with the inbound and outbound containers being equally esteemed and considered simultaneously.

Table 1 compares the present research with some previous studies in terms of problems domain, export/import (I/E) container, and methodology. The present research has a wider scope which includes YC, YT, QC, VSP, YSP, VSP and YSP constraints, and both import and export containers. The consideration of both import and export containers has complicated the integrated scheduling problem considerably. In addition, the present research investigates more methodologies (SIM, ISFLA, SFLA, FA, GA, PSO, and RA). Various heuristic/metaheuristic-based optimization approaches have been developed to solve this integrated scheduling problem.

The [10] reviewed studies with simulation and optimization. A total number of 107 papers published within the last two decades were reviewed and classified into three areas: terminal, shipping line, and hinterland transport. Of these papers, 20 papers were classified into the “yard” area while 8 papers were classified into the “terminal” area. This shows more simulation and optimization-based studies are required for the container terminals. Meanwhile, the [10] pointed out that software for the integration of simulation and optimization effectively was relatively unavailable – which is regarded as one research gap in this research.

III. DEFINITION OF PROBLEM, ANALYSIS OF CONTAINER TERMINAL OPERATIONS, FORMULATION OF A MATHEMATICAL MODEL

A. PROBLEM DEFINITION

The problem to be dealt with in this research is defined as an integrated scheduling problem of QC, YT, and YC in a traditional container terminal. The traditional container terminal is a non-automated container terminal that uses single-spreader-single-trolley QCs and tired-type YCs and YTs to complete the changes of storage positions of import and export containers of a ship between the yard and vessel. Each import/export container is assigned with a yard storage position and a vessel stowage position defined in the YSP and VSP, respectively. This problem also considers the YSP and VSP constraints. All of the QCs, YTs and YCs work with the single-cycle operation mode.

TABLE 1. The comparisons of features of some integrated studies with the present research.

Integrated Study	Problems domain					E	I	Methodology
	YSP	YC	YT	QC	VSP			
Chen et al. (2007) [30]		v	v	v		-	-	TS
Zheng [31]	v	v		v	v	-	-	RULE
Cao et al. (2010) [32]		v	v			v		RULE, GA
Chen et al. (2013) [33]		v	v	v		v	v	RULE, CP DG
Wu et al. (2013) [34]	v	v	v			v		MIP, GA, NLMIP
Xue et al. (2013) [35]			v	v		v	v	ACO
He et al. (2015) [36]		v	v	v		v	v	MIP, SIM, PSO, GA
Luo et al. (2016) [37]	v	v	v				v	MIP, GA
Kizilay et al. (2018) [38]		v		v		v	v	MIP,CP
Yang et al. (2018) [39]		v	v	v		v	v	MIP, GA
Jonker et al. (2019) [40]		v	v	v		v	v	SA
Karam et al. (2021) [41]			v	v		-	-	LGN-based heuristic
Vahdani et al. (2019) [42]			v	v		-	-	MMOPSO, MNSGA
Karam and Eltawil(2016) [43]			v	v		-	-	LGN, SOP-based heuristic
Hsu et al. (2021) [13]	v	v	v			v		SIM ,SBB MGPSO, GA,PSO
Hsu et al. (2021) [44]	v	v	v	v	v	v		SIM,SBB, MGPSO, GA,PSO
This research	v	v	v	v	v	v	v	SIM,ISFLA, SFLA,PSO, GA,FA,RA

YSP: yard storage plan; YC: yard crane; YT: yard truck; QC: quay crane; VSP: vessel stowage plan; E: export container; I: import container; TS: Tabu search; RULE: rule-based heuristic; GA: genetic algorithm; PSO: particle swarm optimization; CP: constraint programming; NLMIP: non-linear mixed integer programming; SIM: simulation; SA: simulated annealing; LGN: Lagrangian relaxation method; SOP: Subgradient Optimization Procedure; MIP: Mixed Integer Programming; RA: Random; MGPSO; Multi-group PSO; DG: differential evolution; MNSGA: modified non-dominated sorting GA; MMOPSO: modified multi-objective PSO; SIM: simulation; SFLA: shuffled frog-leaping algorithm; ISFLA: improved shuffled frog-leaping algorithm; FA: firefly algorithm; SBB: sort by bay; -: not available

B. OPERATIONS IN THE CONTAINER YARD

Fig. 4 (a) shows the moving paths for the YC1 to retrieve the container j at the storage position (x_j, y_j, z_j) , after loading the container i with the storage position (x_i, y_i, z_i) along the path 1. The hoist first moves to the container j along the path 2, then picks up this container. Subsequently, the hoist moves the container j to the truck lane along the path 3. Finally, it loads the container j onto the YT2. Eq. (1) is used to estimate the total retrieval time for the container j (RT_j).

$$RT_j = \text{Max}\{d(|x_j - x_i|)/v_x, d(y_j)/v_y\} + 2 \cdot d(H - z_j)/v_z + d(y_j)/v_y + 2 \cdot d(H - 1)/v_z \quad (1)$$

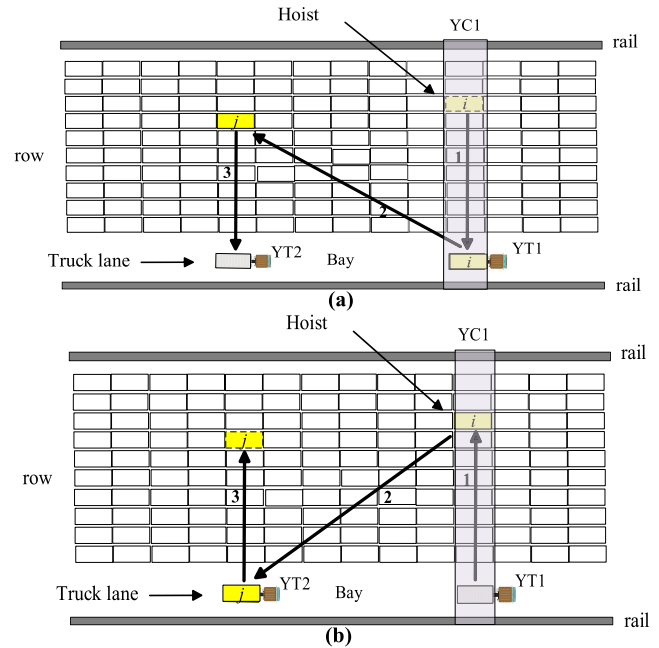


FIGURE 4. (a). The movement of hoist to retrieve container j . (b). The movement of hoist to store container i .

The right-hand side includes four terms. The first term is the moving time for the hoist to reach the position above the target container j from the position of the previous container i , where the $d(|x_j - x_i|)$ is a moving distance between the two bays of containers i and j ; the v_x is a moving speed along the x (bay) direction; the $d(y_j)$ is the distance between the row of container j and the truck lane; the v_y is a moving speed along the y (row) direction. The second term estimates the roundtrip time for the hoist to lower down and pick up the container j , where $d(H - z_j)$ is the one-trip distance; the v_z is a moving speed along the z (tier) direction. The third term is the time required for the hoist to move the container j to the truck lane. The fourth term side is the roundtrip time for the hoist to load the container j onto the YT2, where the $d(H - 1)$ is a one-trip distance to load the container. The fourth time components form the cycle time of retrieval.

Let a_j and a_p be the available times of container j and its assigned YC p ($p = 1$ in this case), respectively. Then, Eq. (2) is used to determine the start retrieval (SR_j) time of the YC p .

$$SR_j = \text{Max}\{a_j, a_p\} \quad (2)$$

The end retrieval (ER_j) time of the export container j is estimated by Eq. (3).

$$ER_j = SR_j + RT_j \quad (3)$$

Fig. 4 (b) shows the moving paths for the YC1 to store the container j to its yard storage position (x_j, y_j, z_j) . After storing the container i along the path 1, the hoist starts storing the container j . After arriving at the YT2 along the path 2, the hoist starts picking up the container j . Following this, the YC1 moves and stores the container j into its yard storage position. Eq. (4) is used to estimate the total storage time for

the container j (ST_j).

$$ST_j = \text{Max}\{d(|x_j - x_i|)/v_x, d(y_i)/v_y\} + 2 \cdot d(H - 1)/v_z + d(y_j)/v_y + 2 \cdot d(H - z_i)/v_z \quad (4)$$

Let a_j and a_p be available times of the container i and its assigned YC p , respectively. Eq. (5) is used to estimate the start storage (SS_j) time of the import container j for the YC p .

$$SS_j = \text{Max}\{a_j, a_p\} \quad (5)$$

The end storage (ES_j) time is estimated by Eq. (6).

$$ES_j = SS_j + ST_j \quad (6)$$

When storing or retrieving containers, the sequence of these containers is important. Given a set of containers to be retrieved or stored, the following constraints should be taken into account.

- YCs should not cross over each other.
- Containers of the same bay are assigned to the same YC.
- For some containers with the same bay and the same row number, a container with a lower-tier storage position is stored earlier and retrieved later.

The last one is regarded as YSP constraints whose violation is given a penalty cost (PC) which is an additional time required for recovering this unsuitable order. Eq. (7) shows the cost matrix (M1) of storing operations. The $c1_{ij}$ is set with the value 0 if there is no problem storing container j after container i . Otherwise, the $c1_{ij}$ is set with a penalty cost value PC ($PC > 0$).

$$M1 = \begin{bmatrix} c1_{11} & \cdots & c1_{1N} \\ \vdots & c1_{ij} & \vdots \\ c1_{N1} & \cdots & c1_{NN} \end{bmatrix} \quad (7)$$

Eq. (8) shows the cost matrix (M2) of container retrievals. The $c2_{ij}$ is set with the value 0 if there is no problem retrieving container j after the container i . Otherwise, the $c2_{ij}$ is set with the penalty cost PC ($PC > 0$).

$$M2 = \begin{bmatrix} c2_{11} & \cdots & c2_{1N} \\ \vdots & c2_{ij} & \vdots \\ c2_{N1} & \cdots & c2_{NN} \end{bmatrix} \quad (8)$$

The penalty costs are taken into account in the objective function to rule out those solutions violating the VSP constraints seriously.

C. OPERATIONS IN THE CONTAINER SHIP

Let (b_i, r_i, t_i) and (b_j, r_j, t_j) be vessel stowage positions of the containers i and j , respectively.

Fig. 5(a) shows the paths for the QC1 to load the container j into a ship. After loading the container i into the ship, the hoist starts to pick up the container j on the YT2 in the truck lane along the path 1. After picking up the container j , the hoist starts moving the container j to its stowage position along the path 2. Finally, the container j is stored into its stowage

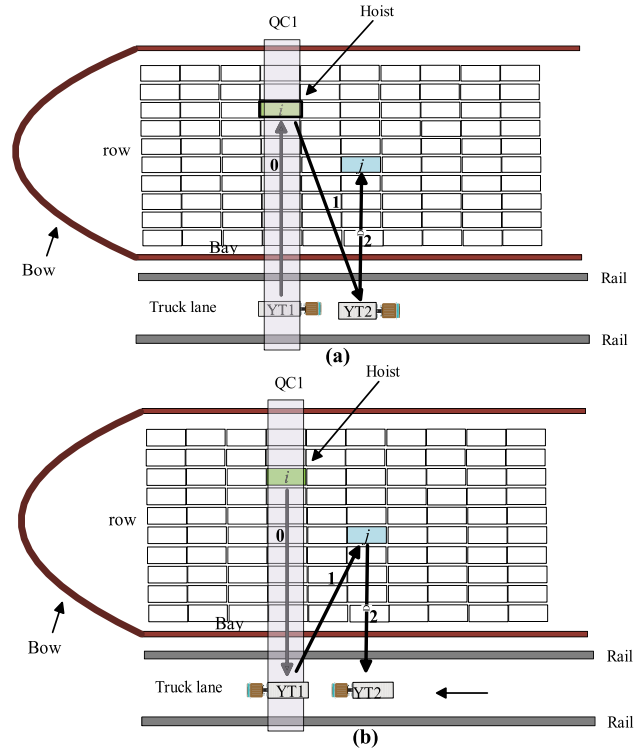


FIGURE 5. (a). The QC loading operation in the seaside. (b). The QC unloading operation in the seaside.

position (b_j, r_j, t_j) . Eq. (9) estimates the total loading (LT_j) time for the container j .

$$LT_j = \text{Max}\{d(|b_j - b_i|)/v_b, d(r_i)/v_r\} + 2 \cdot CH/v_t + d(r_j)/v_r + 2 \cdot d(H - t_j)/v_t \quad (9)$$

On the right-hand side, the first term is the time for the hoist to reach above the container j , from the position of previous container i , where the $d(|b_j - b_i|)$ indicates a distance between the two bays of containers i and j ; the v_b is a moving speed along the bay direction; the $d(r_i)$ indicates a distance between the bay r_i and the truck lane; the v_r is a moving speed along the row direction. The second term is the time for the QC1 to pick up the container j , where the CH is the one-trip distance to reach the container j on the YT2; the v_t is the moving speed of the hoist along the tier direction.

The third term is the time for the QC1 to reach the row position r_j of the container j . The $d(r_j)$ is the moving distance to reach the row position r_j of the container j . The fourth term is the roundtrip time to put the container j at its stowage position (b_j, r_j, t_j) , where $d(H - t_j)$ is the one-trip distance to reach the tier position t_j .

Let a_j be the available time of the container j and a_q be the available time of the assigned QC q , Eq. (10) is the start loading (SL_j) time of the QC q .

$$SL_j = \text{Max}\{a_j, a_q\} \quad (10)$$

And, Eq. (11) is the end loading (EL_j) time.

$$EL_j = SL_j + LT_j = \text{Max}\{a_j, a_q\} + LT_j \quad (11)$$

Fig. 5(b) shows the paths for the QC1 to unload container j from the ship. After unloading the container i from the ship to the YT1 along the path 0, the hoist starts to handle the container j . After reaching above the row position of the container j along the path 1, the QC1 starts picking up the container. After picking up this container j , the QC1 starts moving this container to the YT2 in the truck lane along the path 2. Finally, the container j is loaded onto the YT2. Eq. (12) estimates the total unloading time (ULT_j) of the container j .

$$ULT_j = \text{Max}\{d(|b_j - b_i|)/v_b, d(r_j)/v_r\} + 2 \cdot d(H - t_j)/v_t + d(r_j)/v_r + 2 \cdot CH/v_t \quad (12)$$

On the right-hand side, the first term is the time for the hoist moving from the container i to the container j . The second term is the pick-up time of the container. The third term is the time for the hoist moving the container j to the truck land. The fourth term is the round-trip loading time for the hoist to load the container onto the YT2.

Let a_j be the available time of the container j and a_q be the available time of the quay crane q assigned to the container j . Eq. (13) is the start unloading (SUL_j) time for the QC q .

$$SUL_j = \text{Max}\{a_j, a_q\} \quad (13)$$

Eq. (14) is the end unloading (EUL_j) time of the QC q .

$$EUL_j = SUL_j + ULT_j = \text{Max}\{a_j, a_q\} + ULT_j \quad (14)$$

For the QC operation, the following constraints are considered [10].

- QCs should not cross over each other.
- Containers of the same bay are assigned to the same QC.
- For some containers with the same bay and row numbers, the container with a lower-tier number is loaded earlier and unloaded later.

The last constraint is identified as a VSP constraint. Each violation of the VSP constraint is given a penalty cost. Eq. (15) shows the matrix of penalty cost (M3) for loading the container j into a ship. If the container j is loaded after the container i and this causes no problem then the $c3_{ij}$ is set with the value 0; Otherwise, $c3_{ij}$ is set with the penalty cost PC ($PC > 0$).

$$M3 = \begin{bmatrix} c3_{11} & \cdots & c3_{1N} \\ \vdots & c3_{ij} & \vdots \\ c3_{N1} & \cdots & c3_{NN} \end{bmatrix} \quad (15)$$

Eq. (15) shows the matrix of penalty cost (M4) for unloading the container j from a ship. If the container j is unloaded after the container i and this causes no problem then the $c4_{ij}$ is set with the value 0; Otherwise, $c4_{ij}$ is set with a penalty cost PC ($PC > 0$).

$$M4 = \begin{bmatrix} c4_{11} & \cdots & c4_{1N} \\ \vdots & c4_{ij} & \vdots \\ c4_{N1} & \cdots & c4_{NN} \end{bmatrix} \quad (16)$$

The penalty costs will be taken into account in the objective function to rule out those solutions violating the VSP constraints seriously.

D. OPERATION OF THE YT

YTs are used to move containers between QCs and YCs. After loading an export container j onto an assigned YT, the YT will transport this container to an assigned QC at the seaside. Let a_j be the available time of container j and $\text{Min}_{k \in \text{KE}}\{a_k\}$ be the available time of the first available YT k in the block, then the start transportation time of the export container j (STE_j) by the YT k is determined by Eq. (17).

$$STE_j = \text{Max}\left\{a_j, \text{Min}_{k \in \text{KE}}\{a_k\}\right\} \quad (17)$$

Given an average one-trip transportation time for an export container (ETK), then the end transportation time (ETE_j) of the container j is determined by Eq. (18)

$$ETE_j = SME_j + ETK = \text{Max}\left\{a_j, \text{Min}_{k \in \text{KE}}\{a_k\}\right\} + ETK \quad (18)$$

For an import container j , after being loaded onto a YT k , the YT then starts transporting the container j to its assigned YC in the container yard. Let a_j be the available time of container j and $\text{Min}_{k \in \text{KE}}\{a_k\}$ be the available time of the first available YT k at the seaside, then the start transportation time of the container j (STI_j) by the YT k is determined by Eq. (19).

$$STI_j = \text{Max}\left\{a_j, \text{Min}_{k \in \text{KE}}\{a_k\}\right\} \quad (19)$$

Given an average one-trip transportation time for an import container (ITK), the end transportation time of the import container j (ETI_j) by the YT k is determined by Eq. (20).

$$ETI_j = STI_j + ITK = \text{Max}\left\{a_j, \text{Min}_{k \in \text{KE}}\{a_k\}\right\} + ITK \quad (20)$$

E. ASSUMPTION AND OBJECTIVE FUNCTION USED IN THIS RESEARCH

The assumptions and objective function of the simultaneous problems are defined as follows:

1) ASSUMPTIONS

- The import and export containers are stored in two different blocks.
- Each import container has a current VSP position in the vessel and a future YSP position in the container yard.
- Each export container has a current YSP position in the container yard and a future VSP position in the vessel.
- Each kind of equipment (YCs, YTs, QCs) works with the same average moving speed.
- YCs do not cross over each other.
- QCs do not cross over each other.
- No interruption on operations of loading, unloading, and transporting of a container.
- No reshuffling of containers.

2) THE OBJECTIVE FUNCTION

Eq. (21) shows the objective function used in this research, which aims to minimize the makespan as well as the cost of additional time caused by violating the YSP and VSP constraints.

$$\begin{aligned}
 \text{Min } Z &= \text{Max}(EL_j) + \sum_{p=1}^{\|P\|} \sum_{j=1}^N \sum_{i=1}^N (c1_{ij}X1_{ijp} + c2_{ij}X2_{ijp}) \\
 &+ \sum_{q=1}^{\|Q\|} \sum_{j=1}^N \sum_{i=1}^N (c3_{ij}Z1_{ijp} + c4_{ij}Z2_{ijp})
 \end{aligned}
 \tag{21}$$

where

- $X1_{ijp}$ =1, if the import container i is stored before the container j on the YC p ; =0, otherwise
- $X2_{ijp}$ =1, if the export container i is retrieved before the container j on the YC p ; =0, otherwise
- $Y1_{ijk}$ =1, if the import container i is transported before the container j on the YT k ; =0, otherwise
- $Y2_{ijk}$ =1, if the export container i is transported before the container j on the YT k ; =0, otherwise
- $Z1_{ijq}$ =1, if the import container i is unloaded before the container j on QC q ; =0, otherwise
- $Z2_{ijq}$ =1, if the export container i is loaded before the container j on the QC q ; =0, otherwise

IV. HIERARCHICAL SIMULATION-BASED OPTIMIZATION APPROACH

A. A HIERARCHICAL SIMULATION-BASED OPTIMIZATION FRAMEWORK (HSBOF)

A hierarchical simulation-based optimization framework (HSBOF), shown in Fig. 6, is used to develop simulation-based optimization approaches. It includes the three main layers, (1) load-balancing, (2) sequencing, and (3) simulation, which are introduced as follows:

- (1) Load-balancing layer: this layer aims to balance the workload for both QCs and YCs by using a load-balancing heuristic. Workload balance among available equipment is expected to provide a good foundation for the generation of good solutions.
- (2) Sequencing layer: this layer aims to form operation sequences for containers assigned to the same equipment. Heuristic/metaheuristics such as PSO, Random, SFLA, ISFLA, GA, and FA are used as the sequencing tools. The operational sequences of containers obtained in this layer will be used as inputs to the simulation layer.
- (3) Simulation and evaluation layer: based on the obtained operational sequences, this layer aims to simulate the YC, YT, and QC operations, meanwhile evaluating and finding the best one.

Also, the HSBOF includes functions such as constraints discovery, input, and output. The constraints discovery aims

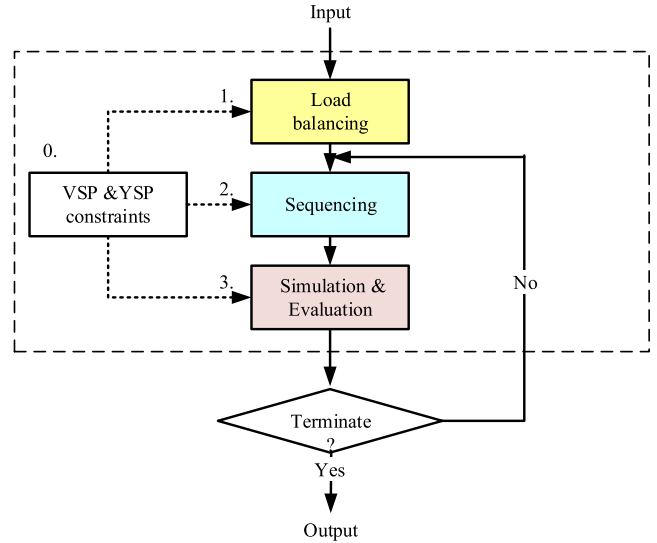


FIGURE 6. A hierarchical simulation-based optimization framework (HSBOF).

to discover operational constraints from YSP and VSP. The input is a basic function to get the necessary data for experiments. The output function is a basic function to output the best planning result. A termination check is used to terminate the simulation or routing the process back to the sequencing layer again. A total number of iterations is used as the termination condition.

The three core layers are further detailed as follows.

B. THE LOAD-BALANCING HEURISTIC

This layer uses a load-balancing heuristic (Algorithm 1) to balance the workload among equipment.

Algorithm 1 The Pseudo Code of the Load-Balancing Heuristic

- (1) SET N ; set $WL_e = 0$ ($e \in P \cup Q$)
- (2) ESTIMATE the upper workload limit as follow. $WL_limit = RD(N/E)$
- (3) SORT containers into a list L according to their bay numbers increasingly; set $j = 1$; $e = 1$.
- (4) REPEAT
- (5) Assign the i -th containers in L to the e ;
- (6) $WL_e = WL_e + 1$; // add 1 container
- (7) IF ($WL_e >= WL_limit$) // workload limit is reached or over
- (8) IF ($y_j \neq y_{j-1}$) //bay numbers are different
- (9) $e = e + 1$; // change to the next equipment
- (10) END IF
- (11) END IF
- (12) UNTIL $j = N$

In Algorithm 1, Step (1) initializes parameter values for N (the total number of import and export containers), E (the total number of equipment), and WL_e (current workload of equipment e), where $e \in P \cup Q$. Step (2) determines the

upper workload limit of each equipment. The function $\mathbf{RD}(\bullet)$ rounds a real value up to an integer. Step (5) assigns the first container in \mathbf{L} to the equipment e . Step (7) checks whether the upper limit (WL_limit) has been achieved? Step (8) checks whether the next container to be assigned is located at a different bay; If “yes” then change the assignment to the next equipment by using Step (9). This algorithm ensures containers of the same bay are assigned to the same equipment. It also separates equipment into different working spaces to avoid collision(s).

C. SEQUENCING METHODS

Heuristics/Meta-heuristics are used as different sequencing methods to generate alternative simulation-based optimization approaches. The standard SFLA and an improved SFLA (ISFLA) are introduced below.

1) STANDARD SFLA

Proposed by Eusuff and Lansey [11], the SFLA is a population-based metaheuristic mimicking the foraging behavior of frogs, which aims to find the most amount of food (the best position in a solution space). The goodness of a position can be measured by a predefined objective function. The jumps of frogs lead to search (exploration or exploitation) in the solution space. These frogs are separated into groups (termed memplexes) and each frog can be influenced by the elites (i.e., the best frog of the same group or the global best frog). Given m groups and F frogs, the best frog is assigned to the memplex 1; the 2nd best to the memplex 2; the m^{th} best to the memplex m ; the $m + 1^{\text{th}}$ best back to the memplex 1, and so on. Each group finally contains F/m frogs. However, only a specified number of q frogs ($q < F/m$) in a group are allowed to evolve. These frogs are sampled from the original memplex to a sub-memplex by using the triangular distribution shown in Eq. (22).

$$P_f = 2(n + 1 - f)/n(n + 1) \quad f = 1, \dots, n \quad (22)$$

Let $X_f(t)$ and $X_b(t)$ be the positions of frogs f and b , where the b indicates the best frog in the sub-memplex, then the leaping distance for the frog f is determined by Eq. (23).

$$\Delta D_f(t) = \begin{cases} \min \{ \text{int} \{ R(X_b(t) - X_f(t)), S_m \} \} & \text{for + step} \\ \max \{ \text{int} \{ R(X_b(t) - X_f(t)), -S_m \} \} & \text{for - step} \end{cases} \quad (23)$$

The R is a random number within the range $[0,1]$; the S_m is the maximum leaping step allowed. With a leaping distance, the next position of frog f is determined by Eq. (24).

$$X_f(t + 1) = X_f(t) + \Delta D_f(t) \quad (24)$$

If the next position is better then accept this position; Otherwise, replace the $X_b(t)$ with the $X_g(t)$ in Equation (23) and jump this frog again. If this jump finds a better position then accept this position; Otherwise, give a random jump for this f . Repeat the above procedure until the last frog in the

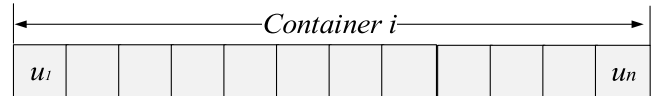


FIGURE 7. The encoding scheme of frog position.

sub-memplex is processed. Then, continue the next round of local searches for frogs. When all rounds of local searches are completed, go to the next iteration. Before starting the next iteration, a reshuffle of frogs into groups and then the sub-groups is necessary. The above procedure repeats until the satisfaction of the termination condition [44].

2) ISFLA

The improved SFLA (ISFLA) is employed by Simulation(ISFLA) to generate alternative sequences of containers.

- Position Representation of a frog

Fig. 7 shows the position scheme used for frogs in this research. The position scheme has a variable dimension depending on the number of containers (n) assigned to specific equipment (YC or QC). In this position scheme, each u_i indicates the container No. assigned to the container i . Given the position vector $[4,3,2,1,5,6]$ with $n = 6$ as an example, it indicates that container 4 is assigned with order 1; container 3 with order 2, and so on.

- The novel features of ISFLA

The ISFLA includes the following features:

a: DECREASING THE GROUP NUMBER WITH INCREASING GROUP SIZE

An exploration-to-exploitation strategy is employed by the ISFLA. Eq. (25) implements this strategy by changing the number of groups at each iteration.

$$G(t) = \mathbf{RD}(\sqrt{\rho} + \sqrt{\rho}/2 - t \cdot \sqrt{\rho}/Total_iteration) \quad (25)$$

The $\mathbf{RD}(\bullet)$ is a function rounding a decimal value up to its nearest integer. The ρ indicates the total number of frogs in the swarm, which decreases iteratively until to a bottom line. The most number of groups appears at the first iteration ($t = 1$), giving the maximum momentum for exploration. The last iteration has the least momentum for exploration, but the maximum momentum for exploitation, as each group has the most number of forges in exploiting the target frog which can be the best in the same group or the global best frog.

b: DISCRETE OPERATORS WITH A SELF-ADAPTIVE JUMP

To make a better jump, discrete operators “~”, “⊕” and “⊗” are used by the ISFLA to operate on discrete vectors.

Based on their current positions, frogs jump toward the target frog adaptively. Three steps are initiated for this. First, measure the total distance between the two frogs; Second, determine the adaptive leaping distance for the frog; Third, jump this frog to the next position for local search. Fig. 8 illustrates an example that gives $X_f(t) = [1, 2, 3, 4, 5, 6]$ and

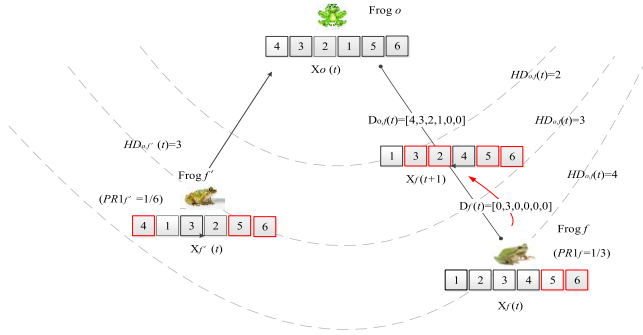


FIGURE 8. Frogs i and j jump towards the target frog o .

$X_o(t) = [4, 3, 2, 1, 5, 6]$ as the position vectors of the frogs f and o , respectively.

First, we define the total-distance vector, denoted $D_{o,f}(t)$, between the frogs f and o , in Eq. (26).

$$D_{o,f}(t) = [X_{o,k}(t) \sim X_{f,k}(t); k = 1, \dots, D] \quad (26)$$

where $X_{o,k}(t) \sim X_{f,k}(t)$ the k -th element in $D_{o,f}(t)$, determined by the k -th element in $X_o(t)$ and the k -th element in $X_f(t)$ as well as the operator “ \sim ” which works as follow.

$$X_{o,k}(t) \sim X_{f,k}(t) = \begin{cases} 0, & \text{if } X_{o,k}(t) = X_{f,k}(t) \\ X_{o,k}(t), & \text{if } X_{o,k}(t) \neq X_{f,k}(t) \end{cases} \quad (27)$$

Given that $X_f(t) = [1, 2, 3, 4, 5, 6]$ and $X_o(t) = [4, 3, 2, 1, 5, 6]$, based on Eqs. (27) and (28), we derive the $D_{o,f}(t)$ as follow.

$$\begin{aligned} D_{o,f}(t) &= [4, 3, 2, 1, 5, 6] \sim [1, 2, 3, 4, 5, 6] \\ &= [4, 3, 2, 1, 0, 0]. \end{aligned}$$

Now, we continue to determine the adaptive leaping distance for the frog f , denoted as $\tilde{D}_f(t)$, by using Eq. (28).

$$\tilde{D}_f(t) = D_{o,f}(t) \oplus AD_f(t) \quad (28)$$

The $AV_j(t)$ is a binary vector, with its k -th element being determined by Eq. (29).

$$AD_{f,k}(t) = \begin{cases} 0, & \text{if } Rand() \geq PR1_f(t) \\ 1, & \text{if } Rand() < PR1_f(t) \end{cases} \quad (29)$$

The $Rand()$ is a random number and the $PR1_f(t)$ is a threshold controlling the generation probability of the binary value 1 for the frog f and it is determined by Eq. (30).

$$PR1_f(t) = \begin{cases} \frac{HD_{o,f}(t) - 2}{D}, & \text{if } HD_{o,f}(t) > 2 \\ 0, & \text{if } HD_{o,f}(t) \leq 2 \end{cases} \quad (30)$$

The $HD_{o,f}(t)$ is termed Hamming Distance which counts the total number of different elements between two vectors and can be determined by Eq. (31).

$$HD_{o,f}(t) = D - \sum_{k=1}^D (X_{o,k}(t) \sim X_{f,k}(t)) / X_{o,k}(t) \quad (31)$$

Using Eqs. (30) and (31), we can obtain $PR1_f(t) = 2/6$ and $HD_{o,f}(t) = 4$, respectively.

However, to derive the $\tilde{D}_f(t)$, it needs to define the operation of the operator “ \oplus ” on $D_{o,f}(t)$ and $AD_{f,k}(t)$. The operator “ \oplus ” works as follows.

$$D_{o,f}(t) \oplus AD_{f,k}(t) = \begin{cases} D_{o,f}(t), & \text{if } AD_{f,k}(t) = 1 \\ 0, & \text{if } AD_{f,k}(t) = 0 \end{cases} \quad (32)$$

Suppose that $AD_f(t) = [0, 1, 0, 0, 0, 0]$, based on Eqs. (28)–(32), then we can obtain $\tilde{D}_f(t)$.

$$\begin{aligned} \tilde{D}_f(t) &= [4, 3, 2, 1, 0, 0] \oplus [0, 1, 0, 0, 0, 0] \\ &= [0, 3, 0, 0, 0, 0] \end{aligned}$$

After determining $\tilde{D}_f(t)$, the next position of the frog f is determined by Eq. (33).

$$X_f(t+1) = X_f(t) \otimes \tilde{D}_f(t), \quad (33)$$

where operator “ \otimes ” works in this way. First, it finds the first non-zero value out of the $\tilde{D}_f(t)$ and replaces the value with the same position in the vector $X_f(t)$. Then, the replaced value further takes the position of the non-zero value in the $\tilde{D}_f(t)$. Finally, the resulting $X_f(t)$ becomes the $X_f(t+1)$.

Based on Eq. (33) and the operator “ \otimes ” then we can derive the next position of for the frog f as follow.

$$\begin{aligned} X_f(t+1) &= X_f(t) \otimes \tilde{D}_f(t) \\ &= [1, 2, 3, 4, 5, 6] \otimes [0, 3, 0, 0, 0, 0] \\ &= [1, 3, 2, 4, 5, 6] \otimes [0, 0, 0, 0, 0, 0] \\ &= [1, 3, 2, 4, 5, 6] \end{aligned}$$

c: ADVANCED MOVEMENT MECHANISMS FOR FROGS

Also, the ISFLA employs two mechanisms, *Tabu jump* and *neighborhood search*, to better move a frog. The *Tabu jump* is used to prevent a frog f from jumping to the target frog o with one step. This will waste one local search as this position has been already visited by the target frog. This mechanism works in this way. First, it measures the $HD_{o,f}(t)$ between the frogs o and f . Then, the following rule is employed.

Rule: If the $HD_{o,f}(t) \leq 2$ then stop the frog from jumping toward the target frog o .

This rule is implemented by stopping generating binary value 1 for the vector $AD_f(t)$. However, as *Tabu jump* stops jumping for a frog, the frog can also lose one local search. For improvement, the ISFLA allows such a frog to make a *neighborhood search*, which exchanges two randomly-selected elements in the position vector of the frog f .

d: THE LOGIC FLOW OF THE ISFLA

Algorithm 2 shows the main logic flow of the ISFLA.

D. THE SIMULATION MODEL

Referring to [45] [46], a simulation model termed Timed Predicate/Transition net (TP/T net) is developed for discrete-event simulation.

Algorithm 2 The Main Logic Flow of the ISFLA

```

1  Set parameter values ( $N, Rm, P, iterations, l\_iter,$ 
   etc.)
2  Initialize frog positions
3  FOR ( $t = 1; t \leq iterations; t++$ ){
4    Calculate Z values for all frogs using Eq. (21).
5    Rank frogs according to their Zs
6    Generate group number  $G(t)$  for this iteration.
7    FOR ( $li = 1; li \leq l\_iter; li++$ )
8      FOR ( $int j = 1; j \leq G(t); j++$ )
9        FOR ( $int k = 1; k \leq$ 
           the_number_of_frogs_in_j;  $k++$ )
10         IF frog  $k$  is not the best frog in the group  $j$ 
11           Jump frog  $k$  one step towards the best
             frog in  $j$ 
12           Estimate the Z value for this position
             using Eq. (21).
13           IF Z value of this movement is
             improved
14             Store the Z value for this frog  $k$ 
15             Store the current position for
             this frog  $k$ 
16           ELSE
17             Jump the frog  $k$  toward the global
             best frog
18             Calculate Z value using Eq. (21).
19             IF Z value of this movement
             is improved
20               Store the Z value for this frog  $k$ 
21               Store the current position for
             this frog  $k$ 
22             ELSE
23               Let the frog  $k$  have a random
             jump
24             END IF
25           END IF
26           Compare and store the global best
             position
27         END IF
28       END FOR  $j$ 
29     END FOR  $li$ 
30   END FOR  $t$ 

```

Definition 1: A Timed Predicate/Transition net (TP/T net) is defined as 6- tuple:

$$TP/T \text{ net} = (P, T, A, \Sigma, L, F)$$

where

P: a set of predicates; $P = P_t \cup P_{nt}$, where the P_t is a set of timed predicates and the P_{nt} is a set of predicates without consuming time. $P_i \in P_t$ or P_{nt} , $P_t \cap P_{nt} = \emptyset$.

T: a set of transitions $\{T_1, \dots, T_n\}$.
A: a set of arcs.
 Σ : is a structure consisting of individual tokens (Tk_i), operations (O_j), and relations (R_k), i.e., $\Sigma = (Tk_1, \dots, Tk_i; O_1, \dots, O_j; R_1, \dots, R_k)$.
L: labeling of arcs with a formal sum of n attributes of token's variables; zero-attributes indicating a token with no argument.
F: a set of inscriptions in transitions; being logical formula built from the operations and relations of the structure Σ ; Variables occurring free in a formula have to occur at an adjacent arc.
M: a marking of P with formal sums of n -tuples of individual tokens. M_0 indicates an initial marking of tokens.

Fig. 9 depicts a TP/T net model with the following components: $P = \{\text{Open_task}, \text{Close_task}, \text{Avail_R}, \text{Using_R}\}$ (in which **Using_R** is a timed predicate and the others are non-timed predicates); $A = \{A_1, \dots, A_{18}\}$, $T = \{T_1, T_2, T_3, T_4, T_5, T_6\}$, and $M_0 = C_tokens \cup R_tokens$. The container tokens $C_token = \{ \langle 1, 1, 1, 0 \rangle, \dots, \langle 1, \|TE\|, 1, 0 \rangle \} \cup \{ \langle 2, 1, 3, 0 \rangle, \dots, \langle 2, \|TI\|, 3, 0 \rangle \}$ and resource tokens $R_token = \{ \langle 1, 1, 1, 0 \rangle, \dots, \langle 1, p, 2, 0 \rangle \}, \{ \langle 2, 1, 1, 0 \rangle, \dots, \langle 2, k, 1, 0 \rangle \},$ and $\{ \langle 3, 1, 1, 0 \rangle, \dots, \langle 3, q, 2, 0 \rangle \}$ are staying with the **Open_task** and **Avail_R** predicates, respectively. Each container token contains four attributes, C_TP, CID, R_TP, and CT, corresponding to container type, container ID, resource type required, and available time, respectively. The C_TP=1 and R_TP=1 are initialized for an import container token while the C_TP=2 R_TP=3 are initialized for an export container token. Each resource token includes four attributes, R_TP, RID, SM, and R_T, corresponding to resource type, resource id, service mode (1: import; 2: export), and available time, respectively. The values of R_TP=1, 2, and 3 indicate the resource type of YC, YT and QC, respectively. The SM=1 indicates service mode for import container and SM=2 indicates service mode for export containers. The $\langle C_TP, CID, R_TP, CT \rangle + \langle R_TP, RID, SM, RT \rangle$ on A_6 is a formal sum. The “R_TP<3”, “C_TP=1”, and “SM=1” in the transition T1 are logical formula (conditions) for triggering this transition.

Each transition in the TP/T net model corresponds to a discrete event that can be triggered by a rule [43]. The rules used to trigger transitions T1 to T6 are detailed as follows:

Rule 1: If (one import container token in the **Open_task** predicate and one resource token in the **Avail_R** predicate & $R_TP < 3$ & $C_TP = 1$ & $SM = 1$) then trigger the transition T1. After triggering the transition T1, one import container token and one resource token will be transitioned to the **Using_R** predicate.

Rule 2: If (one export container token in the **Open_task** predicate and one resource token in the **Avail_R** predicate & $R_TP > 1$ & $C_TP = 2$ & $SM = 2$) then trigger the T2. The T2

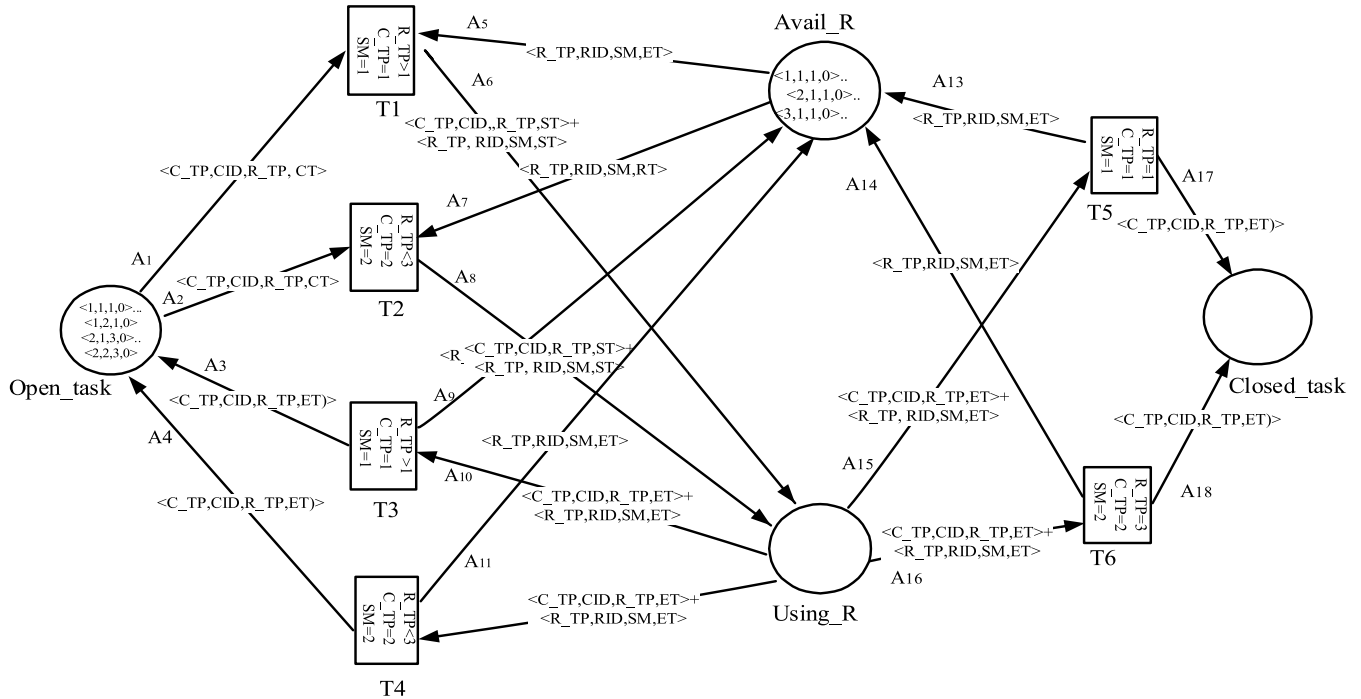


FIGURE 9. A simulation model based on Predicate/Transition net.

will transit the export container token and the resource token to the **Using_R** predicate.

Rule 3: If (one import container token+one resource token in the **Using_R** predicate & $R_TP < 3$ & $C_TP = 1$ & $SM = 1$) then trigger the T3. After triggering the T3, one import container token will be transitioned back to the **Open_task** predicate and one resource token will be returned to the **Avail_R** predicate.

Rule 4: If (one export container token+one resource token in the **Using_R** predicate & $R_TP > 1$ & $C_TP = 2$ & $SM = 2$) then trigger the T4. After triggering the T4, the export container token will be transitioned back to the **Open_task** predicate and the resource token will be returned to the **Avail_R** predicate.

Rule 5: If (one import container token+one resource token in the **Using_R** predicate & $R_TP = 1$ & $C_TP = 1$ & $SM = 1$) then trigger the T5. After triggering the T5, the import container token will be transitioned to the **Close_task** predicate and the resource token will be returned to the **Avail_R** predicate.

Rule 6: If (one export container token+one resource token in the **Using_R** predicate & $R_TP = 3$ & $C_TP = 2$ & $SM = 2$) then trigger the T6. After triggering the T6, the export container token will be transitioned to the **Close_task** predicate and the resource token will be returned to the **Avail_R** predicate.

The following simulation procedure is used.

1) Make initial marking M_0 . Let export and import container tokens stay with the **Open_task** predicate and all resource tokens stay with the **Avail_R** predicate.

- 2) Trigger the enabled transitions (rules) in this model. The T1 is used to trigger operations (YC, YT, or QC) for an import container token. The R_TP specifies the operation to perform. An import container uses QC, YT, and YC sequentially. The T2 is used for export container tokens which will use YC, YT, and QC sequentially. The trigger of T1 or T2 will send one container token $\langle C_TP, CID, R_TP, ST \rangle$ and one resource token $\langle R_TP, RID, SM, ST \rangle$ to the **Using_R** predicate and the start event time is $\text{Max}\{CT, RT\}$, where the CT is container available time and RT is resource available time. The load-balancing heuristic and sequence method are used to determine the assignment and sequence of containers on available resources.
- 3) After a resource usage duration at the **Using_R** predicate, the transitions T3, T4, T5, or T6 will be triggered, depending on the conditions (Rules). If T3 or T4 is triggered then the container token $\langle C_TP, CID, R_TP, ET \rangle$ will go back to the **Avail_task** predicate and the resource token $\langle R_TP, RID, SM, ET \rangle$ will return to the **Avail_R** predicate. Meanwhile, the R_TP will be decreased by 1 for an import container token to indicate the next resource type (operation) required (for an export container the R_TP is increased by 1). If Rule 5 is satisfied then it will trigger the T5 which will transition the import container token to the **Close_task** predicate and return the resource token $\langle R_TP, RID, SM, ET \rangle$ to the **Avail_R** predicate. The T6 is used for export containers.
- 4) Check whether all container tokens are staying with the **Close_task** predicate? When all import and export

TABLE 2. The parameter setting for YC, YT, and QC.

	YSP	YC	YT	QC	VSP
X	40	$\ P\ $ 2	$\ K\ $ 5	$\ Q\ $ 2	$\ B\ $ 10
Y	10	v_x 1 M/s	ETK 300s	v_b 1 M/s	$\ R\ $ 10
Z	10	v_y 1 M/s	ITK 300s	v_r 1 M/s	$\ T\ $ 10
H	11	v_z 1 M/s		v_t 1 M/s	
				CH 30 M	

X : the total number of bays in a block; Y : the total number of rows in a block; Z : the total number of tiers in a block; H : the tier used to transfer a container $H=Z+1$; $\|P\|$: the total number of YCs in a block; $\|K\|$: the total number of YTs; $\|Q\|$: the total number of QCs; CH : the height of QC (meters) to reach YT; $\|B\|$: the total number of bays in a vessel; $\|R\|$: the total number of rows in a bay; $\|T\|$: the number of tiers in a vessel.

TABLE 3. The parameter setting for different approaches.

Method	Simulation (Random)	Simulation (FA)	Simulation (GA)	Simulation (SFLA)	Simulation (PSO)	Simulation (ISFLA)
N	20,100,200	20,100,200	20,100,200	0,100,200	20,100,200	20,100,200
P	400	400	400	400	400	400
$G(t)$	121	121	121	121	121	121
q	-	-	-	11	-	Eq. (25)
n_ls	-	-	-	6	-	$P/G(t)$
IT	1	1	1000	4	1000	4
w	-	-	-	250	0.5	-
\bar{V}	-	-	-	-	2	-
\underline{V}	-	-	-	-	-2	-
Rm	-	-	0.3	-	-	-
Rc	-	-	0.4	-	-	-

N : the number of containers; P : population; $G(t)$: the number of groups at the iteration t ; q : the number of frogs in each group; n_ls : the number of local searches; IT : the total number of iterations; w : weight; \bar{V} : the upper bound of velocity; \underline{V} : the lower bound of velocity; Rm : mutation rate; Rc : cross-over rate; -: not used

container tokens stay with the **Closed_task** predicate it means all operations have been completed. The STs and ETs of resource instances will be recorded during the simulation process. If “yes” go to Step 5); Otherwise, go to step 2).

5) Stop running and output the best panning result.

V. NUMERICAL EXPERIMENTS

Based on the proposed framework, we have developed six heuristic/metaheuristic-based simulation optimization approaches, namely Simulation(Random), Simulation(SFLA), Simulation(FA), Simulation(GA), Simulation(PSO), and Simulation(ISFLA), by using Java programming language. To investigate their effectiveness, experiments were conducted on a personal computer with an Intel PENTIUM CPU 2117U (64 bits and 1.8 GHz) and 4GB DRAM.

A. PARAMETER VALUES SETTING

Table 2 shows the parameter values set for experiments. Table 3 shows the parameter values for different approaches.

The storage block uses the following parameter values: $X = 50$; $Y = 10$; $Z = 4$; $H = Z+1$. The YCs are set with the following parameter values: $\|P\| = 2$; $v_x = v_y = v_z = 1$ m/sec. The YTs are set with the following parameter values: $\|K\| = 5$ and $ITK=ETK=300$ seconds. The

QCs are set with the following parameter values: $\|Q\| = 2$; $v_b = v_r = v_t = 1$ m/sec. The VSP are set with the following parameter values: $\|B\| = 10$; $R = 10$; $\|T\| = 10$.

The $N = 20, 100, 200, 400$ as well as $P = 121$ are common parameter values used for all approaches. The $N = 20$ is used for a small-sized example. In addition, different approaches may have some different parameter values. For example, Simulation(Random) is set with $IT=1000$; Simulation(GA) is set with $IT=1000$, $Rm=0.3$, and $Rc=0.4$; Simulation(PSO) is set with $IT=1000$, $w = 0.5$, $\bar{V} = 2$, $\underline{V} = -2$; Simulation(ISFLA) is set with $n = P/G(t)$, $n_ls=4$, and $IT=250$.

B. EXPERIMENT DATA GENERATION

Experimental instances were generated with the parameter values defined in Table 2 considered. For example, in Table 4 the data in rows 3, 4, and 5 constitute the (b_j, r_j, t_j) which represents the stowage position of container j in the VSP (vessel). These values are within these ranges $1 \leq b_j \leq \|B\|$, $1 \leq r_j \leq \|R\|$, $1 \leq t_j \leq \|T\|$. Similarly, the data in rows 17, 18, and 19 constitute the (x_j, y_j, z_j) which represents the storage position of container j in the YSP (block). The x_j, y_j, z_j are confined in these ranges $1 \leq x_j \leq X$; $1 \leq y_j \leq Y$; $1 \leq z_j \leq Z$, where the values of the X, Y, and Z are defined in Table 2. The position data of each container is generated by the computer automatically. For an import container, the container type is set to 1 ($C_TP=1$). In Tables 4 and 5, these marked rows contain the data generated by the computer automatically, while other unmarked rows 6–16 are those relating to decision variables whose values are determined by the approach used.

Table 5 shows the data of export containers. The rows 3, 4 and 5 constitute the current storage position (x_j, y_j, z_j) in the YSP while the rows 17, 18, and 19 represent the future stowage position (b_j, r_j, t_j) in the VSP for the container j to be transported by the ship.

C. A SMALL-SIZED EXAMPLE

This subsection illustrates the inputs and outputs of a small-sized problem ($N = 20$) obtained from the Simulation(ISFLA).

1) INPUT DATA

The input data are generated by the computer automatically, based on the parameter values set in Tables 2 and 3. These input data include the container ID; container type ($C_TP=1$: import container; $C_TP=2$: export container); the vessel storage positions (B: bay number; R: row number; T: tier number); the yard storage position (X: bay number; Y: row number; Z: tier number), etc.

2) VSP AND YSP CONSTRAINTS

Based on the input VSP data, the VSP constraints for both import and export containers and the YSP constraints for both import and output containers are generated automatically. These constraint data are shown in Tables 6 and 7 with $PC=600$ seconds used as the penalty cost.

TABLE 10. Comparisons of different approaches.

	Random	GA	FA	PSO	SFLA	ISFLA
Z(s)	5926.2	5915.7	5630	5401.7	5437.2	5397.4
iteration	378	566	1	943	142	245
Time(s)	0.4	2.5	0.09	2.8	3.8	6.1

Z: Z values

the QCs, the export containers 17,11,6,9,20,1,15,13,10, and 16 are assigned to QC1 with this loading sequence while the export containers 8,18,5,14,7,12,3,2,4, and 19 are assigned to QC2 with this loading sequence. Both QC1 and QC2 are workload balanced, each loading 10 containers.

Table 10 shows the data of the best solutions found by different approaches. It is found that the Simulation(ISFLA) approach finds the best solution, but with the most computational cost.

D. EXTENSIVE EXPERIMENTS

Extensive experiments of different problem sizes have been further conducted to investigate the effectiveness of different approaches. Table 11 shows the results obtained from these different approaches. They are summarized as follows:

- Fig. 10 shows the comparative results of different approaches at the problem size $N = 100$. It is found that Simulation(ISFLA) outperforms Simulation(PSO) with the edge 2.1%, Simulation(SFLA) with the edge 2.6% Simulation(GA) with the edge 2.9%, Simulation(FA) with the edge 3.3%, Simulation(Random) with the edge 4.9%, in terms Z value.
- Fig. 11 shows the comparative results of different approaches with the problem size $N = 200$. It is found that Simulation(ISFLA) outperforms Simulation(PSO) with the edge 2.1%, Simulation(SFLA) with the edge 2.9%, Simulation(GA) with the edge 2.9%, Simulation(FA) with the edge 3.2%, Simulation(Random) with the edge 5.0%, in terms of Z value.
- Fig. 12 shows the comparative results of different approaches at the problem size $N = 400$. It is found that Simulation(ISFLA) outperforms Simulation(PSO) with the edge 2.5%, Simulation(SFLA) with the edge 3.0%, Simulation(GA) with the edge 3.1%, Simulation(FA) with the edge 3.4%, Simulation(Random) with the edge 4.7%, in terms of Z value.

E. STATISTICALLY TEST

The statistical t -test is used to test the Hypotheses H_0 and H_1 at the significance level of $\alpha = 0.05$. The H_0 is a null hypothesis assuming that the average Z values obtained from comparing approaches have no significant difference; H_1 assumes that these average Z values are different.

Table 12 shows test results of comparing each approach to Simulation(ISFLA). The symbol “+” that Simulation(ISFLA) is better; the symbol “-” indicates that Simulation(ISFLA) is inferior; the symbol “N” indicates no difference. It shows the p -values of each comparison are less than 0.005. For example, at the problem

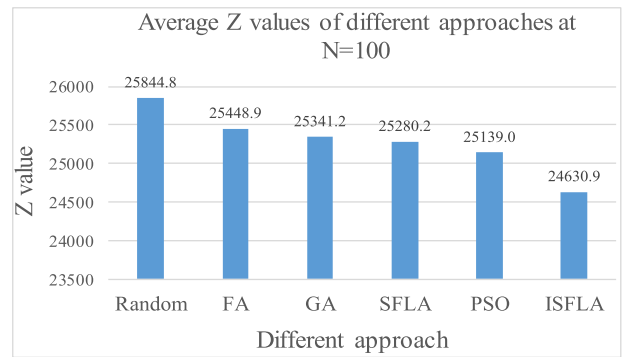


FIGURE 10. Z values of different approaches at N=100.

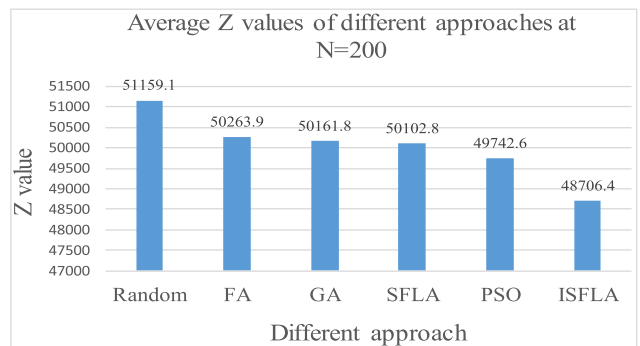


FIGURE 11. Z values of different approaches at N=200.

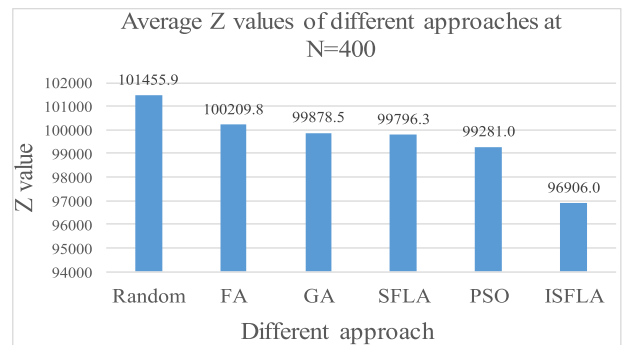


FIGURE 12. Z values of different approaches at N=400.

size $N = 100$, the comparisons of Simulation(ISFLA) with Simulation(Random), Simulation(FA), Simulation(GA), and Simulation(SFLA), Simulation(PSO) have the following p -values, 1.34839E-6, 3.91071E-8, 4.84051E-5, 0.0002207, 5.40429E-7, respectively, which lead to rejection of H_0 and the acceptance of H_1 . Simulation(ISFLA) is concluded to outperform the others.

F. ANALYSIS AND DISCUSSIONS

- While balancing workload, the load-balancing heuristic meanwhile can separate YCs and QCs into different working spaces. This avoids crossover of each other. Subsequently, a heuristic/metaheuristic is used to initiate alternative operational sequences for containers assigned on each equipment. Finally, the simulation

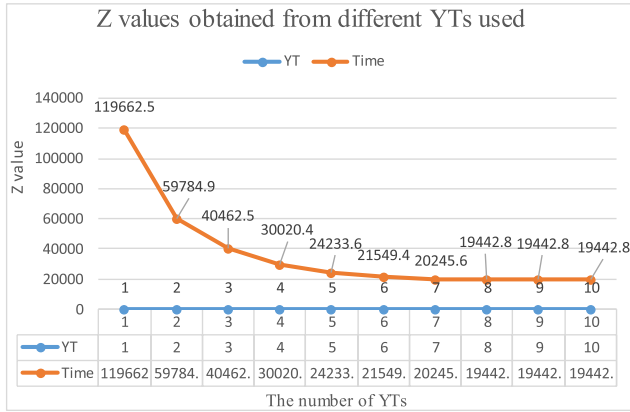


FIGURE 13. Sensitivity analysis of using different number of YTs.

In addition, the YTs are also with a balanced workload and well utilized with no spare time.

- Fig. 13 shows the experimental results of sensitivity analysis using different numbers of YTs, together with 2 QCs and 2 YCs in handling 100 import and 100 export containers for a ship. The use of one YT leads to a huge cost ($Z=119662.5$) that can be further reduced by increasing the number of YTs from 2 to 8. However, the numbers of YTs exceeding 8 are found unable to reduce the Z value (19442.8) further, meaning that the best number of YTs to cooperate with the 2 QCs and 2 YCs for this ship is 8. If the number of YTs is below 8 then improvement is available by increasing this number.
- In our experiments, Simulation(ISFLA) is found better than the other approaches in terms of makespan. The advantage is believed to result from the novel features of the ISFLA, such as decreasing the number of frog groups, self-adaptive leaps, and the advanced movement mechanism. In addition, the ISFLA is with more population advantage due to allowing all frogs to evolve [47].
- Containers can be treated as groups or individuals when planning [35] [44]. Though the treatment of containers as groups makes the planning easier, it cannot facilitate real operations due to rough resolution. By treating containers as individuals, our approach can offer a high-resolution solution that makes implementation easier.
- Effective software for the integration of simulation and optimization effectively is still unavailable [10], which is considered one research gap in the existing literature. To address this gap, we have proposed a hierarchical simulation-based optimization framework (HSBOF) for developing alternative heuristic/metaheuristic and simulation-based optimization approaches to deal with the integrated scheduling problem of QCs, YTs, and YCs. This helps container terminal planners to pursue global optimality, instead of local optimality.
- Managers should note that local optimality is not certainly beneficial as it may form bottleneck(s) in the

downstream operations. It is important to consider all MHE as well as both import and export containers. To achieve global optimality, the simulation-based optimization approach is useful.

VI. CONCLUSION

This research proposed a hierarchical simulation-based optimization framework (HSBOF) for developing heuristic/metaheuristic-based simulation optimization approaches to deal with the simultaneous YCSP, YTSP, and QCSP, taking YSP, VSP, and their constraints into consideration. Specifically, Simulation(Random), Simulation(ISFLA), Simulation(PSO), Simulation(SFLA), Simulation(GA), and Simulation(FA) have been developed in this research. The objective is to minimize costs of makespan as well as penalty costs of constraint violation. A small-size instance has been used to illustrate the planning results, and extensive experiments of different sizes ($N = 100, 200, \text{ and } 400$) have been conducted to investigate the effectiveness of different approaches. Finally, the statistical t -tests have been used to validate the experimental results. The Simulation(ISFLA) is found to outperform the others.

The contributions of this research are listed as follows: (1) we have proposed a hierarchical simulation-based optimization framework (HSBOF) for the integration of simulation and optimization. Based on this framework, various heuristic/metaheuristic-based simulation optimization approaches have been developed; (2) these approaches have been successfully applied to solve the YCSP, YTSP, and QCSP simultaneously, taking the YSP and VSP and their constraints into consideration; (3) we have proposed an improved version (ISFLA) to advance the standard SFLA; (4) we have performed experiments to investigate the effectiveness of different approaches, including Simulation(ISFLA), Simulation(PSO), Simulation(SFLA), Simulation(GA), Simulation(FA) and Simulation(Random).

The following research directions are suggested. (1) this study can be extended to include multiple ships; (2) the uses of other methods such as Bee-inspired Algorithms (BA), Bacterial Foraging Optimization (BFO), Fish Swarm Optimization (FWO), and Cuckoo Search Algorithm (CSA), Whale Optimization Algorithm (WOA) as the sequencing methods can be investigated in the future research; (3) The inclusion of transshipment containers; (5) the inclusion of storage allocation problems; (6) the inclusion of 3D storage plan in the vessel; (7) the consideration of 40-foot containers.

REFERENCES

[1] United Nations Conference on Trade and Development. (2019). *Review of Maritime Transport*. [Online]. Available: http://unctad.org/en/publicationsLibrary/rmt2019_en.pdf

[2] H.-P. Hsu and C.-N. Wang, "Resources planning for container terminal in a maritime supply chain using multiple particle swarms optimization (MPSO)," *Mathematics*, vol. 8, no. 5, p. 764, May 2020, doi: 10.3390/math8050764.

[3] D. Steenken, S. Voss, and R. Stahlbock, "Container terminal operation, and operations research: A classification and literature review," *OR Spectr.*, vol. 26, no. 1, pp. 3–49, 2004.

- [4] C. Bierwirth and F. A. Meisel, "A survey of berth allocation and quay crane scheduling problems in container terminals," *Eur. J. Oper. Res.*, vol. 202, pp. 615–627, May 2010.
- [5] A. Imai, H. C. Chen, E. Nishimura, and S. Papadimitriou, "The simultaneous berth and quay crane allocation problem," *Transp. Res. E, Logistics Transp. Res.*, vol. 44, no. 5, pp. 900–920, Sep. 2008.
- [6] H.-P. Hsu, "A HPSO for solving dynamic and discrete berth allocation problem and dynamic quay crane assignment problem simultaneously," *Swarm Evol. Comput.*, vol. 27, pp. 156–168, Apr. 2016.
- [7] B. Raa, W. Dullaert, and R. Van Schaeren, "An enriched model for the integrated berth allocation and quay crane assignment problem," *Expert Syst. Appl.*, vol. 38, no. 11, pp. 14136–14147, Oct. 2011.
- [8] H. J. Carlo, I. F. A. Vis, and K. J. Roodbergen, "Storage yard operations in container terminals: Literature overview, trends, and research directions," *Eur. J. Oper. Res.*, vol. 235, no. 2, pp. 412–430, Jun. 2014.
- [9] D. Kizilay and D. T. Eliyi, "A comprehensive review of quay crane scheduling, yard operations and integrations thereof in container terminals," *Flexible Services Manuf. J.*, vol. 33, no. 1, pp. 1–42, Mar. 2021, doi: [10.1007/s10696-020-09385-5](https://doi.org/10.1007/s10696-020-09385-5).
- [10] C. Zhou, N. M. Ma, X. Cao, L. H. Lee, and E. P. Chew, "Classification and literature review on the integration of simulation and optimization in maritime logistics studies," *IJSE Trans.*, vol. 53, no. 10, pp. 1157–1176, 2021, doi: [10.1080/24725854.2020.1856981](https://doi.org/10.1080/24725854.2020.1856981).
- [11] M. M. Eusuff and K. E. Lansey, "Optimization of water distribution network design using the shuffled frog leaping algorithm," *J. Water Resources Planning Manage.*, vol. 129, no. 3, pp. 210–225, 2003.
- [12] A. T. Azevedo, L. L. de Salles Neto, A. A. Chaves, and A. C. Moretti, "Solving the 3D stowage planning problem integrated with the quay crane scheduling problem by representation by rules and genetic algorithm," *Appl. Soft Comput.*, vol. 65, pp. 495–516, Apr. 2018.
- [13] H.-P. Hsu, H.-H. Tai, C.-N. Wang, and C.-C. Chou, "Scheduling of collaborative operations of yard cranes and yard trucks for export containers using hybrid approaches," *Adv. Eng. Informat.*, vol. 48, Apr. 2021, Art. no. 101292, doi: [10.1016/j.aei.2021.101292](https://doi.org/10.1016/j.aei.2021.101292).
- [14] M. Fazli, A. M. Fathollahi-Fard, and G. Tian, "Addressing a coordinated quay crane scheduling and assignment problem by red deer algorithm," *Int. J. Eng.*, vol. 32, no. 8, pp. 1186–1191, 2019.
- [15] D. Sun, L. Tang, and R. Baldacci, "A benders decomposition-based framework for solving quay crane scheduling problems," *Eur. J. Oper. Res.*, vol. 273, no. 2, pp. 504–515, Mar. 2019.
- [16] S. Y. Huang and Y. Li, "A bounded two-level dynamic programming algorithm for quay crane scheduling in container terminals," *Comput. Ind. Eng.*, vol. 123, pp. 303–313, Sep. 2018.
- [17] N. Rouky, M. N. Abourraja, J. Boukachour, D. Boudebous, A. E. H. Alaoui, and F. E. Khoukhi, "Simulation optimization based ant colony algorithm for the uncertain quay crane scheduling problem," *Int. J. Ind. Eng. Comput.*, vol. 10, no. 1, pp. 111–132, 2019.
- [18] 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, doi: [10.1016/S0377-2217\(03\)00133-4](https://doi.org/10.1016/S0377-2217(03)00133-4).
- [19] I. E. Nielsen, N. A. D. Do, V. D. Nguyen, P. Nielsen, and Z. Michna, "Reducing truck emissions in import operations at container terminal—A case study in a Singaporean port," in *Technology Management for Sustainable Production and Logistics*. 2015, pp. 133–151, doi: [10.1007/978-3-642-33935-6_7](https://doi.org/10.1007/978-3-642-33935-6_7).
- [20] K. H. Kim and J. W. Bae, "A look-ahead dispatching method for automated guided vehicles in automated port container terminals," *Transp. Sci.*, vol. 38, no. 2, pp. 121–255, 2004.
- [21] E. K. Bish, F. Y. Chen, Y. T. Leong, B. L. Nelson, J. W. C. Ng, and D. Simchi-Levi, "Dispatching vehicles in a mega container terminal," *OR Spectr.*, vol. 27, no. 4, pp. 491–506, Aug. 2005.
- [22] W. C. Ng, K. L. Mak, and Y. X. Zhang, "Scheduling trucks in container terminals using a genetic algorithm," *Eng. Optim.*, vol. 39, no. 1, pp. 33–47, Jan. 2007, doi: [10.1080/03052150600917128](https://doi.org/10.1080/03052150600917128).
- [23] K. Ahmed and A. ElAttia, "Integrating collaborative and outsourcing strategies for yard trucks assignment in ports with multiple container terminals," *Int. J. Logistics Syst. Manage.*, vol. 32, nos. 3–4, pp. 372–391, 2019.
- [24] J. He, D. Chang, W. Mi, and W. Yan, "A strategy for yard crane scheduling based on hybrid parallel genetic algorithm," in *Proc. Int. Symp. Knowl. Acquisition Modeling*, Wuhan, China, Dec. 2008, pp. 678–683, doi: [10.1109/KAM.2008.81](https://doi.org/10.1109/KAM.2008.81).
- [25] H. Javanshir and S. R. S. Ganji, "Yard crane scheduling in port container terminals using genetic algorithm," *J. Ind. Eng. Int.*, vol. 6, no. 11, pp. 39–50, 2010.
- [26] D. Chang, Z. Jiang, W. Yan, and J. He, "Developing a dynamic rolling-horizon decision strategy for yard crane scheduling," *Adv. Eng. Informat.*, vol. 25, no. 3, pp. 485–494, Aug. 2011.
- [27] J. He, Y. Huang, and W. Yan, "Yard crane scheduling in a container terminal for the trade-off between efficiency and energy consumption," *Adv. Eng. Informat.*, vol. 29, no. 1, pp. 59–75, Jan. 2015.
- [28] O. Sharif and N. Huynh, "Yard crane scheduling at container terminals: A comparative study of centralized and decentralized approaches," *Maritime Econ. Logistics*, vol. 14, no. 2, pp. 139–161, Jun. 2012, doi: [10.1057/mel.2012.1](https://doi.org/10.1057/mel.2012.1).
- [29] X. Guo, S. Y. Huang, W. J. Hsu, and M. Y. H. Low, "Dynamic yard crane dispatching in container terminals with predicted vehicle arrival information," *Adv. Eng. Informat.*, vol. 25, no. 3, pp. 472–484, Aug. 2011.
- [30] L. Chen, N. Bostel, P. Dejax, J. Cai, and L. Xi, "A Tabu search algorithm for the integrated scheduling problem of container handling systems in a maritime terminal," *Eur. J. Oper. Res.*, vol. 181, no. 1, pp. 40–58, Aug. 2007.
- [31] K. Zheng, Z. Lu, and X. Sun, "An effective heuristic for the integrated scheduling problem of automated container handling system using twin 40' cranes," in *Proc. 2nd Int. Conf. Comput. Modeling Simulation*, Sanya, China, Jan. 2010, pp. 406–410, doi: [10.1109/iccms.2010.290](https://doi.org/10.1109/iccms.2010.290).
- [32] J. X. Cao, D.-H. Lee, J. H. Chen, and Q. Shi, "The integrated yard truck and yard crane scheduling problem: Benders' decomposition-based methods," *Transp. Res. E, Logistics Transp. Res.*, vol. 46, no. 3, pp. 344–353, May 2010, doi: [10.1016/j.tr.2009.08.012](https://doi.org/10.1016/j.tr.2009.08.012).
- [33] L. Chen, A. Langevin, and Z. Lu, "Integrated scheduling of crane handling and truck transportation in a maritime container terminal," *Eur. J. Oper. Res.*, vol. 225, no. 1, pp. 142–152, Feb. 2013.
- [34] Y. Wu, J. Luo, D. Zhang, and M. Dong, "An integrated programming model for storage management and vehicle scheduling at container terminals," *Res. Transp. Econ.*, vol. 42, no. 1, pp. 13–27, Jun. 2013.
- [35] Z. Xue, C. Zhang, L. Miao, and W.-H. Lin, "An ant colony algorithm for yard truck scheduling and yard location assignment problems with precedence constraints," *J. Syst. Sci. Syst. Eng.*, vol. 22, no. 1, pp. 21–37, Mar. 2013.
- [36] J. He, Y. Huang, W. Yan, and S. Wang, "Integrated internal truck, yard crane and quay crane scheduling in a container terminal considering energy consumption," *Expert Syst. Appl.*, vol. 42, no. 5, pp. 2464–2487, 2015.
- [37] J. Luo, Y. Wu, and A. B. Mendes, "Modelling of integrated vehicle scheduling and container storage problems in unloading process at an automated container terminal," *Comput. Ind. Eng.*, vol. 94, pp. 32–44, Apr. 2016.
- [38] D. Kizilay, D. T. Eliyi, and P. Van Hentenryck, "Constraint and mathematical programming models for integrated port container terminal operations," in *Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, W.-J. van Hoes, Ed. Cham, Switzerland: Springer, 2018, pp. 344–360.
- [39] Y. Yang, M. Zhong, Y. Dessouky, and O. Postolache, "An integrated scheduling method for AGV routing in automated container terminals," *Comput. Ind. Eng.*, vol. 126, pp. 482–493, Dec. 2018.
- [40] T. Jonker, M. B. Duinkerken, N. Yorke-Smith, A. de Waal, and R. R. Negenborn, "Coordinated optimization of equipment operations in a container terminal," *Flexible Services Manuf. J.*, vol. 33, no. 2, pp. 281–311, Jun. 2021, doi: [10.1007/s10696-019-09366-3](https://doi.org/10.1007/s10696-019-09366-3).
- [41] A. Karam, A. Eltawil, and K. H. Reinaw, "Energy-efficient and integrated allocation of berths, quay cranes, and internal trucks in container terminals," *Sustainability*, vol. 12, no. 8, p. 3202, Apr. 2020, doi: [10.3390/su12083202](https://doi.org/10.3390/su12083202).
- [42] B. Vahdani, F. Mansour, M. Soltani, and D. Veysmoradi, "Bi-objective optimization for integrating quay crane and internal truck assignment with challenges of trucks sharing," *Knowl.-Based Syst.*, vol. 163, pp. 675–692, Jan. 2019, doi: [10.1016/j.knsys.2018.09.025](https://doi.org/10.1016/j.knsys.2018.09.025).
- [43] A. Karam and A. B. Eltawil, "A Lagrangian relaxation approach for the integrated quay crane and internal truck assignment in container terminals," *Int. J. Logistics Syst. Manage.*, vol. 24, no. 1, p. 113, 2016.
- [44] H.-P. Hsu, C.-N. Wang, H.-P. Fu, and T.-T. Dang, "Joint scheduling of yard crane, yard truck, and quay crane for container terminal considering vessel stowage plan: An integrated simulation-based optimization approach," *Mathematics*, vol. 9, no. 18, p. 2236, Sep. 2021, doi: [10.3390/math9182236](https://doi.org/10.3390/math9182236).

- [45] H.-P. Hsu, "A fuzzy knowledge-based disassembly process planning system based on fuzzy attributed and timed predicate/transition net," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 1800–1813, Aug. 2017.
- [46] H. P. Hsu, C. N. Wang, C. C. Chou, Y. Lee, and Y. F. Wen, "Modeling and solving the three seaside operational problems using an object-oriented and timed predicate/transition net," *Appl. Sci.*, vol. 7, no. 3, pp. 1–21, 2017.
- [47] H.-P. Hsu and C.-N. Wang, "A hybrid approach combining improved shuffled frog-leaping algorithm with dynamic programming for disassembly process planning," *IEEE Access*, vol. 9, pp. 57743–57756, 2021, doi: 10.1109/ACCESS.2021.3072831.



HSIEN-PIN HSU received the Ph.D. degree from the Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu, Taiwan, in 2004. He is currently a full-time Professor with the Department of Supply Chain Management, National Kaohsiung University of Science and Technology, Taiwan. He is familiar with manufacturing execution systems (MES), such as WORKSTREAM and PROMIS. He has worked as the MIS Manager in some semiconductor assembly and testing firms in Taiwan. He has published many articles in international journals, including *IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING*, *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS*, *IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING*, *IEEE ACCESS*, *IJPR*, *SAEC*, *ESWA*, and *IJAMT*. His research interests include MES, simulation, system modeling, petri net, predicate/transition net, soft computing, metaheuristics application, and operations research in the areas of factory and container terminal.



CHIEN-CHANG CHOU received the Ph.D. degree from the Department of Shipping Transportation Management, National Taiwan Ocean University, Keelung, Taiwan, in 2005. He is currently a full-time Professor with the Department of Shipping Technology, National Kaohsiung University of Science and Technology, Taiwan. He has published several academic articles in international journals, including *IEEE ACCESS*, *Applied Soft Computing*, *Applied Mathematics and Computation*, *Computers and Mathematics with Applications*, *Mathematical and Computer Modelling*, *Tourism Management*, *Marine Pollution Bulletin*, and *Ocean Engineering*. His research interests include shipping logistics, transportation management, maritime engineering, fuzzy mathematics, and decision-making science.



CHIA-NAN WANG received the Ph.D. degree from the Department of Industrial Engineering and Management, National Chiao Tung University, Taiwan, in 2004. He is currently a Professor with the Department of Industrial Engineering and Management, National Kaohsiung University of Science and Technology, Taiwan. He has published 80 journal articles and 40 patents. He achieves around 25 projects of research and academy and has some successful cases of technology transfer. His research interests include manufacturing automation and industrial applications. He achieves awards from many international invention competitions (two gold medal awards, five silver medal awards, and several bronze medal awards). In addition, he has plenty of practical experience in many industries and provides consultant services for several stock companies in Taiwan.

• • •