An Improved Multi-strategy Exponential Triangle Optimization Algorithm for Container Feeder Vessel Scheduling

Yuyan He*, Yumeng Zhang, Lei Li, Zihan Ren

School of Economics and Management, Guangxi University of Science and Technology, Liuzhou 545006, China

*yuyan_517@126.com

Abstract. The expansion of international trade has driven the development of container transportation, and the scheduling of vessels in feeder services within an axial hub-and-spoke network is crucial for the efficiency of shipping companies. This paper constructs a mixed-integer programming model with the maximum sailing time of vessels as a constraint and aims to minimize the total sailing cost. A multi-strategy Exponential Triangle Optimization Algorithm (ETOA-M) is designed to solve this model. Through fixed and random scenario experiments, ETOA-M is compared with Simulated Annealing (SA), Random Search (RS), and ETOA. The results indicate that ETOA-M outperforms the others in terms of convergence speed and cost optimization, particularly showing significant advantages in the 12-48 hours sailing time range, while also validating the necessity of the constraint. This research provides decision support for shipping companies to reduce operational costs and improve scheduling efficiency.

Keywords: Vessel transportation; scheduling optimization; sailing time of vessels; Exponential Triangle Optimization Algorithm; multiple strategies.

1 Introduction

The development of international trade and advancements in container technology have driven the rapid growth of containerization in transportation and the market, resulting in a hub-and-spoke network structure in maritime shipping. In this structure, large container ships are responsible for trunk transportation between hub ports, while medium and small vessels handle feeder services between hub ports and feeder ports. Among these, the precise scheduling of feeder vessels and control of sailing times are particularly critical, as they contribute to improving connection efficiency and the overall effectiveness of the network, while reducing delays and connection costs. For instance, in the aftermath of the 2021 incident involving the "MSC LAUREN" in the Suez Canal, inaccurate fuel budgeting and sailing time calculations often forced vessels to detour or refuel urgently, incurring high additional costs, including

T. Chamaratana et al. (eds.), *Proceedings of the 2026 12th International Conference on Digital Humanities and Frontiers in Social Sciences (DHFSS 2026)*, Advances in Social Science, Education and Humanities Research 1031,
https://doi.org/10.2991/978-2-38476-611-6_62

emergency refueling premiums, port charges, and schedule delays [1]. Therefore, for ports, scientifically designing feeder scheduling plans and accurately controlling sailing times are of significant practical importance for shortening cycles, reducing costs, and enhancing competitiveness.

Recent research on ship scheduling has made significant progress. For instance, AGARWAL et al. considered time windows and delivery demands, established a mathematical model, and developed a hybrid genetic algorithm for solving it[2]. CHUANG et al. took into account uncertain market demand and berthing time constraints, constructing a fuzzy demand path planning model and designing a multi-objective genetic algorithm for resolution[3]. Ding Yi et al. examined the random sailing times and soft time window constraints present in irregular shipping, formulating a stochastic programming model aimed at minimizing total costs, and employed a two-phase heuristic algorithm that combines the scanning method with tabu search for solution[4]. Yang Jin et al. investigated the ship scheduling problem under the collaborative model of locks and ship lifts, developing a dual-objective integer programming model based on sliding time windows, and utilized a non-dominated sorting genetic algorithm with an elite strategy for resolution[5]. Wen Jie et al. considered the impact of tidal variations on the draft depth of vessels, constructing an integrated model for berth allocation and ship scheduling based on rectangular arrangements, and solving it using a two-phase greedy algorithm and a two-dimensional genetic algorithm[6]. Du Zunfeng et al. studied the multi-objective optimization problem in hybrid fleet scheduling, formulating a multi-objective mixed-integer programming model that accounts for transportation costs, satisfaction, and carbon emissions, and employed a hybrid genetic algorithm that combines local search and simulated annealing for solution[7]. Hu Pinghua et al. considered complex constraints such as heterogeneous fleets and time windows, establishing a ship routing planning model aimed at maximizing total profit. They employed a hybrid metaheuristic algorithm combining large neighborhood search and variable neighborhood descent for solving the model[8]. Dong Shengping et al. investigated the special safety constraints of LNG vessel navigation, constructing a port scheduling model aimed at minimizing total waiting time, which was solved and validated using the CPLEX solver[9]. Zhen et al. explored the optimization problem of ship scheduling for entering and exiting ports in one-way channels, developing a scheduling model that considers navigational constraints and safety requirements, solved using the Q-learning reinforcement learning algorithm[10]. Guo et al. addressed the collaborative optimization problem of ship scheduling and tugboat allocation, creating an end-to-end deep reinforcement learning framework that utilized graph neural networks and the bi-proximal policy optimization (bi-PPO) algorithm for solution[11]. Cao et al. studied the ship scheduling and routing optimization problem considering risk control, constructing an optimization model that integrates CVaR and LSTM-MSNet predictions, and solved it using a hybrid heuristic algorithm and the Gurobi solver[12]. Wang et al. examined the multi-port pool ship scheduling problem under tidal influences, establishing a tidal phase coordination model solved using a multi-stage elite genetic algorithm[13]. Zhao et al. researched the ship scheduling problem under uncertain berth service times, constructing a two-stage robust optimization model solved using an improved column-and-constraint generation algorithm[14]. Zheng et al. considered the multi-resource collaborative scheduling problem of locks and channels, constructing a multi-objective mixed-integer programming model and solving it using a multi-objective heuristic algorithm based on Q-learning (QLMHA) [15]. Zhang et al. studied the integrated optimization problem of precise tidal synchronization and vessel scheduling in long-channel ports, developing a mixed-integer linear programming model and employing an improved branch-and-price algorithm for solution[16].

In summary, most of the current literature considers various real-world conditions such as time windows, port congestion, and ship arrival times. A small portion of the literature takes into account the stochastic sailing times of vessels, but these considerations are primarily focused on mainline vessel scheduling rather than feeder vessel scheduling. Additionally, the efficiency of the swarm intelligence algorithms employed needs improvement. Therefore, this paper establishes a mixed-integer programming model with the maximum sailing time of vessels as a hard constraint, aiming to minimize total sailing costs. An improved triangular exponential optimization algorithm is designed for solving this model, providing decision-making support for shipping enterprises to maximize benefits while ensuring operational safety and equipment durability.

2 Problem Description and Assumptions

2.1 Problem Description

The transportation of feeder container vessels is a core component of the hub-and-spoke maritime network. Its objective is to minimize total operating costs by optimizing vessel routes and operational schedules while ensuring service quality, time constraints, and navigational safety. The fundamental structure of the hub-and-spoke network views the hub port as the axis and the other connected feeder ports as spokes, as illustrated in Figure 1[8]. Feeder vessels transport containers from the hub port to various feeder ports while collecting containers from these feeder ports to return to the hub port.

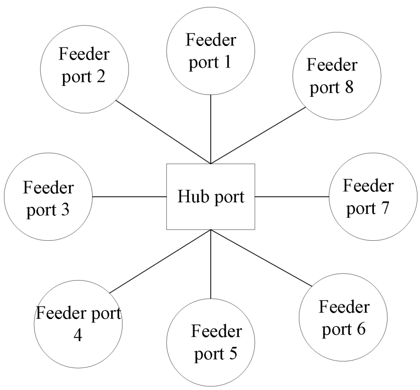

Fig. 1. Basic structure of hub and spoke network.

All vessels depart from the same hub port, planning their docking sequence and sailing routes based on the container demands of different feeder ports. The scheduling problem of container feeder vessels based on a hub-and-spoke network is illustrated in Figure 2[8]. The hub point H represents the hub port, while the spoke points S_k (k=1, 2, ..., K) denote the feeder ports. Containers destined for each feeder port S_k are uniformly unloaded at the hub port H, from where small to medium-sized container vessels transport all the containers along a designated route to the specified feeder ports. After collecting containers that require trunk transportation from one or several feeder ports, they return to the hub port. Clearly, throughout this process, the constraints of vessel loading capacity and sailing time must be satisfied.

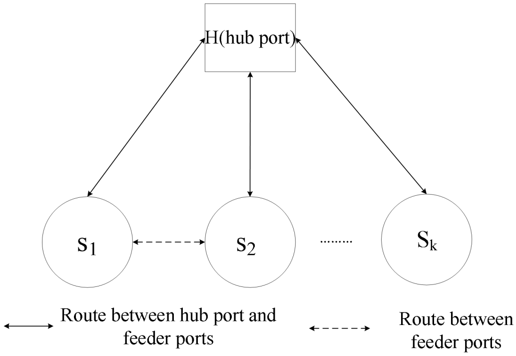

Fig. 2. Container feeder transportation vessel scheduling.

In the scheduling problem of feeder vessels, the determination of the maximum sailing time must comprehensively consider multiple dimensions of operational characteristics: (1) The transportation of empty containers needs to balance the demand for transporting full containers and recovering empty containers within a limited time, avoiding delays caused by detours or multiple port calls; (2) The isolation requirements for domestic and foreign trade goods necessitate separate voyages for vessels, making strict time constraints critical for ensuring the flight connections and compliance with navigation regulations; (3) Although the operational flexibility allows for dynamic adjustments in the sequence of tasks, the soft constraints on port stay times require the imposition of a maximum sailing time to enforce operational windows, preventing localized delays from impacting the overall scheduling; (4) Environmental and operational constraints (such as severe weather, channel congestion, and port prioritization) compel the model to allocate time redundancies to enhance robustness, while the port operation prioritization system further compresses the flexibility of vessel berthing; (5) The operational characteristics of feeder vessels, which are "small, short, and frequent" (small vessel type, short distance, frequent port calls), directly limit their endurance and time management, necessitating the design of modes such as "multi-vessel relay" or "segmented routes" with the maximum sailing time as a hard constraint. These factors collectively determine that the maximum sailing time is not only a boundary condition for route feasibility but also a core parameter that balances transport efficiency, cost control, and risk response. This principle is integrated into the path generation, speed optimization, and robustness design of the vessel scheduling model. Therefore, it is essential to ensure that the total sailing and operational time of each vessel does not exceed the preset threshold in order to achieve a balance between cost and navigational safety.

2.2 Assumption

Assumption 1: The departure and arrival ports of each vessel are the same hub port;

Assumption 2: When the same feeding port is called multiple times within the scheduling cycle, it is considered as different ports;

Assumption 3: The compatibility of vessels with ports, waterways, and container types is known (e.g., draft depth, hazardous material restrictions, etc.);

Assumption 4: The positions of vessels, container demands, and port operation times are known 12 hours in advance during the scheduling cycle;

Assumption 5: Each feeding port has the soft time windows (allowing early arrival with waiting or late arrival with penalties);

Assumption 6: Domestic and foreign trade containers are transported separately, and the mixed loading is prohibited;

Assumption 7: The port operation efficiency is constant and does not vary with vessel type;

Assumption 8: The operation time at the hub port is negligible;

Assumption 9: The maximum continuous sailing time of vessels (including sailing and operation time) must be strictly limited, and exceeding this time will trigger the high penalties;

Assumption 10: The empty container slots are dynamically updated, and the loading quantity must match the remaining container capacity of the vessel;

Assumption 11: The economic sailing speed and unit distance sailing cost of all vessels are known.

3 Model Establishment

3.1 Parameters and Decision Variables

$F=\{1,2,3,...,l\}$ the set of feeder ports; $i,j\in 0\cup F=P,$ P the set of hub port and feeder ports(The value 0 represents hub port); $k\in\{1,2,3,...,m\}_V$, V :the set of all the vessels; d_{ij} : The voyage distance between port i and port j; α_k : The unit voyage cost coefficient of vessel k (yuan/nmile), which is primarily determined by its rated container capacity; $\beta_{ki}=1$: Vessel k is compatible with port i ; $\beta_{ki}=M$, Vessel k is incompatible with port i , (The incompatibility between vessel k and port i is determined by a large certificate M, which serves as a penalty coefficient. This incompatibility includes factors such as the compatibility of the vessel with the port, the waterway, the type of containers, and whether foreign and domestic trade containers are mixed.); q_{ki} : The slack capacity of vessel *k* upon arrival at port *i* (if unloading occurs, plus the discharged container volume); g_i : The container demand volume at port *i* (TEU); ET_i : The earliest allowable service commencement time at port i ; LT_i : The latest allowable service commencement time at port i ; T_k : The departure time of vessel *k* from the hub port; A_{ki} : The arrival time of vessel *k* at port i ; N_{ki} : The container handling volume of vessel *k* at port i ; Eff_i : The container handling productivity at port *i* (TEU/h); V_k : The economic speed of vessel *k* (n mile/h); WT_{ki} : The berth waiting time of vessel *k* at port i ; DT_{ki} : The berthing delay time of vessel *k* at port i ; P_{Lk} : The per-unit-time demurrage cost for Vessel *k* arriving after the port's latest permitted operational commencement time. (yuan/h); P_{Ek} : The unit-time idling cost for vessel *k* arriving prior to the port's earliest permissible operational commencement time. (yuan/h); $T_k^{\max}$: The maximum continuous sailing duration for vessel *k*(h); t_k^s : The cumulative sailing time of vessel k since departure from the hub port(h).

Decision variables:

$x_{ijk}=1$: The voyage of vessel *k* from port *i* to port *j*, otherwise $x_{ijk}=0$; $y_{ki}=1$: Vessel k calling at port *i*, otherwise $y_{ki}=0$;

3.2 Objective Function and Constraints

$$\min Z=\sum_{i=0}^{l}\sum_{j=0}^{l}\sum_{k=1}^{m}\alpha_{\mathrm{k}}\beta_{kj}d_{ij}x_{ijk}+\sum_{k=1}^{m}\left[P_{Ek}\sum_{i=0}^{l}WT_{\mathrm{ki}}\cdot y_{ki}+P_{Lk}\sum_{i=0}^{l}DT_{ki}\cdot y_{ki}\right]+M\sum_{k=1}^{m}\max\left(t_k^s-T_k^{\max},0\right) \tag{1}$$

$$\sum_{i=o}^{l}\sum_{k=1}^{m}x_{ijk}\geq 1,\forall j\in F \tag{2}$$

$$\sum_{j\in F}x_{0jk}=1,\sum_{i\in F}x_{i0k}=1,\forall k\in V \tag{3}$$

$$A_{kj}=\left[A_{ki}+WT_{ki}+\frac{N_{ki}}{Eff_i}+\frac{d_{ij}}{v_k}\right]x_{ijk}, \quad A_{k0}=T_k,\forall i\in P,\forall j\in F,\forall k\in V \tag{4}$$

$$WT_{ki}=\max\{ET_i - A_{ki}, 0\}, \forall i \in F, \forall k \in V \tag{5}$$

$$DT_{ki} = \max\{A_{ki} - LT_i, 0\}, \forall i \in F, \forall k \in V \tag{6}$$

$$q_{kj} = q_{ki} + g_i x_{ijk}, \forall i \in P, \forall j \in F, i \neq j, \forall k \in V \tag{7}$$

$$q_{ki} - g_i y_{ki} \geq 0, \forall i \in P, \forall k \in V \tag{8}$$

$$t_k^s = \sum_{i,j \in P} \left(\frac{d_{ij}}{v_k} + \frac{N_{ki}}{Eff_i} \right) \cdot x_{ijk} \leq T_k^{\max}, \forall k \in V \tag{9}$$

Equation (1) represents the objective function, which aims to minimize the total sailing cost of all vessels while considering the berth utilization, subject to the soft time window constraints and the maximum sailing time of the vessels. Equation (2) indicates that each feeder port must be visited at least once; Equation (3) states that all vessels depart from the hub port and ultimately return to the hub port; Equations (4) to (6) describe the time constraints considering the soft time windows; Equation (7) specifies the allocation of empty container slots when vessel k arrives at port j; Equation (8) ensures that the number of empty container slots on the vessel at the loading port is greater than or equal to the amount of cargo to be loaded; Equation (9) restricts the total time that each vessel can spend in the continuous sailing and operations, ensuring that it does not exceed a predetermined safety threshold.

4 Algorithm Design and Improvement

The optimization of container feeder vessel scheduling has been proven to be an NP-Hard problem [17]. The complexity of the problem lies in the optimization of multiple constraints and objective functions. Traditional exact search algorithms based on mixed-integer programming cannot meet practical demands, while ETOA has demonstrated faster convergence rates and better global exploration capabilities, providing a superior solution for the optimization of container feeder vessel scheduling.

4.1 Improvement Strategy

The Exponential Trigonometric Optimization Algorithm (ETOA) is faced with issues of convergence accuracy degradation and susceptibility to local optima. To enhance the optimization performance of the algorithm, this paper proposes the multi-strategy Exponential Trigonometric Optimization Algorithm (ETOA-M), which incorporates the eight improvement strategies: multi-strategy chaos initialization, the Osprey optimization algorithm, self-adaptive mutation strategy, dynamic reverse learning strategy, descent mechanism, adaptive artificial bee colony escape mechanism, golden section time optimization mechanism, and post-optimization processing [18].

4.1.1 Multi-strategy chaos initialization. The traditional Logistic map exhibits uneven distribution within the interval [0,1]. After improvements, as shown in Equation (10), the chaotic sequence is then mapped to the solution space through a linear transformation, as illustrated in Equation (11).

$$x_{n+1}=\begin{cases}4x_n\left(1-x_n\right), if\, 0.03\le x_n\le 0.97\\ rand\left(0.01,0.99\right), otherwise\end{cases} \tag{10}$$

$$p=L_{lb}+\left(U_{ub}-L_{lb}\right)\times x_{n+1} \tag{11}$$

In this context, x_{n+1} represents the (n+1)th generation chaotic vector derived from x_n; p denotes the position of an individual in the population, which corresponds to a candidate solution; and L_{lb} and U_{ub} refer to the lower and upper bounds of the search space, respectively.

4.1.2 Osprey optimization algorithm (OOA). The fishing mechanism of the osprey simulates the hunting behavior of the osprey. Each osprey considers individuals with superior fitness to itself as target fish groups, selecting target locations according to the following rules (Equation (12)). After identifying a target, the osprey dives toward it, updating its position as per Equation (13). This is followed by boundary processing (Equation (14)), and finally, a greedy selection is performed to retain the better solution (Equation (15))[18].

$$f_{selected}=\begin{cases}X_{\text{best}}, r_1\le 0.5\\ X\left(f_{fish\_pos}\left(k\right)\right), r_1>0.5\end{cases} \tag{12}$$

$$X_{i,j}^{N1}=X_{i,j}+r_{i,j}\times\left(f_{selected}-I_{i,j}\cdot X_{i,j}\right) \tag{13}$$

$$X_{i,j}^{N1}=\begin{cases}X_{i,j}^{N1}, lb_j\le X_{i,j}^{N1}\le ub_j\\ lb_j, X_{i,j}^{N1}<lb_j\\ ub_j, X_{i,j}^{N1}>ub_j\end{cases} \tag{14}$$

$$X_i=\begin{cases}X_i^{N1}, F\left(X_i^{N1}\right)\le F\left(X_i\right)\\ X_i, F\left(X_i^{N1}\right)>F\left(X_i\right)\end{cases} \tag{15}$$

$f_{selected}$ represents the position of the target fish selected by the osprey, which determines the direction in which individual i moves; X_{best} denotes the position vector of the fittest individual in the current population; $f_{fish\_pos} = find\left(f_{fit} < fit(i)\right)$, f_{fish_pos} are the coordinate sets of the underwater fish positions; $find(\bullet)$ is a function used to find the coordinates of eagles that have a fitness value better than their own; f_{fit} is the collection of fitness values for all eagle individuals; $fit(i)$ is the fitness value of the i-th eagle; $X_{i,j}$ indicates the current position of the i-th osprey in the j-th dimension, while $X_{i,j}^{N1}$ indicates the new position of the i-th eagle in the j-th dimension; N1 signifies the new position obtained after the eagle completes the first stage of position update (targeting and fishing stage); $r_{i,j}$ is a random number within the range [0,1] that indicates the degree of offset; $I_{i,j}$ represents the direction of the predation behavior, which is a random number of either 1 or 2; X_i indicates the complete position vector of the i-th individual; and $L_{lb,j}$ and $U_{ub,j}$ represent the upper and lower bounds of the j-th dimension.

4.1.3 Descent mechanism. The falling mechanism applies disturbances with a probability of 30% to prevent the algorithm from getting trapped in local optima[18]:

$$X_{fall} = r_1 X - r_2 X + r_3 \Delta \tag{16}$$

Where $r_1 X$ represents the positive displacement from the current position (simulating initial velocity in free fall), $-r_2 X$ denotes the negative displacement from the current position (simulating air resistance), $r_3 \Delta$ signifies the random step disturbance (the simulating random airflow during descent).

Dynamic step-size update: $\Delta = (UB - LB) e^{-\frac{C_i t}{T}}$. The calculation of attenuation coefficient C_i: $\begin{cases} C_i = 2 \times W_i \times n \\ W_i = 0.1 - 0.05 \times \dfrac{t}{T} \end{cases}$, where *n* indicates the problem dimensionality.

4.1.4 Dynamic reverse learning mechanism. To enhance population diversity, we utilize the generation of reverse solutions through adversarial learning, as detailed below:

$$R_{rp,j} = k \times \left(a_j + b_j\right) - p_j \tag{17}$$

In the optimization process, let the minimum and maximum values of the j-th dimension be denoted as a_j and b_j, respectively. k is a random weight within the range [0, 1], while p_j represents the current solution of the j-th dimension.

4.1.5 Self-adaptive mutation. Introducing an adaptive mutation rate[18], enhancing exploration in the early stages and focusing on development in the later stages:

$$\mu(t) = \max\left(0.01, 1 - \left(\frac{m}{M}\right)(1-\theta)\right) \tag{18}$$

Where m represents the current iteration number, M is the total number of iterations, and θ is the lower threshold for the mutation rate; the mutation rate linearly decreases from 1 to 0.01 as iterations progress. The algorithm can be terminated early when $\mu(t) < \theta$.

4.1.6 Adaptive Artificial Bee Colony escape. Due to the periodic oscillation problem of the ETOA caused by the trigonometric functions, which can lead to stagnation in the search process, an escape mechanism is designed for stagnated individuals: when an individual stagnates for several consecutive generations, an escape operation is performed with a larger step size to help the individual jump out of local optima. The escape step size decays exponentially with the number of iterations, resulting in smaller escape disturbances in the later stages. The formulas for updating the escape step size and escape position are as follows:

$$step\_size = 0.5 \times \left(UB - LB\right) \times e^{-C_i \times \frac{iter}{Maxiter}} \tag{19}$$

$$escape\_pos = position + \left(2 \times rand\left(1, \dim\right) - 1\right) \times step\_size \tag{20}$$

Where $C_i = 2 \times 0.1 \times \dim$, $\dim$ represents the dimension of the problem.

4.1.7 Golden section time optimization. Introducing the golden section time optimization strategy for improvement. The golden ratio is set to $\phi = \frac{\sqrt{5}-1}{2} \approx 0.618$. The iterative process is as follows: First, initialize the interval $[a,b] = [0, T\_\max]$. Next, calculate two interior points $x_1 = b - \phi(b-a)$ and $x_2 = a + \phi(b-a)$. Then, compute the function values $f_1 = \mathrm{cost\_function}(x_1)$ and $f_2 = \mathrm{cost\_function}(x_2)$. By comparing f_1 and f_2; if $f_1 < f_2$, then set

$b = x_2$, $x_2 = x_1$, and $x_1 = b - \varphi(b-a)$. Otherwise, set $a = x_1$, $x_1 = x_2$, and $x_2 = a + \varphi(b-a)$. Repeat the above steps until the interval length is less than the tolerance.

4.1.8 Post optimization processing. The 2-opt algorithm is a local search technique that eliminates path crossings by swapping two edges. It iterates the over all possible edge pairs in a path, attempting to reverse a sub path segment to improve the total distance. For a given path: $R = [r_1, r_2, \cdots, r_n]$, select two indices i and j ($1 \le i \le j \le n$)to generate a new path $R^{'} = [r_1, \cdots, r_{i-1}, r_j, r_{j-1}, \cdots, r_i, r_{j+1}, \cdots, r_n]$, the operation reverses the sub path $[r_i, r_{i+1}, \cdots, r_j]$. The total distance of the new path is then computed. If improved, R is replaced by $R^{'}$.

4.2 Algorithm Flowchart

The overall procedure of the ETOA-M algorithm is presented in Fig. 3, which integrates eight improvement strategies to enhance convergence speed and solution quality.

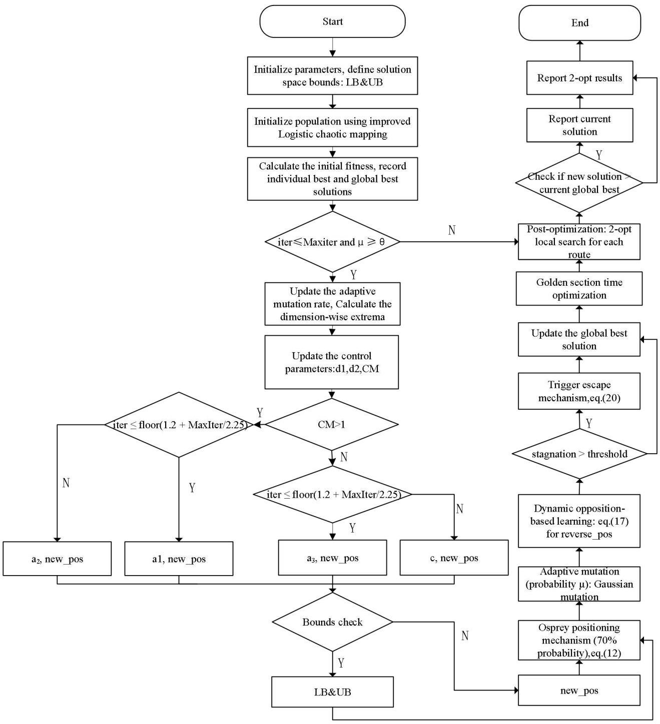

Fig. 3. Flowchart of the ETOA-M algorithm

5 Numerical Analysis

The test cases involve 15 feeder ports and 3 feeder container vessels, with each vessel having a sailing speed of 20 km/h. The corresponding sailing costs are set, including a unit distance cost of 100, a waiting cost of 10, and a delay cost of 20. The maximum sailing time is set between 12 and 112 hours.

In the ETOA, the population size is set to 200, and the number of iterations is set to 500. In the SA, the maximum number of iterations is set to 500, with an initial temperature of 1000, a termination temperature of 1e-5, and a cooling coefficient of 0.95. In the RS, the maximum number of iterations is also set to 500. The experimental environment for the algorithms is a 64-bit Windows 11 operating system, with an Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.11GHz processor and 16GB of RAM. The programming environment used for the experimental simulation is MATLAB R2024b.

5.1 Comparison of Results in Different Scenarios

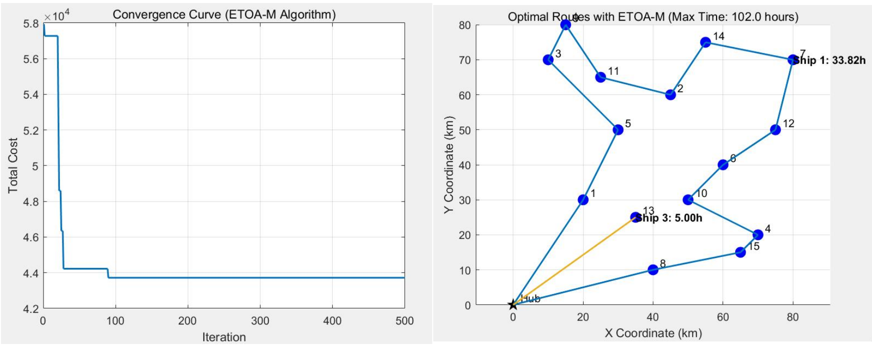

Fig. 4. Iteration and routs with a maximum sailing time of 102 hours.

Figure 4 illustrates the iterations and routs with a maximum sailing time of 102 hours, showing a total cost of 43218.30 HKD and a number of iterations equal to 50. Vessel 1 docks at 14 feeding ports, with the operating route as 0-1-5-3-9-11-2-14-7-12-6-10-4-15-8-0, sailing for 33.82 hours at a cost of 42307.87 HKD; Vessel 2 was not utilized; Vessel 3 only docked at feeding port 13, sailing for 5 hours at a cost of 860.33 HKD.

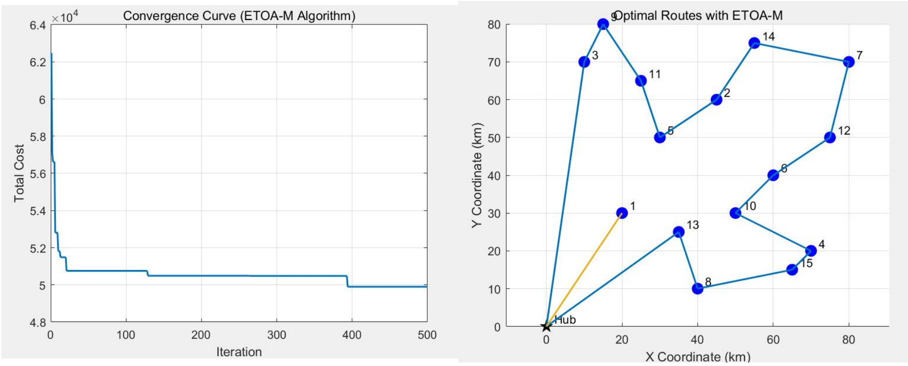

Fig. 5. Iteration and routs without considering the maximum sailing time.

Figure 5 illustrates the iterations and route map without considering the maximum sailing time constraint, showing a total cost of 43,754.39 HKD with 394 iterations. Vessel 1 calls at 14 feeder ports, with a route of 0-3-9-11-8-2-14-7-12-8-10-4-15-8-13-0, incurring a cost of 42,686.02 HKD and a sailing time of 34 hours; Vessel 2 was not utilized; Vessel 3 only called at Feeder Port 1, with a sailing time of 4.41 hours and a cost of 7,211.10 HKD.

The comparison between Figures 4 and 5 reveals that the costs without considering the maximum sailing time constraint outperform those with the constraint, showing an improvement of approximately 1.24%. This indicates that the maximum sailing time constraint mitigates the risk costs and enhances the optimization decisions. Additionally, the fewer iterations required when considering the maximum sailing time further demonstrates the effectiveness of the ETOA-M.

5.2 Comparison of Results with Different Algorithms

To comprehensively evaluate the performance of ETOA-M, it is compared with SA, RS, and ETOA under both fixed and random value scenarios. The specific results are presented in Tables 1 and 2.

Table 1. Comparison of results of different algorithms under the fixed scenarios.

Number	T_k^{max}	SA	RS	ETOA	ETOA-M	GAP1(%)	GPA2(%)	GPA3(%)
1	12	1,529,670,145.68	1,475,556,402.67	1,042,064,794.46	906,671,942.10	40.73	38.55	12.99
2	18	65,640.82	251,715,510.98	5,736,130.03	58,596.19	10.73	99.98	98.98
3	24	61,770.42	75,068.89	65,213.52	61,327.98	0.72	18.30	5.96
4	30	62,456.49	75,068.89	66,294.37	49,712.39	20.40	33.78	25.01
5	36	61,092.88	75,008.68	57,477.61	42,796.58	29.95	42.94	25.54
6	42	61,092.88	75,008.68	53,830.26	44,221.82	27.62	41.04	17.85
7	48	59,556.52	75,008.68	51,651.04	42,796.58	28.14	42.94	17.14

8	54	59,556.52	75,008.68	51,651.05	43,200.16	27.46	42.41	16.36
9	60	59,556.52	75,008.68	51,651.04	43,218.30	27.43	42.38	16.33
10	66	59,556.52	75,008.68	51,651.05	43,218.30	27.43	42.38	16.33
11	72	59,556.52	75,008.68	51,651.04	43,218.30	27.43	42.38	16.33
12	78	59,556.52	75,008.68	51,651.05	43,218.30	27.43	42.38	16.33
13	84	59,556.52	75,008.68	51,651.04	43,218.30	27.43	42.38	16.33
14	90	59,556.52	75,008.68	51,651.05	43,218.30	27.43	42.38	16.33
15	96	59,556.52	75,008.68	51,651.05	43,218.30	27.43	42.38	16.33
16	102	59,556.52	75,008.68	51,651.05	43,218.30	27.43	42.38	16.33
17	112	59,556.52	75,008.68	51,651.05	43,218.30	27.43	42.38	16.33

Note: GAP1=(the result of SA-the result of ETOA-M)/the result of SA; GAP2=(the result of RS-the result of ETOA-M)/the result of RS; GAP3=(the result of ETOA-the result of ETOA-M)/the result of ETOA

Table 1 shows the results of different algorithms under the fixed scenarios. The results indicate that ETOA-M achieves the optimal results, followed by ETOA, SA, and RS. Additionally, as the maximum sailing time increases, the costs obtained by the four algorithms—SA, RS, ETOA, and ETOA-M—exhibit an overall decreasing trend, stabilizing after the maximum sailing time reaches 60 hours. When the maximum sailing time of the vessel is 12 hours, the cost of ETOA-M is HKD 906,671,942.10, which is 40.73%, 38.55%, and 12.99% lower than SA, RS, and ETOA, respectively. As the maximum sailing time increases to 24 hours, the cost differences among the four algorithms begin to narrow, yet ETOA-M still achieves the lowest cost of HKD 61,327.98, which is 0.72%, 18.3%, and 5.96% lower than SA, RS, and ETOA, respectively. When the maximum sailing time ranges from 30 to 48 hours, the optimization effect of ETOA-M further expands, with the cost dropping to HKD 49,712.39 under the 30-hour constraint, which is 20.40%, 33.78%, and 25.01% lower than SA, RS, and ETOA, respectively. When the maximum sailing time exceeds 60 hours, the costs obtained by the four algorithms tend to stabilize, yet ETOA-M still maintains the lowest cost of HKD 43,218.30, which is 27.43%, 42.38%, and 16.33% lower than SA, RS, and ETOA, respectively.

Table 2. Results of different algorithms under the random scenarios.

Number	T_k^{max}	SA	RS	ETOA	ETOA-M	GAP1(%)	GPA2(%)	GPA3(%)
1	12	2,994,209,943.51	3,165,620,517.33	3,100,596,973.98	2,761,094,392.93	8.44	14.65	12.30
2	18	1,261,912,979.69	1,778,560,412.67	1,478,984,948.43	743,627,970.26	69.70	139.17	98.89
3	24	120,703,624.98	198,511,230.97	193,041,151.79	51,589,708.92	133.97	284.79	274.19
4	30	45,945,217.03	91,593.06	83,607.03	64,877.17	70718.78	41.18	28.87
5	36	72,915.09	87,182.02	74,321.74	51,393.20	41.88	69.64	44.61
6	42	73,619.55	80,031.44	72,611.54	64,256.27	14.57	24.55	13.00
7	48	71,201.85	77,757.31	74,812.77	60,611.47	17.4	28.29	23.43
8	54	71,277.78	79,597.37	71,451.15	54,321.28	31.22	46.53	31.53
9	60	65,992.66	79,065.73	73,982.76	51,290.80	28.66	54.15	44.24
10	66	70,431.34	81,194.78	71,993.77	50,929.26	38.29	59.43	41.36

11	72	73,906.51	82,034.84	70,864.71	69,061.33	7.02	18.79	2.61
12	78	75,646.69	83,917.80	70,934.94	69,734.75	8.48	20.34	1.72
13	84	66,505.00	76,627.11	70,359.88	56,373.15	17.9	35.93	24.81
14	90	71,066.68	78,366.52	71,644.82	50,009.60	42.11	56.70	43.26
15	96	70,766.34	81,076.63	70,536.69	52,840.76	33.92	53.44	33.49
16	102	69,447.25	79147.13	72,125.66	53,932.12	28.77	46.7	33.7
17	112	74,234.49	80,055.00	69,249.88	59,002.57	25.82	35.68	17.37

Note: The result is the average of 12 results in a random value scenario.

Table 2 presents the results of different algorithms under the random scenarios. The results indicate that the cost fluctuations of the four algorithms vary significantly, with the average cost obtained by ETOA-M being notably lower than that of the other three algorithms. When the maximum sailing time of the vessel is 12 hours, the average cost of ETOA-M is 2,761,094,392.93 HKD, which is lower than SA, RS, and ETOA by 8.44%, 14.65%, and 12.30%, respectively. When the maximum sailing time is 30 hours, the exceptionally high cost of SA results in an optimization gap of 70718.78%, while the average cost of ETOA-M is only 64,877.17 HKD, which is 41.18% and 28.87% less than RS and ETOA, respectively. For the maximum sailing times ranging from 36 to 112 hours, the optimization range of ETOA-M fluctuates to a certain extent, with the minimum optimization gap being 1.72% and the maximum being 69.64%, demonstrating the effectiveness of ETOA-M.

In summary, whether in the fixed or random scenarios, ETOA-M demonstrates significant superiority over SA, RS, and ETOA in terms of convergence speed, cost optimization, and robustness. This advantage is particularly pronounced when the sailing time of vessels increases, validating the effectiveness and practicality of ETOA-M in the optimization of scheduling for container feeder vessels.

6 Conclusion

This paper focuses on the optimization of scheduling for container feeder vessels. Addressing the shortcomings of existing research that overlook the constraint of maximum sailing time, this paper constructs a mixed-integer programming model aimed at minimizing total sailing costs and designed the ETOA-M for solving this problem. Through numerical experiments conducted under both fixed and random scenarios, ETOA-M is used to compared with SA, RS, and standard ETOA. The results indicate that ETOA-M outperforms SA, RS, and ETOA in terms of convergence speed and cost optimization, particularly when the maximum sailing time is within the range of 12 to 48 hours, where the advantages are significant. It is also noted that a larger maximum sailing time does not necessarily lead to better scheduling results or lower costs. This study validates the necessity of introducing a maximum sailing time constraint and demonstrates the effectiveness of ETOA-M in addressing such NP-Hard problems, providing a decision support for shipping companies to reduce the operational costs and enhance the scheduling efficiency.

In future research, a predictive model for dynamically adjusting the maximum sailing time of vessels can be constructed by incorporating the external factors such as meteorological data and channel traffic conditions, thereby enhancing the adaptability of scheduling schemes to complex marine environments. Additionally, scheduling scenarios involving multiple hub ports can also be considered. Although the proposed ETOA-M demonstrates superior efficiency and optimization results, there is still room for improvement in its performance when applied to larger-scale port networks (such as those including multiple hub ports and more vessels) or more complex constraints (such as mixed loading of multiple types of containers and dynamic cargo flow demands). Future explorations may focus on the integrating ETOA-M with other intelligent optimization algorithms to enhance its applicability in more complex scenarios.

Acknowledgments

This work was supported by Specific Research Project of Guangxi for Research Bases and Talents (Grant Number AD23026317), National Natural Science Foundation of China (NSFC) (Grant Number 72461004) and Guangxi University of Science and Technology Doctoral Fund Project (Grant Number 20S08).

Disclosure of Interests

The authors have no competing interests to declare that are relevant to the content of this article.

References

1. United Nations Conference on Trade and Development. (2021). Review of maritime transport 2021 (Report No. UNCTAD/RMT/2021). Geneva: UNCTAD, p. 15.https://unctad.org/system/files/official-document/rmt2021_en_0.pdf
2. Agarwal, R., & Ergun, Ö. (2008). Ship scheduling and network design for cargo routing in liner shipping. Transportation Science, *42*(2), 175–196. https://doi.org/10.1287/trsc.1070.0205
3. Chuang, T. N., Lin, C. T., & Kung, J. Y. (2010). Planning the route of container ships: A fuzzy genetic approach. Expert Systems with Applications, *37*(4), 2948–2956. https://doi.org/10.1016/j.eswa.2009.09.040
4. Ding, Y., Zhong, Y., Lin, G. L., et al. (2015). Tramp ship routing with soft time windows and stochastic sailing times. Journal of Central China Normal University (Natural Sciences), *49*(3), 387–391. https://doi.org/10.19603/j.cnki.1000-1190.2015.03.011
5. Yang, J., Li, Y. B., Yang, J. Y., et al. (2024). Vessel scheduling optimization under coordinated control of ship lock and ship lift. Journal of Shanghai Maritime University, *45*(1), 16–22. https://doi.org/10.13340/j.jsmu.202211140332
6. Wen, J., Liu, H. Y., Wang, Y. C., et al. (2024). Vessel scheduling problem based on cuboid arrangement under tidal influence. Navigation of China, *47*(3), 81–88.https://doi.org/10.3969/j.issn.1000-4653.2024.03.010

7. Du, Z. F., Fan, T., & Zhang, Q. W. (2024). Integrated scheduling optimization for feeder transportation based on different powered container ships. Navigation of China, *47*(1), 111–120+130. https://doi.org/CNKI:SUN:ZGHH.0.2024-01-014
8. Hu, P. H., Wu, L. B., Yuan, M., et al. (2025). Optimization of maritime vessel scheduling considering heterogeneous fleet and draft constraints. Journal of Transportation Engineering and Information, 1–16. https://doi.org/10.19961/j.cnki.1672-4747.2025.04.006
9. Dong, S. P., Yang, G. Y., Liao, S. G., et al. (2025). Port scheduling optimization model considering navigation characteristics of LNG carriers. Journal of Shanghai Maritime University, *46*(3), 60–66. https://doi.org/10.13340/j.jsmu.202404080061
10. Zhen, R., Sun, M., & Fang, Q. (2025). Optimization of inbound and outbound vessel scheduling in one-way channel based on reinforcement learning. Journal of Marine Science and Engineering, *13*(2), 237. https://doi.org/10.3390/JMSE13020237
11. Guo, W., Zhang, X., Li, Y., et al. (2025). Collaborative optimization of vessel scheduling and tugboat allocation in seaports using a multi-action deep reinforcement learning framework. Engineering Applications of Artificial Intelligence, *156*(Part A), 111239. https://doi.org/10.1016/J.ENGAPPAI.2025.111239
12. Cao, Z., Zhu, Z., Lu, W., et al. (2025). Risk-aware vessel scheduling and routing optimization with CVaR and LSTM-MS Net prediction. Journal of Marine Science and Engineering, *13*(2), 207. https://doi.org/10.3390/JMSE13020207
13. Wang, J., Liu, K., Yu, Y., et al. (2025). Vessel scheduling in multi-basin coastal ports affected by tidal currents. Ocean Engineering, *325*, 120822. https://doi.org/10.1016/J.OCEANENG.2025.120822
14. Zhao, R., Shu, S., Wu, L., et al. (2025). Robust vessel traffic scheduling with uncertain berth service times in a seaport. Transportation Research Part B: Methodological, *200*. https://doi.org/10.1016/j.trb.2025.103294.
15. Zheng, Q., Xuan, H., Zhang, Y., et al. (2025). Q-learning-based multi-objective heuristic algorithm for solving multi-resource co-scheduling problem of ship-lock-channel considering water discharge. IEEE Transactions on Intelligent Transportation Systems, 1–16. https://doi.org/10.1109/TITS.2025.3607755
16. Zhang, X., Guo, W., Yang, Z., et al. (2025). Optimization of integrated accurate ride-tide planning and vessel scheduling in multi-functional ports with long channels. Expert Systems with Applications, *273*, 126894. https://doi.org/10.1016/J.ESWA.2025.126894
17. Xu, T., Chen, C., Meng, F., et al. (2025). Exponential-trigonometric optimization algorithm with multi-strategy fusion for UAV three-dimensional path planning. The Journal of Supercomputing, *81*(7), 828. https://doi.org/10.1007/S11227-025-07335-2
18. He, Y., Kong, L. L., & Zheng, Z. M. (2025). Improved kingfisher algorithm with multi-strategy fusion for microgrid scheduling. Electronic Measurement Technology, *48*(7), 55–65. https://doi.org/10.19651/j.cnki.emt.2417499