scientific reports

Check for updates

OPEN

Research on tugboat scheduling optimization model considering the reliability of tugboat matching scheme

Yangjun Ren[1]✉, Mengchi Li[2], Yushun Lei[3], Yan Zhou[3], Di Liu[4], Jianjun Tu[5] & Botang Li[6]

The selection and scheduling of tugboat matching schemes are key tasks in tugboat assistance operation management. With large ships requiring more tugboat assistance, a two-stage multi-criteria decision-making method is proposed. This includes a normal distribution-based multi-attribute group decision-making method with triangular fuzzy numbers to determine tugboat matching scheme reliability. A tugboat scheduling planning model for multiple berthing bases is then established, targeting the minimization of total fuel cost and total matching scheme reliability. This bi-objective problem is solved using the posteriori method, with actual data from Nansha Port validating the proposed method. Meanwhile, a priority-based encoding Memetic algorithm is designed to address the characteristics of the problem, and the solution results for 25 test cases generated from the actual data range of the Guangzhou Port are compared and analyzed using CPLEX, genetic algorithms, and simulated annealing algorithms. The results verify the feasibility of the proposed priority-based encoding Memetic algorithm. The enhanced multi-attribute group decision-making method helps decision-makers quickly select suitable matching schemes and optimize tugboat scheduling, demonstrating effective reliability evaluation and planning optimization.

Keywords Tugboat scheduling, Tugboat matching, Multiple berthing facilities, Multi-attribute group decision-making, Reliability evaluation, Memetic algorithm

The docking process of ships is crucial for container terminal efficiency. Ships must queue in anchorages and wait for their turn based on port authority's berth allocation. Tugboats are essential for ship docking due to the ship's momentum, which may cause facility damage if using own power. With increasing vessel sizes and numbers, and the trend towards larger ships, port tugboat operations face greater pressure. Simply increasing tugboat numbers results in higher investment costs. Optimizing tugboat scheduling quality is vital for relieving terminal pressures and reducing harbor congestion[1]. A well-planned tugboat scheduling plan ensures services are executed timely and efficiently, reducing tugboat idleness and vessel waiting times, ensuring a smooth tugboat service system.

Port authorities are tasked with formulating tugboat scheduling plans that meticulously consider the chronological order and projected timelines of ship arrivals and departures within a given planning horizon. These plans must integrate the real-time positioning and operational speeds of the tugboats to facilitate seamless coordination with the assigned ships. Moreover, the precise positioning of harbor basins and berths during each vessel's loading and unloading processes must be taken into account to determine the theoretical duration of each task, followed by the appropriate allocation of the limited tugboat resources. For smaller vessels, a tugboat of reduced power may be adequate to meet their power requirements; however, larger vessels, characterized by their increased load capacity and deeper draft, necessitate the collaborative operation of two high-powered tugboats or a combination of one high-powered and two low-powered tugboats for effective berthing assistance[2]. This implies that each ship's inbound and outbound movements involve a multitude of tugboat number and power configuration options for assistance operations. The challenge is to select a suitable matching schedule

[1]School of Digital Commerce and Jiangsu Institute of Costume Culture, Changzhou Vocational Institute of Textile and Garment, Changzhou 213164, China. [2]School of Digital Economics and Trade, Guangzhou Maritime University, Guangzhou 510725, China. [3]School of Maritime Law and Traffic Management, Guangzhou Maritime University, Guangzhou 510725, China. [4]School of Sciences and Arts, Guangzhou Maritime University, Guangzhou 510725, China. [5]Department of Maritime, Guangdong Communication Polytechnic, Guangzhou 510800, China. [6]Department of Intelligent Transportation and Engineering, Guangzhou Maritime University, Guangzhou 510725, China. ✉email: yjren2019125@163.com

for each ship and to coordinate the use of tugboat resources with other ships for scheduling. This requirement should be stringently managed within the tugboat scheduling plan to ensure optimal operational efficiency.

In response to the scale of port construction and the increase in the number of inbound and outbound vessels, many ports have constructed multiple tugboat berthing bases. Nowadays, tugboat bases typically adopt a decentralized layout, with tugboats of different power classes choosing to berth at the nearest base after the end of the previous planning period, awaiting the assigned tasks for the upcoming period. This illustrates that the optimization of port tugboat scheduling involves numerous complex constraints. Moreover, as the scale of the problem grows, it becomes increasingly challenging to solve, posing a significant test for ports.

Given the complexities of tugboat scheduling in bustling ports, it is imperative to carefully weigh various factors to arrive at scientific and rational decision-making. This study investigates various tugboat matching schemes, multiple berthing bases, and multi-attribute group decision-making and scheduling issues involving different power classes of tugboats. Firstly, a multi-attribute group decision-making evaluation method based on the normal distribution and incorporating triangular fuzzy numbers is proposed to determine the reliability of tugboat matching schemes. Subsequently, considering the reliability of tugboat matching schemes and their selection, the choice of berthing bases, and the selection of the number and power of tugboats, a tugboat scheduling planning model for multiple berthing bases is established with the objective of minimizing the total fuel cost and the reliability of tugboat matching schemes. The main purpose of this research is to assist towing companies in making real-time daily decisions for tugboat assistance operations during the inbound and outbound movements of ships.

The novelty and significance of the main discovery is that the impact of multiple tugboat matching schemes and multiple berthing bases on tugboat scheduling plans. The distinct contributions of this paper are as follows: (1) It takes into account the variety of tugboat matching schemes, thereby enhancing the optimality of tugboat scheduling. (2) It gives greater consideration to multiple berthing bases, ensuring that the model is more reflective of real-world conditions. (3) It is the first to incorporate a reliability assessment of tugboat matching schemes within the context of tugboat scheduling.

The remainder of this paper is organized as follows. Section 2 provides a literature review. Section 3 introduces the background of the problem and describes the composition of the model. Section 4 describes the solution methodology. Section 5 presents a case study, and Sect. 6 summarizes the results.

Literature review

The literature reviews various multi-criteria multi-attribute decision-making methods for port assessment and tugboat scheduling. This paper summarize recent advancements and highlight the research gap relative to our study.

Scheme reliability assessment

Multi-criteria multi-attribute decision-making methods, as crucial tools for evaluating alternative schemes under multiple factors, have recently been applied to various port-related issues. For instance, García-Morales et al.[3] considered climate change and integration and development factors, using joint distribution functions obtained from predefined management alternatives to compare strategies and make decisions based on stochastic multi-indicator analysis. They designed four alternative schemes to address port congestion issues, determining the relative importance of each criterion in the decision-making process. Pesce et al.[4] employed multi-criteria decision analysis to balance various standards and indicators related to environmental, economic, and social sustainability, choosing sustainable cruise routes. They assessed four representative channel projects submitted to the Italian Venice local authorities and consulted experts from local authorities, private port enterprises, and the cruise industry to evaluate the effectiveness and weight distribution of the selected criteria. Through sensitivity analysis, they assessed the robustness of the proposed recommendations by evaluating the impact of weight changes on the ranking of alternative routes. Yang et al.[5] addressed the challenges of decision-making consensus due to the uncertainty in evaluation procedures and the complexity of indicators, proposing a new decision-making framework based on the Fermatean fuzzy comprehensive weighted distance measure and the TOPSIS method for green and low-carbon port assessment. They evaluated the green and low-carbon development status of five major Chinese ports. Díaz and Soares[6] discussed the optimal port selection for storing and assembling offshore wind components, presenting a comprehensive multi-criteria decision method based on ranking and weighted product methods. They identified 21 criteria based on existing data and expert analysis to rank ports and determine the best solution, applying the method to Atlantic ports on the Iberian Peninsula. Ashraf et al.[7] considered the psychological behavior of decision-makers in evaluating emergency judgments, proposing a method based on the complex probability hesitant fuzzy soft set. They demonstrated the method's effectiveness through a case study of major fire emergency decisions in the Tianjin port explosion.

Despite the mature application of multi-criteria multi-attribute decision-making methods in other fields, there has been limited research on their use for assessment of port support operations, and there is a lack of literature on the reliability evaluation of alternative tugboat matching schemes. Consequently, this paper will be the first to apply a multi-criteria multi-attribute decision-making approach to assess the reliability of different tugboat matching schemes.

Tugboat matching and scheduling

Tugboats are crucial for port operations, but their limited numbers require efficient scheduling to manage heavy workloads. Early research focused on optimization models for towing operations, especially for barges that need towing. For example, Zhen et al.[8] developed a mixed-integer model for tugboat scheduling in estuarial ports, while Wu et al.[9] addressed barge evacuation planning in reclamation projects. They formulated the problem as a nonlinear mixed-integer programming model and proposed a targeted heuristic method. Through extensive

numerical experiments and case studies, the effectiveness and efficiency of the proposed solution method were demonstrated. Zhu et al.[10] proposed a model to optimize tugboat and barge schedules for energy efficiency, and Hao et al.[1] studied joint barge and tugboat scheduling in tidal channel transit ports. These studies contribute to better management of port resources and reduction of operational time.

Tugboats are commonly used to assist ships in port operations, and ports schedule vessels and allocate berths after receiving estimated arrival times. After ports receive estimated arrival times for ships, they schedule the ships' order of arrival and berthing. Most ships require pilots and tugboats for port operations. In practice, tugboats and pilots are categorized into different levels, and ports have limited resources of both. To avoid excessive scheduling workload, ports should match ships with appropriate tugboats and pilots of the same level without altering the ships' schedules or berthing plans, to minimize delays.

Existing literature focuses on pilot assignment, tidal conditions, and canal influences on tugboat scheduling. In joint pilot and tugboat scheduling, Abou Kasm et al.[11] developed a mixed-integer programming model and an exact solution method for vessel scheduling with pilot and tugboat constraints, which showed significant improvements over first-come, first-served strategies in congestion. For tidal impacts on vessel scheduling, Zhong et al.[12] created a bi-objective, mixed-integer linear programming model for eco-friendly tugboat scheduling in tidal ports, aiming to reduce completion time and fuel consumption. This model was solved using an NSGA-II framework, offering guidance for port tugboat scheduling. Regarding canal features, Petris et al.[13] developed models and algorithms for vessel scheduling and tugboat allocation in canal ports. Two mathematical models and four heuristics were introduced and validated using Venice Port as a case study. For joint berth, tugboat, and shore crane scheduling in container ports, Yang et al.[2] created an economic cost-based optimization model, considering economic variability. A chaotic quantum butterfly optimizer algorithm was developed and tested on two Chinese ports, showing enhanced performance.

Research on tugboat scheduling for general port operations is limited, with Wei et al.[14] developing a mixed-integer linear programming model and branch-and-cut algorithm for tugboat scheduling using real ship traffic data from Singapore. Kang et al.[15] addressed uncertainty in large container ports with a mixed-integer linear programming model and a self-organizing algorithm for towage chain generation. Jia et al.[16] developed an integer programming model for busy ports, considering berthing plans and towing requirements, aiming to minimize delays and costs, and developed an iterative solution combining Lagrangian relaxation and Benders decomposition.

However, the above literature neglects a problem: Before assigning tugboats to assist ships in entering or leaving the port, it is necessary to select a reasonable tugboat matching scheme for each auxiliary operation of the ship under the real-time tugboat resource situation, to leave more appropriate tugboat resources for subsequent ships entering or leaving the port. For example, Abou Kasm et al.[17] considered channel restrictions and tugboat allocation policies, introducing a mixed-integer programming model to simulate ship scheduling under channel restrictions and different tugboat allocation strategies. Through case studies, they compared three tugboat matching policies and found that optimizing tugboat configurations is important, even when the ship sequence is predetermined. Currently, in addition to the tugboat matching policies and regulations issued by each region or port, ports can also obtain other tugboat matching schemes for certain ship operations based on the experience of port dispatchers.

From the perspective of tugboat scheduling modeling, existing literature often starts with classical models of the assignment problem[18–22], vehicle routing problem[23], and flowshop problem[12]. In addition, some studies have explored various optimization problems with uncertainty and algorithms[24–42]. When considering a combination of various classical models, modeling from the perspective of the assignment problem requires adding variables, increasing model complexity. Modeling from the perspective of the flowshop problem is challenging to handle cases where ships do not arrive at the port in their predetermined order. Only by modeling based on the classical vehicle routing problem can a better definition of the tugboat scheduling optimization model be achieved.

In the literature review provided, it is observed that the process of tugboat scheduling is typically organized in accordance with the anticipated arrival times of ships. This involves identifying the specific tugs and their required horsepower before assigning them to the relevant tasks. Although numerous scholars have studied tugboat scheduling, there is a scarcity of research that concurrently addresses each berthing operation's dependence on tugboat assistance, along with the evaluation of multiple tugboat matching schemes and the complexities of tugboat scheduling. Only a few studies have integrated constraints related to the quantity and horsepower of tugs in tugboat scheduling. To date, no research has simultaneously investigated both the reliability assessment of tugboat matching schemes offered by towage companies and the intricacies of tugboat scheduling. This manuscript introduces a novel two-stage multi-attribute group decision-making optimization approach for assessing the reliability of tugboat matching solutions and for tugboat scheduling, thereby closing the existing research gap. The proposed tugboat scheduling framework and the integrated method presented here have not been previously investigated in the academic literature.

Problem description and modeling

Problem description

This research focuses on a tugboat scheduling problem involving multiple berthing bases, tugboats, matching schemes, and task allocations, where tugboats can dock at any base and must select a matching scheme to complete tasks before returning to a base or moving on to the next task.

Tugboat fees are generally structured on a per-task basis, with the total cost determined once daily tasks are assigned. The primary measure of scheduling efficiency is thus cost-related, particularly fuel expenses. Timeliness of service is also crucial, as tasks must be finished within their designated start times to avoid penalties. This paper incorporates these delay costs into the objective: to minimize the overall cost, which includes both fuel and delay-related expenses.

Additionally, each alternative tugboat matching scheme must be evaluated for reliability based on various performance indicators. It must also be compared against potential solutions for that day's tasks to determine the most suitable match. Hence, the second objective of this paper is to maximize the overall reliability of the tasks for that day.

Under these conditions, this paper first employs a multi-attribute multi-objective decision-making method to evaluate the reliability of each candidate tugboat matching scheme for a task. The objective is to minimize the total cost, which is the sum of the tugboat's fuel cost and the cost of task delays, and to minimize the reliability of the tugboat matching schemes. The research focuses on the scheduling issues during the execution of tasks by tugboats.

Stage 1: modeling the reliability assessment of tugboat matching schemes

In diverse uncertain environments, the normal fuzzy number-based multi-attribute group decision-making method is relatively in line with human thinking and decision-making processes due to the normal membership function. However, people's evaluations of things are often ambiguous. This paper employs the normal fuzzy number-based multi-attribute group decision-making method[43] and combines it with triangular fuzzy numbers to convert the reliability of candidate tugboat matching schemes into quantitative parameters.

In this paper, this paper define the set of all candidate tugboat matching schemes as the decision alternative set $X = \{X_1, X_2, \cdots, X_m\}$; the evaluation indicators of the tugboat matching schemes are the attributes (evaluation indicators) set $U = \{U_1, U_2, \cdots, U_n\}$ and their weights $\rho = \{\rho_1, \rho_2, \cdots, \rho_n\}$; this paper set the relevant managers within the tugboat company as the decision-making group $D = \{D_1, D_2, \cdots, D_l\}$ and their weights $v = \{v_1, v_2, \cdots, v_l\}$. According to the method described, the calculation process for assessing the reliability of tugboat matching schemes is as follows (note: the mathematical symbols used in this section are independent of those used in Sect. 2):

Step 1 Given a decision matrix, the triangular fuzzy decision matrix regarding the attributes for the t-th decision maker is $A^t = \begin{bmatrix} \left(\tilde{a}_{11}^t, \left(\tilde{\sigma}_{11}^t\right)^2\right) & \left(\tilde{a}_{12}^t, \left(\tilde{\sigma}_{12}^t\right)^2\right) & \cdots & \left(\tilde{a}_{1n}^t, \left(\tilde{\sigma}_{1n}^t\right)^2\right) \\ \left(\tilde{a}_{21}^t, \left(\tilde{\sigma}_{21}^t\right)^2\right) & \left(\tilde{a}_{22}^t, \left(\tilde{\sigma}_{22}^t\right)^2\right) & \cdots & \left(\tilde{a}_{2n}^t, \left(\tilde{\sigma}_{2n}^t\right)^2\right) \\ \vdots & \vdots & \ddots & \vdots \\ \left(\tilde{a}_{m1}^t, \left(\tilde{\sigma}_{m1}^t\right)^2\right) & \left(\tilde{a}_{m2}^t, \left(\tilde{\sigma}_{m2}^t\right)^2\right) & \cdots & \left(\tilde{a}_{mn}^t, \left(\tilde{\sigma}_{mn}^t\right)^2\right) \end{bmatrix}$, where $\tilde{a}_{mn}^t = (a_{mn}^{1t}, a_{mn}^{2t}, a_{mn}^{3t})$and satisfies $a_{mn}^{1t} \le a_{mn}^{2t} \le a_{mn}^{3t}$; $\tilde{\sigma}^t{}_{mn} = (\sigma^{1t}_{mn}, \sigma^{2t}_{mn}, \sigma^{3t}_{mn})$ and satisfies $\sigma^{1t}_{mn} \le \sigma^{2t}_{mn} \le \sigma^{3t}_{mn}$. According to the principle of α − cut ($(0 \le \alpha \le 1)$), $\tilde{a}_{mn}^t = \left[a_{mn}^{Lt}, a_{mn}^{Rt}\right]$, where $a_{mn}^{Lt} = \left(a_{mn}^{2t} - a_{mn}^{1t}\right)\alpha + a_{mn}^{1t}$, $a_{mn}^{Rt} = a_{mn}^{3t} - \left(a_{mn}^{3t} - a_{mn}^{2t}\right)\alpha$; $\tilde{\sigma}^t{}_{mn} = \left[\sigma^{Lt}_{mn}, \sigma^{Rt}_{mn}\right]$, where $\sigma^{Lt}_{mn} = \left(\sigma^{2t}_{mn} - \sigma^{1t}_{mn}\right)\alpha + \sigma^{1t}_{mn}$, $\sigma^{Rt}_{mn} = \sigma^{3t}_{mn} - \left(\sigma^{3t}_{mn} - \sigma^{2t}_{mn}\right)\alpha$. By applying the "3a principle" of the normal distribution, that is, $\left(a_{mn}^{Lt} - 3\sigma^{Rt}_{mn}, a_{mn}^{Rt} + 3\sigma^{Rt}_{mn}\right)$, the normal fuzzy numbers are transformed into interval numbers. The corresponding interval decision matrix $B^t = \begin{bmatrix} \left[b_{11}^{Lt}, b_{11}^{Rt}\right] & \left[b_{12}^{Lt}, b_{12}^{Rt}\right] & \cdots & \left[b_{1n}^{Lt}, b_{1n}^{Rt}\right] \\ \left[b_{21}^{Lt}, b_{21}^{Rt}\right] & \left[b_{22}^{Lt}, b_{22}^{Rt}\right] & \cdots & \left[b_{2n}^{Lt}, b_{2n}^{Rt}\right] \\ \vdots & \vdots & \ddots & \vdots \\ \left[b_{m1}^{Lt}, b_{m1}^{Rt}\right] & \left[b_{m2}^{Lt}, b_{m2}^{Rt}\right] & \cdots & \left[b_{mn}^{Lt}, b_{mn}^{Rt}\right] \end{bmatrix}$ is obtained.

Step 2 Calculate the weights of each manager within the decision-making group. Due to the influence of the managers' job responsibilities and professional knowledge, the weights of each manager are uncertain and are expressed in the form of normal fuzzy numbers, denoted as $v = \{v_1, v_2, \cdots, v_l\}$, where $v_t = \left(a_v^t, \left(\sigma_v^t\right)^2\right), t = 1,2, \cdots, l$. Then, calculate the credibility values of each manager, where ϵ represents the attitude towards the evaluation values when assessing the manager's weight. If the expected value of the expert is considered important, then $0.5 \le \epsilon \le 1$. Using the credibility value function, calculate the weight of each expert according to the formula $\tilde{\omega}_t = \phi_t / \sum_{t=1}^{l} \phi_t$.

Step 3 Calculate the group decision matrix $B^t = \begin{bmatrix} \left[p_{11}^L, p_{11}^R\right] & \left[p_{12}^L, p_{12}^R\right] & \cdots & \left[p_{1n}^L, p_{1n}^R\right] \\ \left[p_{21}^L, p_{21}^R\right] & \left[p_{22}^L, p_{22}^R\right] & \cdots & \left[p_{2n}^L, p_{2n}^R\right] \\ \vdots & \vdots & \ddots & \vdots \\ \left[p_{m1}^L, p_{m1}^R\right] & \left[p_{m2}^L, p_{m2}^R\right] & \cdots & \left[p_{mn}^L, p_{mn}^R\right] \end{bmatrix}$, where $p_{ij}^L = \sum_{t=1}^{l} \widetilde{\omega}_t b_{ij}^{Lt}$ and $p_{ij}^R = \sum_{t=1}^{l} \widetilde{\omega}_t b_{ij}^{Rt}$.

Step 4 Calculate the weight values of decision attributes. First, find the basic solutions under different attributes U_j. The attribute weights obtained should minimize the sum of the squared weighted deviations under all attributes. Let $\zeta_j^L = \max_{1 \le i \le m} p_{ij}^L$ and $\zeta_j^R = \max_{1 \le i \le m} p_{ij}^R$, then $\left[\zeta_j^L, \zeta_j^R\right]$ is called the basic solution under attribute U_j. Solving the following quadratic programming model can obtain the weight value $\rho_j, j = 1,2,\cdots,n$ of the j-th attribute U_j: $\min \sum_{j=1}^{n} \rho_j^2 \left(\zeta_j^R - \zeta_j^L\right)^2 \; s.t. \; \sum_{j=1}^{n} \rho_j = 1; \rho_j > 0, j = 1,2,\cdots,n.$

Step 5 Calculate the comprehensive decision matrix. Based on the group decision matrix and the decision attribute values, calculate the comprehensive decision matrix Z. Then $Z = (Z_1, Z_2, \cdots, Z_m)^T = \left(\left[z_1^L, z_1^R\right], \left[z_2^L, z_2^R\right], \cdots, \left[z_m^L, z_m^R\right]\right)^T$, where $Z_1 = \left[z_1^L, z_1^R\right] = \left[\sum_{j=1}^{n} \rho_j p_{ij}^L, \sum_{j=1}^{n} \rho_j p_{ij}^R\right]$.

Step 6 Calculate the comprehensive evaluation value of the reliability of each tugboat matching scheme. Define the basic interval numbers $V = \left[v^L, v^R\right] = \left[\min z_i^L, \max z_i^R\right]$. First, calculate the left and right distances $d_i^L = \left|z_i^L - v^L\right|$, $d_i^R = \left|z_i^R - v^R\right|$ between each tugboat matching scheme and the basic interval numbers. Then, compute the comprehensive evaluation value of the reliability of each tugboat matching scheme: $\Psi(X_i) = \iota\, d_i^L / \left[\iota\, d_i^L + (1-\iota)\, d_i^R\right]$, where ι represents a attitude of a manager. Finally, normalize the comprehensive evaluation values of the reliability of the tugboat matching schemes.

Stage 2: tugboat scheduling optimization model

Assumptions

The paper proposes the following assumptions: (1) The scheduling planning period is for one day; (2) The tugboats are moored at the base where all tasks ended the previous day before the start of all tasks on that day; (3) The distinction between external berths and inner berths is not considered; (4) The time it takes for a tugboat to prepare (debarking and embarking) is short, and the time spent on berthing and unberthing at the base is negligible; (5) Different tugboats have their own fixed speeds (optimal economic cruising speeds); (6) There is no limitation on the number of tugboats that can berth at each base.

Symbol explanation

(1) Set:

I is the set of tasks, $i, j \in I = \{1,2,\ldots,NI\}$;

J is the union set of task 0 and other tasks, $i, j \in J = \{0,1,2,\ldots,NI\}$, where 0 represents the tasks before the 0th moment when the day's tasks begin;

K is the set of tugboats, $k \in K = \{1,2,\ldots,NK\}$;

L is the set of bases, $l \in L = \{1,2,\ldots,NL\}$;

M is the set of tugboat matching schemes, $m \in M = \{1,2,\ldots,NM\}$.

(2) Parameters

f_{im} is the service time for candidate tugboat matching scheme m in task i, $i \in J$, min;

t_{li}^1 is the distance from the base to the location where the task begins, $i \in J$, km;

t_{li}^2 is the distance from the base to the location where the task ends, $i \in J$, km;

Q_{kl} is the location of tugboat k at the base l at the end of the previous day's tasks;

s_i is the earliest possible start time for task i, $i \in J$, min;

e_i is the latest possible start time for a task i, $i \in J$, min;

β is a sufficiently large value;

d_{im} is the number of tugboats required for candidate tugboat matching scheme m in task i, $i \in I$;

v is the speed of the tugboat, km/h;

H_k is the horsepower of tugboat k, HP;

G_{im} is the horsepower required for candidate tugboat matching scheme m in task i, HP;

ρ is the cost of delay per unit time, yuan/min;

ϕ is the unit cost of fuel consumption, yuan/km;

R_{im} is the reliability of candidate tugboat matching scheme m in task i, $i \in J$.

(3) Decision variables.

x_{klij} is a 0–1 variable where if tugboat k has completed task i and returned to base l to proceed with task j, it is set to 1; otherwise, it is set to 0, $i, j \in J$;

y_{im} is a 0–1 variable where if task i chooses tugboat matching scheme m, it is set to 1; otherwise, it is set to 0, $i \in J, m \in M$;

z_i is the start time of the task i, $i \in J$.

Programming model

(1) Objective Function 1.

Based on the problem description and symbol explanation, the objective function of the fuzzy planning model M1 established in this article includes the fuel cost of returning to the base from the location where the task is completed, the fuel cost of traveling from the base to the location where the task operation begins, and the cost of task delay, as illustrated below:

$$\min w_1^* = \phi \sum_{k\in K} \sum_{l\in L} \sum_{i\in I} \sum_{j\in I} t_{li}^2 x_{klij} + \phi \sum_{k\in K} \sum_{l\in L} \sum_{i\in I} \sum_{j\in I} t_{lj}^1 x_{klij} + \rho \sum_{i\in I} (z_i - s_i) \quad (1)$$

(2) Objective Function 2.

Maximize the total task reliability:

$$\max w_2^* = \sum_{i\in J} \sum_{m\in M} R_{im} y_{im} \quad (2)$$

Based on the assumptions of the model, the decision variables must satisfy the following constraints:

$$\sum_{m\in M} y_{im} = 1, \ \forall i \in I \quad (3)$$

$$y_{0m} = 0, \ \ \forall m \in M \quad (4)$$

$$\sum_{k\in K} \sum_{l\in L} \sum_{j\in J} x_{klij} \geq \sum_{m\in M} d_{im} y_{im}, \ \forall i \in I \quad (5)$$

$$\sum_{j\in J} x_{kl0j} \leq Q_{kl}, \ \forall k \in K, l \in L \quad (6)$$

$$\sum_{l\in L} \sum_{i\in J} x_{klih} - \sum_{l\in L} \sum_{j\in J} x_{klhj} = 0, \ \forall h \in I, k \in K \quad (7)$$

$$z_i + \sum_{m\in M} f_{im} y_{im} + 60 \times x_{klij} t_{li}^2 / v + 60 \times x_{klij} t_{lj}^1 / v - \beta (1 - x_{klij}) \leq z_j, \ \forall i, j \in I, k \in K, l \in L \quad (8)$$

$$s_i \leq z_i \leq e_i, \ \forall i \in J \quad (9)$$

$$H_k \geq G_{im} y_{im} x_{klij}, \ \forall i, j \in J, k \in K, l \in L, m \in M \quad (10)$$

$$x_{klij}, y_{im} \in \{0,1\}, \ \forall i, j \in J, k \in K, l \in L, m \in M \quad (11)$$

$$z_i \geq 0, \ \forall i \in J \quad (12)$$

The Eq. (3) indicate that each task can only choose one scheme; Eq. (4) states that no schemes should be chosen for the tasks with a start time of 0; Eq. (5) ensures that the number of tugboats required for task selection is met; Eq. (6) defines the initial berthing location of the tugboats for the day; Eq. (7) implies that a tugboat leaves a task after completion; Eq. (8) requires that the total time from the start of one task to its completion and then returning to a base before proceeding to the next task's starting location must be later than the start time of the next task; Eq. (9) restricts the task start time within a specified range; Eq. (10) ensures that the power of the tugboats does not fall below the required power for the task; Eq. (11) specifies that the variables are 0–1 variables; and Eq. (12) states that the variables must be greater than or equal to zero.

Defuzzified model formulation

Since there is a product of two types of variables in Eq. (10) which causes nonlinearity, this paper linearize it as follows:

Let 0 = 1, then Eq. (10) becomes Eq. (13):

$$H_k \geq G_{im} O_{klijm}, \ \forall i, j \in J, k \in K, l \in L, m \in M \quad (13)$$

And add the following constraint:

$$O_{klijm} + 1 \geq y_{im} + x_{klij}, \ \forall i, j \in J, k \in K, l \in L, m \in M \quad (14)$$

$$y_{im} \geq O_{klijm}, \ \forall i, j \in J, k \in K, l \in L, m \in M \quad (15)$$

$$x_{klij} \geq O_{klijm}, \ \forall i, j \in J, k \in K, l \in L, m \in M \quad (16)$$

$$O_{klijm} \in \{0,1\}, \ \forall i, j \in J, k \in K, l \in L, m \in M \quad (17)$$

Transformation of the objective functions

There are three methods for solving multi-objective optimization problems[44,45] : the prior method, the posterior method, and the interactive method. The prior method involves decision-makers specifying their preferences (i.e., weights) for each objective before the problem-solving process, thereby transforming the multi-objective optimization problem into a single-objective one. The posterior method, on the other hand, involves optimizing all objective functions simultaneously. This method first generates an efficient solution set (Pareto set), and then, at the end of the search process, the decision-maker selects the most suitable solution from this set. The interactive method sees decision-makers expressing their preferences for each objective during the decision-making process, with iterative exchanges between the decision-makers and the computational process until an acceptable solution is identified. In this paper, a compromise method is chosen to solve the model. Given the range of each objective function (i.e., the maximum and minimum values), a single objective is obtained through weighting and combination, and the specific objective function can be represented as follows:

$$\text{Min } w^* = \delta \left(\frac{w_1^* - w_1^{*\,min}}{w_1^{*\,max} - w_1^{*\,min}} \right) + (1-\delta) \left(\frac{w_2^{*\,max} - w_2^*}{w_2^{*\,max} - w_2^{*\,min}} \right) \tag{18}$$

Where $0 \leq \delta \leq 1$.

Regarding the range of objectives, decision-makers generally estimate it based on experience, while this paper adopts the method from literature[46] (see Table 1, where $Z_k(X)$ is the optimal solution of the k-th objective function $X^{(k)}$ and its best solution) to determine it, in order to ensure the feasibility of the two objectives.

Using the matrix from Table 1 and Eqs. (19) and (20) to respectively determine the lower and upper bounds for minimization and maximization problems.

$$Z_k^l = Z_{kk} \; and \; Z_k^u = \max\left(Z_{1k}, Z_{2k}, \cdots, Z_{kk}\right) \tag{19}$$

$$Z_k^u = Z_{kk} \; and \; Z_k^l = \min\left(Z_{1k}, Z_{2k}, \cdots, Z_{kk}\right) \tag{20}$$

In summary, this paper establishes a mixed-integer linear programming model with the objective function represented by Eq. (18), and the constraints represented by Eq. (3) to (9), and equations (11) to (17).

Solution method

The model M3 constructed in the previous section is a mixed-integer linear programming model, which can be directly solved by mature commercial solvers such as CPLEX, LINGO, and GAMS. However, when the problem scale increases, the efficiency of existing optimization solvers is low. In such cases, heuristic algorithms such as genetic algorithms, nested partitioning algorithms, and simulated annealing algorithms can be employed to improve solution efficiency, aiming to obtain better solutions in a shorter amount of time.

Genetic Algorithm (GA) is a stochastic search method based on the principles of natural selection and genetic inheritance[47]. As an evolutionary approach, GA has been of interest and has been successfully applied to combinatorial optimization problems[48]. However, a pure GA often lacks the capability of effective search intensity. Moscato and Norman[49] first proposed the concept of Memetic Algorithm (MA) by integrating local search into GA to enhance the search intensity. MA, similar to GA, is a population-based heuristic search method. Moreover, MA has been proven to be successfully applied to a variety of optimization problems, particularly in providing approximate solutions to NP-hard optimization problems[50]. MA combines domain knowledge with population-based search methods (such as GA), featuring the characteristics of efficient heuristic algorithms. In recent years, MA has been used in various optimization problems, such as production distribution problems[51], scheduling problems[52], minimum spanning frequency assignment problems[53], and partitioning problems[54]. Therefore, Memetic Algorithm is an optimization algorithm based on the evolution of a population of memes (also known as cultural memes, i.e., a feasible solution to the problem). This paper employs Memetic Algorithm to solve the model and uses CPLEX for comparative analysis.

For the representation of solutions in the Memetic Algorithm (MA), Choudhary et al.[55] utilized a generative tree coding method based on node depth levels; however, this method might generate infeasible solutions, necessitating a repair mechanism for the encoding. Another body of literature has employed a priority-based encoding method that does not require a repair mechanism. This encoding method simplifies the algorithm's coding and carries more optimized genetic information[56–59]. Therefore, by drawing on the encoding approach of logistics network models, this paper uses the priority-based encoding method to enable the algorithm to conduct a more comprehensive global and local search, enhancing the efficiency of the algorithm.

At the same time, an important issue in the MA is to make the neighborhood search more effective. Fung et al.[60] used an MA algorithm with three types of neighborhood searches—2-opt, 1–1 exchange moves, and 1 – 0

	$Z_1(X)$	$Z_2(X)$	...	$Z_k(X)$
$X^{(1)}$	Z_{11}	Z_{12}	...	Z_{1k}
$X^{(2)}$	Z_{21}	Z_{22}	...	Z_{2k}
...	...	...	...	...
$X^{(k)}$	Z_{k1}	Z_{k2}	...	Z_{kk}

Table 1. Game table.

	Scheme	(3,1,2)		(2,1,3)		(1,3,2)		(2,3,1)	
Tugboat	Base	Task 2	Base	Task 1	Base	Task 4	Base	Task 3	Base
1	(1,0,0)	2	(2,3,1)	4	(3,1,2)	2	(3,1,2)	3	(2,1,3)
2	(0,1,0)	5	(2,1,3)	2	(3,2,1)	5	(1,3,2)	1	(1,3,2)
3	(0,1,0)	4	(3,1,2)	5	(2,3,1)	1	(2,1,3)	5	(3,2,1)
4	(0,0,1)	1	(3,2,1)	3	(1,3,2)	3	(2,3,1)	4	(3,1,2)
5	(1,0,0)	3	(1,3,2)	1	(2,1,3)	4	(3,2,1)	2	(2,3,1)

Table 2. A feasible tugboat scheduling with priority encoding.

	Scheme	(1,0,0)		(0,0,1)		(0,1,0)		(0,1,0)	
Tugboat	Base	Task 2	Base	Task 1	Base	Task 4	Base	Task 3	Base
1	(1,0,0)	0	(0,0,0)	1	(1,0,0)	0	(0,0,0)	0	(0,0,0)
2	(0,1,0)	1	(0,0,1)	0	(0,0,0)	1	(0,1,0)	0	(0,0,0)
3	(0,1,0)	1	(1,0,0)	1	(0,1,0)	0	(0,0,0)	1	(1,0,0)
4	(0,0,1)	0	(0,0,0)	1	(0,1,0)	0	(0,0,0)	0	(0,0,0)
5	(1,0,0)	0	(0,0,0)	0	(0,0,0)	1	(1,0,0)	0	(0,0,0)

Table 3. Priority decoding of tugboat scheduling.

exchange moves—to solve the open arc path problem. Karaoglan and Altiparmak[61] proposed an MA algorithm that combined simulated annealing (SA) and integer programming formulation (IPF) for neighborhood search to solve a distribution network problem with location and routing selection. Hasani and Khosrojerdi[62] employed a custom hybrid adaptive large neighborhood search algorithm based on a tabu list for neighborhood search and proposed a Taguchi-efficient parallel MA algorithm for solving global supply chain network design problems. However, they only used one type of neighborhood search or simply stacked two or more types, which did not result in high solution efficiency. Therefore, this paper adopts a dynamic neighborhood search method that organically combines 2-opt and 3-opt, thereby improving the local search efficiency of the algorithm.

Individual encoding and decoding operations

Individual encoding operation

This paper uses a priority-based encoding method to represent the tugboat scheduling plan for the port, with a specific example provided in Table 2. Suppose there are 5 available tugboats, 3 bases, and 4 tasks, each with 3 alternative schemes. First, the tasks are sorted by their earliest start times, as shown in Table 2. In the current example, the arrangement is 2, 1, 4, 3. The second column indicates the initial docking base of the tugboats. The first row lists the alternative schemes for each task, randomly sorted with integers 1 to 3 to represent the priority level of each scheme. The third, fifth, seventh, and ninth columns indicate the priority of selecting tugboats for each task, randomly sorted with integers 1 to 5 to represent the priority level for each task's tugboat selection, with each priority level being mutually independent. The fourth, sixth, eighth, and tenth columns indicate the priority of each tugboat docking at one of the three candidate bases after serving each task, randomly sorted with integers 1 to 3. Thus, a feasible priority-based encoded individual for tugboat scheduling is established.

Individual decoding operation

For the priority encoding in Table 1, under the conditions of meeting the power required for the task and being able to arrive at the required starting point of the task on time, the tugboat with the highest priority value in the encoding is selected to determine which base it will depart from, which task it will complete, and which base it will return to. For tugboats not assigned to tasks, their base location remains unchanged. A specific example is shown in Table 3, where tasks 2, 1, 4, and 3 select schemes 1, 3, 2, and 2, respectively, and the number of tugboats required for each task is 2, 3, 2, and 1, respectively. After determining the schemes, and based on power and time constraints, tugboat 1 departs from Base 1, completes Task 1, and docks back at Base 1; tugboat 2 departs from Base 2, completes Task 2, and docks at Base 3, then completes Task 4 and docks at Base 2; tugboat 3 departs from Base 2, completes Task 2 and docks at Base 1, completes Task 1 and docks at Base 2, and after completing Task 3, docks at Base 1; tugboat 4 departs from Base 3, completes Task 1, and docks at Base 2; tugboat 5 departs from Base 1, completes Task 4, and docks back at Base 1.

Fitness and selection operation

In the population of the MA, each meme needs to be evaluated by calculating its fitness. The fitness function represents the advantages and disadvantages of each meme. In this paper, the fitness is the reciprocal of the objective function value (calculated by Eq. (18)).

The selection operation employs the roulette wheel selection method. Assuming the population size is *pop_size*, due to the crossover operation in the Memetic Algorithm, the population size doubles to 2**pop_size*. Therefore, during the selection operation, the roulette wheel needs to be spun 2**pop_size* times. The two memes

	Scheme	(3,1,2)		(2,1,3)		(1,3,2)		(2,3,1)	
Tugboat	Base	Task 2	Base	Task 1	Base	Task 4	Base	Task 3	Base
1	(1,0,0)	2	(2,3,1)	4	(3,1,2)	2	(3,1,2)	3	(2,1,3)
2	(0,1,0)	5	(2,1,3)	2	(3,2,1)	5	(1,3,2)	1	(1,3,2)
3	(0,1,0)	4	(3,1,2)	5	(2,3,1)	1	(2,1,3)	5	(3,2,1)
4	(0,0,1)	1	(3,2,1)	3	(1,3,2)	3	(2,3,1)	4	(3,1,2)
5	(1,0,0)	3	(1,3,2)	1	(2,1,3)	4	(3,2,1)	2	(2,3,1)

Table 4. Individual 1 before crossover.

	Scheme	(3,2,1)		(3,1,2)		(1,2,3)		(2,3,1)	
Tugboat	Base	Task 2	Base	Task 1	Base	Task 4	Base	Task 3	Base
1	(1,0,0)	1	(3,1,2)	3	(1,2,3)	3	(2,1,3)	3	(2,3,1)
2	(0,1,0)	5	(3,2,1)	1	(3,2,1)	5	(1,3,2)	5	(1,2,3)
3	(0,1,0)	3	(2,3,1)	2	(2,3,1)	1	(3,2,1)	2	(3,1,2)
4	(0,0,1)	4	(1,3,2)	4	(1,2,3)	2	(3,1,2)	4	(3,2,1)
5	(1,0,0)	2	(2,1,3)	5	(3,2,1)	4	(2,3,1)	1	(1,3,2)

Table 5. Individual 2 before crossover.

	Scheme	(3,1,2)		(3,1,2)		(1,2,3)		(2,3,1)	
Tugboat	Base	Task 2	Base	Task 1	Base	Task 4	Base	Task 3	Base
1	(1,0,0)	2	(2,3,1)	3	(1,2,3)	3	(2,1,3)	3	(2,3,1)
2	(0,1,0)	5	(2,1,3)	1	(3,2,1)	5	(1,3,2)	5	(1,2,3)
3	(0,1,0)	4	(3,1,2)	2	(2,3,1)	1	(3,2,1)	2	(3,1,2)
4	(0,0,1)	1	(3,2,1)	4	(1,2,3)	2	(3,1,2)	4	(3,2,1)
5	(1,0,0)	3	(1,3,2)	5	(3,2,1)	4	(2,3,1)	1	(1,3,2)

Table 6. Individual 1* after crossover.

	Scheme	(3,2,1)		(2,1,3)		(1,3,2)		(2,3,1)	
Tugboat	Base	Task 2	Base	Task 1	Base	Task 4	Base	Task 3	Base
1	(1,0,0)	1	(3,1,2)	4	(3,1,2)	2	(3,1,2)	3	(2,1,3)
2	(0,1,0)	5	(3,2,1)	2	(3,2,1)	5	(1,3,2)	1	(1,3,2)
3	(0,1,0)	3	(2,3,1)	5	(2,3,1)	1	(2,1,3)	5	(3,2,1)
4	(0,0,1)	4	(1,3,2)	3	(1,3,2)	3	(2,3,1)	4	(3,1,2)
5	(1,0,0)	2	(2,1,3)	1	(2,1,3)	4	(3,2,1)	2	(2,3,1)

Table 7. Individual 2* after crossover.

selected after two spins of the wheel are then compared based on their fitness values, and the meme with the higher fitness value is chosen. This process is repeated for the next two spins to maintain the population size at *pop_size*.

Crossover operation

A single-point crossover method is used. For the following two different individuals (as shown in Tables 4 and 5), a random integer d is generated within the range[1,4], which determines a position (i.e., the d-th section). The data of the two individuals after this position are then crossed over. For example, if d = 2, the crossover results in individual 1* and individual 2* (as shown in Tables 6 and 7).

Dynamic neighborhood search

In the crossover phase, the two offspring memes with the best fitness values are selected for neighborhood search to further optimize them. Before conducting the neighborhood search, the meme is evaluated probabilistically (e.g., with a 70% chance) to determine whether to perform this operation. There are two neighborhood search methods: 2-opt and 3-opt. The neighborhood search process is targeted at the priority of task selection for tugboats. The process for 2-opt is as follows: first, a random integer r is generated within the range [1, total

	Scheme	(3,1,2)		(2,1,3)		(1,3,2)		(2,3,1)	
Tugboat	Base	Task 2	Base	Task 1	Base	Task 4	Base	Task 3	Base
1	(1,0,0)	2	(2,3,1)	4	(3,1,2)	2	(3,1,2)	3	(2,1,3)
2	(0,1,0)	5	(2,1,3)	**3**	(3,2,1)	5	(1,3,2)	1	(1,3,2)
3	(0,1,0)	4	(3,1,2)	5	(2,3,1)	1	(2,1,3)	5	(3,2,1)
4	(0,0,1)	1	(3,2,1)	**2**	(1,3,2)	3	(2,3,1)	4	(3,1,2)
5	(1,0,0)	3	(1,3,2)	1	(2,1,3)	4	(3,2,1)	2	(2,3,1)

Table 8. The 2-opt process.

	Scheme	(3,1,2)		(2,1,3)		(1,3,2)		(2,3,1)	
Tugboat	Base	Task 2	Base	Task 1	Base	Task 4	Base	Task 3	Base
1	(1,0,0)	2	(2,3,1)	4	(3,1,2)	2	(3,1,2)	3	(2,1,3)
2	(0,1,0)	5	(2,1,3)	**5**	(3,2,1)	5	(1,3,2)	1	(1,3,2)
3	(0,1,0)	4	(3,1,2)	**1**	(2,3,1)	1	(2,1,3)	5	(3,2,1)
4	(0,0,1)	1	(3,2,1)	3	(1,3,2)	3	(2,3,1)	4	(3,1,2)
5	(1,0,0)	3	(1,3,2)	**2**	(2,1,3)	4	(3,2,1)	2	(2,3,1)

Table 9. The 3-opt process.

number of tasks], and then two numbers are randomly selected in the column matrix of the r-th section and their positions are swapped. Similarly, for 3-opt, three numbers are randomly selected in the column matrix of the r-th section, and their positions are swapped, ensuring that the three numbers are not in their original positions. Taking a feasible individual from Table 2 as an example, a random integer 2 is generated. In Task 1, the 2-opt operation swaps the positions of 2 and 3 (as shown in Table 8), while the 3-opt operation swaps the positions of 2, 5, and 1 to 5, 1, and 2, respectively (as shown in Table 9).

The following is the design of a dynamic neighborhood search algorithm:

For i=1 to (9+k/8)
If i<5 use the 3-opt neighborhood search
Else use the 2-opt neighborhood search
If the fitness value of the obtained neighborhood meme is better than that of the current meme, then replace the current meme with the obtained neighborhood meme
Else replace the current meme with the obtained neighborhood meme with a probability of (0.06 * pw)
End

Algorithm 1: With a probability of 85%, perform the following steps:

Where k is the number of the current iteration, pw = (tun - k) / (tun * 3), and tun is the total number of iterations.

In summary, the structure of the MA can be described as follows::

Operating Area	Horsepower Type				
	1600HP	3000HP	4000HP	5000HP	Above 6000 HP
Huangpu	1	4	0	0	0
Xinsha	0	3	1	2	0
Xiaohu	0	0	3	1	0
Nansha	0	0	5	2	3
Total	1	7	9	5	3

Table 10. Number of tugboats by horsepower in different operating areas.

P_k	Co(kg/min)	γ(kg/min)
1600HP	6.33	2.45
3000HP	6.67	2.51
4000HP	7.50	2.67
5000HP	10.83	2.92
6000HP	11.25	3.67
6800HP	11.50	4.33
6900HP	11.67	4.33

Table 11. Fuel consumption per unit time for each type of tugboat.

No.	Name	Length(km)	Average Speed (km/h)	Average Navigation Time (min)
1	Dahao Channel	18.8	12.53	90
2	South Linding Channel	28.7	11.48	150
3	North Linding Channel	22	11.00	120
4	Chuanbi Channel	11.9	7.93	90
5	Dahu Waterway	8.2	9.87	50
6	Nizhou Channel	8.6	11.46	45
7	Lianhuashan West Channel	7.8	10.40	45
8	Lianhuashan East Channel	7.7	10.26	45
9	Xinsha Channel	7.1	10.65	40

Table 12. Overview of the outgoing channels of Guangzhou Port.

Step 1: Initialization
Randomly generate 2 * (pop_size) memes
Step 2: Calculate the fitness of each individual
Step 3: Selection operation
Step 4: Crossover operation
Step 5: Dynamic neighborhood search
Step 6: Termination
Determine if the current iteration number exceeds the maximum number of iterations. If it does, output the optimized value; otherwise, return to **Step 2.**

Algorithm 2: Memetic Algorithm

Numerical experiments

Input data generation

To evaluate the performance of the Memetic Algorithm (MA), it was compared with CPLEX 12.8, Genetic Algorithm (GA), and Simulated Annealing (SA). The encoding for GA and SA adopted the encoding method described in this paper, where GA's crossover used the method from Sect. 4.3, and the mutation operation used

Test case	Task	Tugboat	Matching scheme	Base
1	5	3	6	4
2	6	5	6	4
3	7	5	6	4
4	8	6	6	4
5	9	6	6	4
6	10	6	6	4
7	11	15	6	4
8	12	15	6	4
9	13	15	6	4
10	14	20	6	4
11	15	20	6	4
12	16	20	6	4
13	17	20	6	4
14	18	20	6	4
15	19	20	6	4
16	20	25	6	4
17	23	25	6	4
18	26	25	6	4
19	28	25	6	4
20	30	25	6	4
21	32	25	6	4
22	35	25	6	4
23	39	25	6	4
24	43	25	6	4
25	45	25	6	4

Table 13. Scale of the test cases.

the 2-opt method from Sect. 4.4. The operation for generating new solutions in SA adopted the 3-opt method from Sect. 4.4. The MA, GA, and SA codes were written in MATLAB R2015b. The software environment for running MATLAB R2015b and CPLEX included an Intel(R) Core(TM) i7 2.70 GHz processor, 4GB of memory, and the Windows 10 (64-bit) operating system on a laptop computer. The current number of tugboats at each berthing base is shown in Table 10. This paper set the unit time delay cost to 20 yuan/min (based on the actual port charges) and the unit cost of fuel consumption to 80 yuan/km (based on the average fuel consumption from Table 11 and the realistic fuel prices). Tables 10 and 11 are derived from the data in the literature[24], while Tables 12 and 13 are provided by internal resources of Guangzhou Port.

The port areas of Guangzhou are linked via the Pearl River system, with the key waterways detailed in Table 4. Tugboats in Guangzhou Port are typically restricted to a speed of 10 knots, whereas other regions allow for a maximum of 13 knots. According to the average speeds recorded for the 9 channels in Table 4, the tugboats operating in Guangzhou Port's outgoing channels achieve an average speed of 10.62 km/h, satisfying the specified limits. Moreover, Table 4 suggests that the distances between tasks and bases should fall between 7 km and 28 km, and the intervals between tasks are also within this specified range.

This paper randomly generates 25 instances of different sizes based on Tables 10, 11, 12 and 13.

Evaluation of the candidate solution approaches

The crossover rate for MA and GA is set between 60 and 70%, and the mutation rate for GA is between 10 and 15%. The value is taken as 0.5. For test cases 1 to 16, the maximum and minimum values of the three fuzzy objective membership functions are obtained by using CPLEX to calculate the two objectives separately. For test cases 17 to 35, due to the larger problem size, it is not possible to find the optimal solution using CPLEX for the two objectives separately without exceeding the available computer memory, and only feasible solutions can be obtained. Therefore, this paper uses the MA to calculate the objective values for the two targets separately and compares them with the objective values of the feasible solutions obtained by using CPLEX, selecting the better values as the maximum and minimum values for the two objective functions. The reliability of the tugboat matching scheme is evaluated using the criteria from Sect. 5.3.2. Each algorithm is run 20 times for each test case. For the results presented in Tables 14 and 15 (where bold numbers indicate the optimal solutions, and those marked with an asterisk "*" indicate feasible solutions), the notation "Heuristic still looking" indicates that after the exact algorithm in CPLEX fails to compute the solution, it automatically switches to using a heuristic algorithm to search for the optimal solution.

As shown in Table 14, for test cases 1 to 16, the objective values obtained by CPLEX and MA are the same, but the solution time taken by MA is much longer than that of CPLEX, indicating that for small-scale instances, CPLEX is more efficient. For test cases 17 to 25, in the first three cases, CPLEX took more than 9000 s and

	CPLEX		MA		
Example(Population size, Iterations' number)	Objective value	CPU time (seconds)	Objective value	Average value	CPU time (seconds)
1(50,100)	**0.1568**	1.26	**0.1568**	0.1635	63.97
2(50,100)	**0.1369**	3.59	**0.1369**	0.1526	86.25
3(50,100)	**0.1635**	3.98	**0.1635**	0.1754	96.56
4(50,100)	**0.1723**	5.66	**0.1723**	0.1862	103.28
5(50,100)	**0.1295**	5.39	**0.1295**	0.1358	132.57
6(100,150)	**0.1362**	6.28	**0.1362**	0.1542	139.54
7(100,150)	**0.1623**	16.57	**0.1623**	0.1763	206.91
8(100,150)	**0.1262**	26.36	**0.1262**	0.1386	263.79
9(100,150)	**0.1352**	36.19	**0.1352**	0.1559	321.68
10(100,150)	**0.1693**	53.26	**0.1693**	0.1843	362.93
11(200,300)	**0.1529**	168.26	**0.1529**	0.1628	423.75
12(200,300)	**0.1753**	963.02	**0.1753**	0.1962	495.23
13(200,300)	**0.1256**	1963.28	**0.1256**	0.1389	537.95
14(200,300)	**0.1385**	2953.59	**0.1385**	0.1526	639.17
15(200,300)	**0.1665**	3265.31	**0.1665**	0.1745	763.52
16(250,350)	**0.1346**	6295.26	**0.1346**	0.1536	889.56
17(250,350)	0.1593*(Heuristic still looking.)	9000+	0.1526	0.1695	936.41
18(250,350)	0.1636*(Heuristic still looking.)	9000+	0.1562	0.1953	986.09
19(250,350)	0.1561*(Heuristic still looking.)	9000+	0.1399	0.1846	1035.26
20(250,350)	--	--	0.1572	0.1936	1187.36
21(300,400)	--	--	0.1365	0.1625	1236.75
22(300,400)	--	--	0.1986	0.2213	1386.63
23(300,400)	--	--	0.1257	0.1612	1567.28
24(300,400)	--	--	0.1729	0.2063	1652.53
25(300,400)	--	--	0.1356	0.1876	1795.33

Table 14. Objective function values and CPU times of the considered solution approaches.

encountered the "Heuristic still looking" situation, resulting in feasible solutions. However, MA achieved smaller objective values in a much shorter time, less than 9000 s. For the subsequent large-scale test cases, CPLEX still could not provide a feasible solution within the set limit of 10,000 s, while MA was still able to achieve better objective values within 1600 s of solution time. Therefore, based on the content described in this section, it is indicated that the MA algorithm has a certain level of efficiency.

In Tables 14 and 15, the maximum test cases that GA and SA can solve to reach the optimal value are Test case 12 and Test case 9, respectively, while MA can reach up to Test case 16; and for all test cases, the objective values obtained by MA are lower than those of GA and SA, which indicates that MA has higher precision than GA and SA in solving the problem described in this paper. For all test cases, although the solution time of MA is slightly higher than that of GA and SA, this is because as the problem size increases, the time taken by MA increases more, which is due to the time spent on the dynamic neighborhood search operation in MA. However, this process helps to enhance the local search capability of the MA algorithm, thereby obtaining better solutions. Therefore, based on the content described in this section, it is indicated that the MA algorithm is more likely to escape from local optima compared to the pure GA and SA algorithms, demonstrating a certain level of efficiency.

Case analysis

Case description

This study conducts a case analysis with Guangzhou Port as an example, selecting data from four important berthing bases: Huangpu New Port Operating Area, Xinsha Port Operating Area, Xiaohu Operating Area, and Nansha Operating Area.

Based on the actual information of inbound and outbound vessels at Guangzhou Port, 20 task quantities were generated. Each task has 6 optional tugboat matching schemes. Other specific information is as shown in Tables 16 and 17.

Determination of reliability indicators for tugboat matching schemes

For each task of the day, there are 6 alternative schemes for tugboat matching. Considering the ship age, fault conditions, and maneuverability of the tugboats, these aspects are set as evaluation indicators. The decision-making group consists of 10 individuals, including on-site staff and supervisors from Guangzhou Port. After collecting the evaluation data from the decision-making group on the candidate tugboat matching schemes, the method described in Sect. 3.2 is used, and programmed in MATLAB R2015b to obtain the related reliability values in Table 18.

	GA			SA		
Example(Population size, Iterations' number)	Objective value	Average value	CPU time (seconds)	Objective value	Average value	CPU time (seconds)
1(50,100)	**0.1568**	0.1658	58.21	**0.1568**	0.1826	61.25
2(50,100)	**0.1369**	0.1526	76.29	**0.1369**	0.1569	81.29
3(50,100)	**0.1635**	0.1748	83.96	**0.1635**	0.1741	93.48
4(50,100)	**0.1723**	0.18963	92.48	**0.1723**	0.1869	96.75
5(50,100)	**0.1295**	0.1452	126.54	**0.1295**	0.1520	130.31
6(100,150)	**0.1362**	0.1523	127.23	**0.1362**	0.1594	132.52
7(100,150)	**0.1623**	0.1749	186.08	**0.1623**	0.1743	196.35
8(100,150)	**0.1262**	0.1356	245.68	**0.1262**	0.1359	253.22
9(100,150)	**0.1352**	0.1409	305.87	**0.1352**	0.1421	315.26
10(100,150)	**0.1693**	0.1741	342.81	0.1759	0.1857	354.98
11(200,300)	**0.1529**	0.1639	403.86	0.1638	0.1745	415.62
12(200,300)	**0.1753**	0.1926	470.15	0.1896	0.1952	486.71
13(200,300)	0.1328	0.1628	516.49	0.1467	0.1523	526.96
14(200,300)	0.1432	0.1565	603.75	0.1549	0.1657	612.49
15(200,300)	0.1752	0.1946	732.46	0.1829	0.1954	745.16
16(250,350)	0.1529	0.1829	852.31	0.1586	0.1762	864.03
17(250,350)	0.1636	0.1755	920.62	0.1649	0.1849	931.07
18(250,350)	0.2149	0.2309	958.16	0.2186	0.2311	962.15
19(250,350)	0.1988	0.2067	986.45	0.2049	0.2149	995.36
20(250,350)	0.2186	0.2316	1025.92	0.2269	0.2374	1095.36
21(300,400)	0.1839	0.1935	1106.34	0.1926	0.2068	1153.21
22(300,400)	0.2319	0.2508	1189.67	0.2463	0.2832	1396.35
23(300,400)	0.1548	0.1635	1356.05	0.1629	0.1862	1396.37
24(300,400)	0.1926	0.2118	1587.32	0.2116	0.2261	1613.88
25(300,400)	0.1578	0.1832	1604.92	0.1643	0.1953	1680.89

Table 15. Objective function values and CPU times of the considered solution approaches.

Obtaining a single compromised solution

Based on the data from Sect. 5.3.1 and 5.3.2, CPLEX 12.8 was used to solve the objectives 1 and 2, resulting in the game Table 19. Consequently, the weights $w_1^{*\,min} = 3440$, $w_1^{*\,max} = 120997.288135593$, $w_2^{*\,min} = 0.1586$, $w_2^{*\,max} = 6.4131$are obtained. By transforming the value of δ, which varies from 0 to 1, Table 20 is obtained. Table 21 shows the results of task selection for tugboat matching schemes when $\delta = 0.6$, and Figs. 1, 2 and 3 are derived based on Table 20.

It can be seen from Fig. 1 that costs decrease as the value of a increases. Figure 2 shows that total reliability decreases with the increase of δ value, while Fig. 3 displays that the weighted sum of the objective values first increases and then decreases with the increase of δ value, and all these variations are nonlinear. This indicates that in order to satisfy the weighted sum of the objective values, the optimal solution is achieved when the value of δ is 0.6, with objective function 1 valued at 26,720 and objective function 2 valued at 5.9463, as shown in Table 21. From the above data, it can be seen that tugboat companies can adjust parameters to better control their operating costs.

In conclusion, the variation in the multi-objective weighted values affects the scheduling of tugboat dispatching plans. Therefore, managers of tugboat enterprises need to correctly analyze and predict changes in the port environment to determine the intensity of port support operations, that is, to select appropriate multi-objective weighted values while bearing the corresponding risk level, thereby making scientific arrangements for tugboat dispatching plans.

Conclusion

This paper takes tugboat dispatching as the research subject and, considering the existence of various tugboat matching schemes, establishes a multi-objective mixed linear programming model. Then, based on the characteristics of this model, a new solution encoding method based on priority coding is proposed and applied within the framework of the MA. Finally, through comparative analysis with exact optimization methods (CPLEX) and genetic algorithms, as well as simulated annealing algorithms (recognized metaheuristics), the effectiveness of the proposed algorithm is demonstrated using several test cases. A large number of experimental results show that the proposed algorithm can provide optimal and high-quality solutions within an acceptable computation time, even for large-scale numerical examples. It also proposes a two-stage solution approach and demonstrates the applicability of the model through a case study, providing decision support and methodological guidance.

Future research will concentrate on the following areas: (1) Considering the fact that the water depth of port operations changes over time, and if the water depth does not meet the required specifications for vessel

	The service time for each tugboat matching schemes (min)						The number of tugboats required for each tugboat matching scheme					
Task	1	2	3	4	5	6	1	2	3	4	5	6
1	93	82	73	66	61	56	1	1	1	1	1	2
2	97	88	72	67	62	54	1	1	1	1	2	2
3	94	87	74	66	63	53	1	1	1	2	3	2
4	95	83	76	68	62	53	2	2	4	3	3	3
5	92	84	77	66	61	54	2	5	3	4	3	5
6	90	80	75	65	60	51	1	1	1	1	1	2
7	95	81	70	65	62	52	1	1	1	1	2	2
8	97	87	76	69	60	55	1	1	1	2	3	2
9	95	86	75	66	63	51	2	2	4	3	3	3
10	91	83	71	68	60	55	2	5	3	4	3	5
11	90	80	75	65	61	53	1	1	1	1	1	2
12	95	82	70	69	67	54	1	1	1	1	2	2
13	97	86	72	69	62	50	1	1	1	2	3	2
14	93	80	77	68	64	58	2	2	4	3	3	3
15	96	88	79	65	60	55	2	5	3	4	3	5
16	91	80	74	68	61	56	1	1	1	1	1	2
17	96	85	70	66	60	52	1	1	1	1	2	2
18	98	89	73	67	63	57	1	1	1	2	3	2
19	95	84	75	69	60	59	2	2	4	3	3	3
20	94	85	78	67	66	53	2	5	3	4	3	5

Table 16. Service time and number of tugboats required for tugboat matching scheme.

	The horsepower required for each tugboat matching scheme (HP)					
Task	1	2	3	4	5	6
1	1600	3000	4000	5000	6000	3000
2	3000	4000	5000	6000	3000	4000
3	4000	5000	6000	3000	3000	4000
4	5000	6000	3000	4000	5000	6000
5	6000	3000	5000	4000	6000	3000
6	1600	3000	4000	5000	6000	3000
7	3000	4000	5000	6000	3000	4000
8	4000	5000	6000	3000	3000	4000
9	5000	6000	3000	4000	5000	6000
10	6000	3000	5000	4000	6000	3000
11	1600	3000	4000	5000	6000	3000
12	3000	4000	5000	6000	3000	4000
13	4000	5000	6000	3000	3000	4000
14	5000	6000	3000	4000	5000	6000
15	6000	3000	5000	4000	6000	3000
16	1600	3000	4000	5000	6000	3000
17	3000	4000	5000	6000	3000	4000
18	4000	5000	6000	3000	3000	4000
19	5000	6000	3000	4000	5000	6000
20	6000	3000	5000	4000	6000	3000

Table 17. The horsepower required for tugboat matching scheme.

Task	Candidate tugboat matching schemes					
	1	2	3	4	5	6
1	0	0.0411	0.0782	0.1533	0.2393	0.4881
2	0	0.1485	0.164	0.1778	0.1810	0.3287
3	0	0.1206	0.1436	0.1694	0.2734	0.2930
4	0	0.1423	0.1562	0.1872	0.2524	0.2619
5	0	0.0614	0.1508	0.1734	0.2693	0.3452
6	0	0.0377	0.1764	0.1853	0.2750	0.3256
7	0	0.0799	0.1851	0.1979	0.2001	0.3371
8	0	0.1212	0.1675	0.1745	0.2328	0.3040
9	0	0.1143	0.1293	0.2244	0.2366	0.2954
10	0	0.0980	0.1743	0.1766	0.2059	0.3451
11	0	0.0243	0.1586	0.2136	0.2612	0.3423
12	0	0.1193	0.2020	0.2084	0.2121	0.2582
13	0	0.0698	0.1776	0.1778	0.2248	0.3499
14	0	0.0653	0.1509	0.2137	0.2437	0.3264
15	0	0.1280	0.1320	0.2326	0.2454	0.2620
16	0	0.0719	0.1508	0.2069	0.2115	0.3589
17	0	0.0853	0.1339	0.1941	0.2282	0.3585
18	0	0.1078	0.1209	0.2163	0.2694	0.2857
19	0	0.1841	0.1882	0.1953	0.2082	0.2242
20	0	0.0223	0.1630	0.1729	0.3189	0.3229

Table 18. Reliability of candidate tugboat matching schemes for each task.

	$Z_1(X)$	$Z_2(X)$
$X^{(1)}$	3440	0.1586
$X^{(2)}$	120997.288135593	6.4131

Table 19. Bidirectional objective game table.

δ	Objective function 1	Objective function 2	Weighted sum of objective values
0	120997.288135593	6.4131	0
0.1	40,240	6.3925	0.0343
0.2	39,520	6.383	0.0652
0.3	33,840	6.267	0.0939
0.4	33,840	6.267	0.1175
0.5	27,840	6.0274	0.1346
0.6	26,720	5.9463	0.1487
0.7	12,080	4.4302	0.1466
0.8	4640	3.1997	0.1109
0.9	3440	2.8808	0.0565
1	3440	0.1586	−2.2204

Table 20. Variation of objective values.

operations during a specific period, tugboats must cease operations. Future studies will incorporate the realistic consideration of such situations. (2) When faced with uncertainties caused by natural conditions or political factors, tugboat dispatching plans should be adaptable to these changes. (3) As scheduling is classified as an NP problem, future work will delve into developing computational methods specifically designed to address this type of problem.

In addition, for the future development of port tugboat scheduling, the application and promotion of intelligent technology in non-automated ports should be strengthened to enhance data collection and analysis capabilities. At the same time, attention should be paid to personnel training, cultivating compound talents who understand both technology and scheduling. In automated ports, it is necessary to continuously optimize scheduling algorithms and models to better adapt to complex and changing situations. It is necessary to

Task	Candidate tugboat matching schemes					
	1	2	3	4	5	6
1	0	0	0	0	0	1
2	0	0	0	0	0	1
3	0	0	0	0	0	1
4	0	0	0	0	1	0
5	0	0	0	0	1	0
6	0	0	0	0	1	0
7	0	0	0	0	0	1
8	0	0	0	0	0	1
9	0	0	0	0	0	1
10	0	0	0	0	1	0
11	0	0	0	0	1	0
12	0	0	0	1	0	0
13	0	0	0	0	0	1
14	0	0	0	0	0	1
15	0	0	0	0	1	0
16	0	0	0	0	0	1
17	0	0	0	0	0	1
18	0	0	0	0	0	1
19	0	1	0	0	0	0
20	0	0	0	0	1	0

Table 21. Results of task selection for tugboat matching schemes when $\delta = 0.6$.

strengthen information sharing and cooperation among ports to promote overall development. In the future, we can conduct in-depth research on tugboat scheduling strategies under special conditions such as extreme weather, and further explore the potential application of new intelligent technologies in tugboat scheduling. From a broader perspective of the port ecosystem, it is also possible to study the coordinated optimization of tugboat scheduling and other aspects, providing stronger support for the sustainable development of ports.

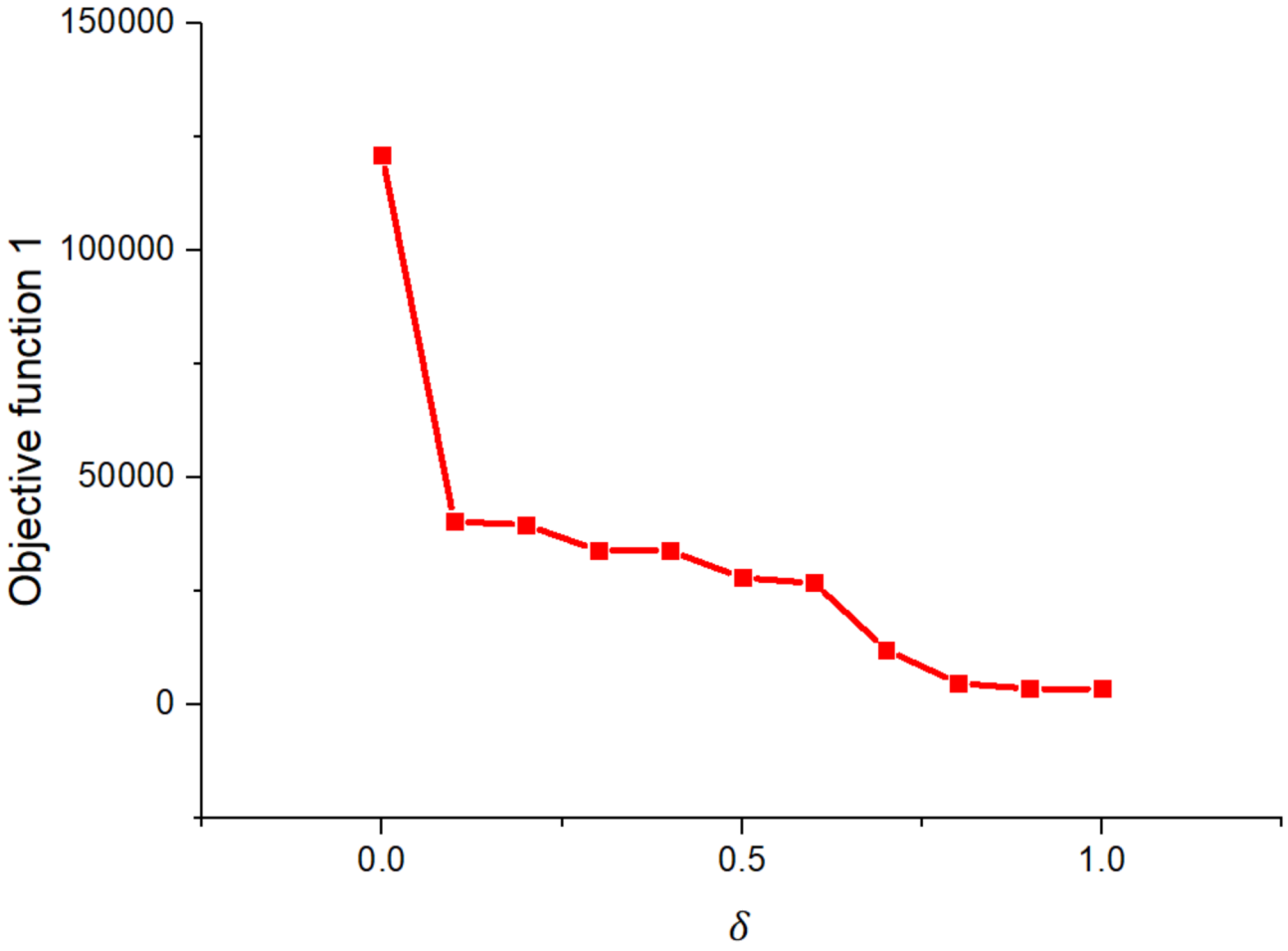


Fig. 1. Variation of objective function 1.

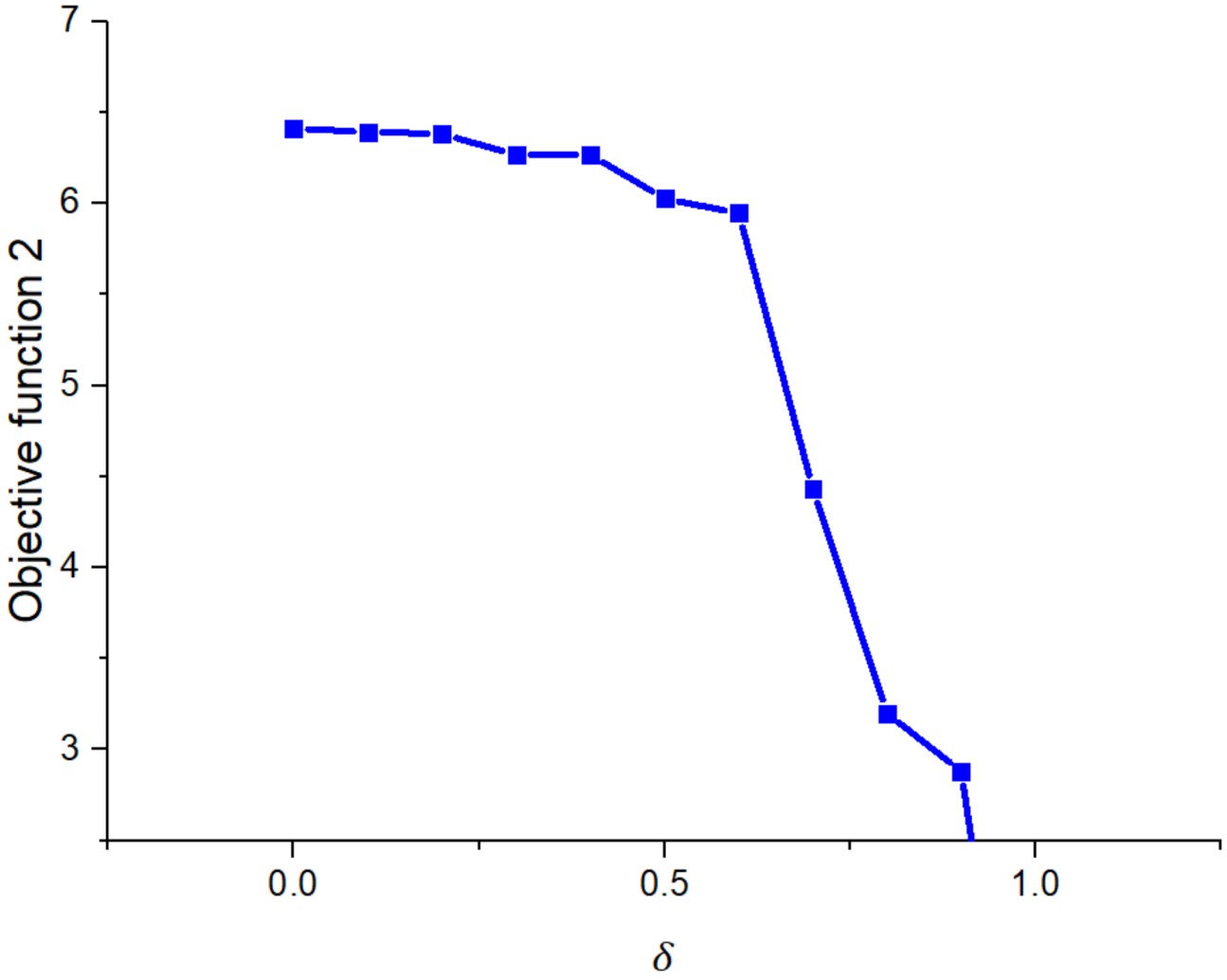


Fig. 2. Variation of objective function 2.

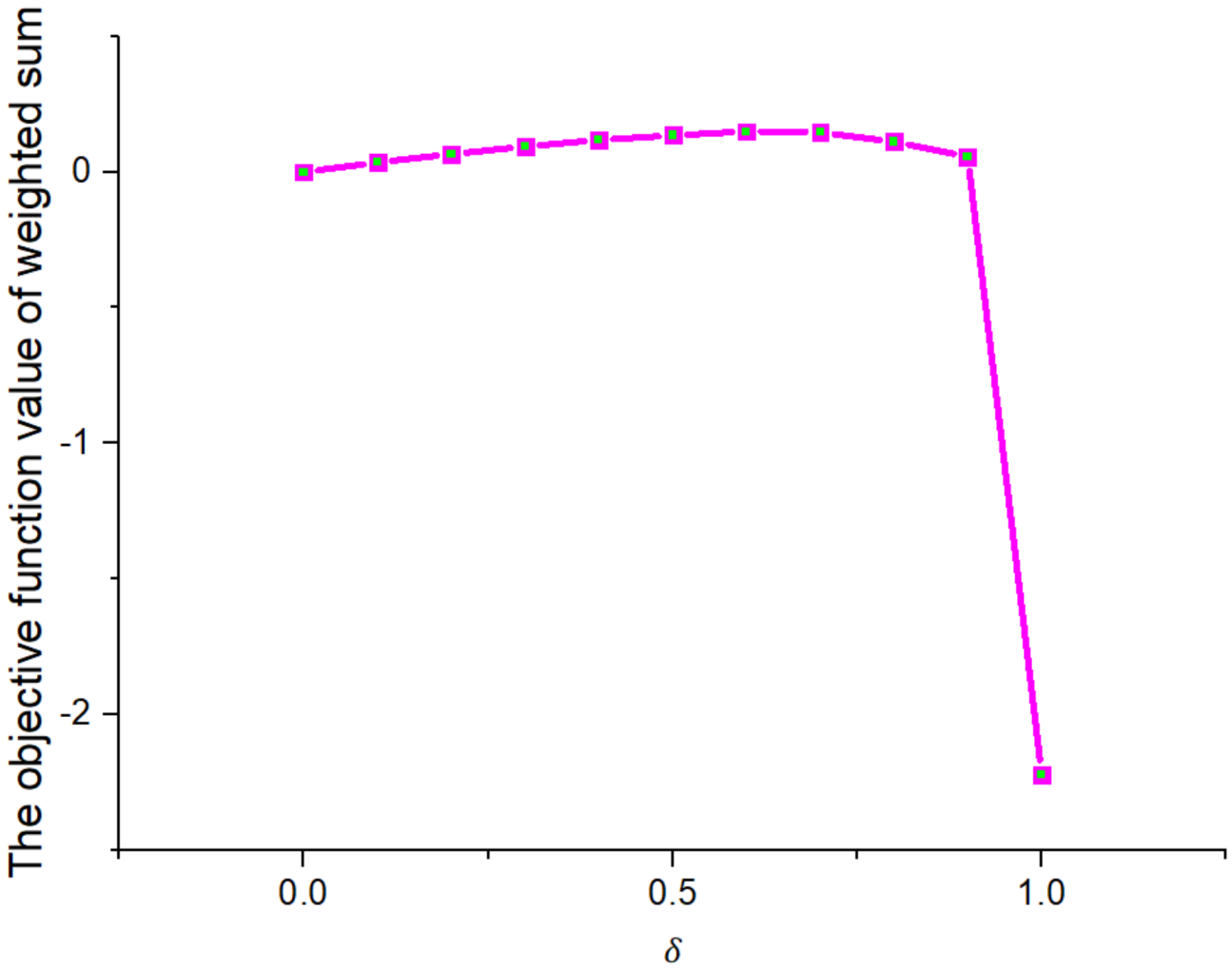


Fig. 3. Variation of the weighted sum of objective values.

Data availability

Data is provided within the manuscript.

Received: 24 January 2025; Accepted: 24 March 2025
Published online: 07 April 2025

References

1. Hao, L., Jin, J. & Zhao, K. Joint scheduling of barges and tugboats for river–sea intermodal transport. *Transp. Res. Part E.* **173**, 103097 (2023).
2. Yang, Z., Cao, X., Xu, R., Hong, W. & Sun, S. Applications of chaotic quantum adaptive satin Bower bird optimizer algorithm in berth-tugboat-quay crane allocation optimization. *Expert Syst. Appl.* **237**, 121471 (2024).
3. García-Morales, R., Baquerizo, A. & Losada, M. Port management and multiple-criteria decision making under uncertainty. *Ocean Eng.* **104**, 31–39 (2015).
4. Pesce, M. et al. Linkov, I. Selecting sustainable alternatives for cruise ships in Venice using multi-criteria decision analysis. *Sci. Total Environ.* **642**, 668–678 (2018).
5. Yang, S., Pan, Y. & Zeng, S. Decision making framework based fermatean fuzzy integrated weighted distance and TOPSIS for green low-carbon Port evaluation. *Eng. Appl. Artif. Intell.* **114**, 105048 (2022).
6. Díaz, H. & Soares, C. Decision-making model for the selection of floating wind logistic support ports. *Ocean Eng.* **281**, 114768 (2023).
7. Ashraf, S., Garg, H. & Kousar, M. An industrial disaster emergency decision-making based on China's Tianjin City Port explosion under complex probabilistic hesitant fuzzy soft environment. *Eng. Appl. Artif. Intell.* **123**, 106400 (2023).
8. Zhen, L., Wang, K., Wang, S. & Qu, X. Tug scheduling for hinterland barge transport: A branch-and-price approach. *Eur. J. Oper. Res.* **265**, 119–132 (2018).
9. Wu, L., Yang, D., Wang, S. & Yuan, Y. Evacuating offshore working barges from a land reclamation site in storm emergencies. *Transp. Res. Part E.* **137**, 101902 (2020).
10. Zhu, S. et al. Green logistics oriented tug scheduling for inland waterway logistics. *Adv. Eng. Inform.* **49**, 101323 (2021).
11. Abou Kasm, O., Diabat, A. & Bierlaire, M. Vessel scheduling with pilotage and tugging considerations. *Transp. Res. Part E.* **148**, 102231 (2023).

12. Zhong, H., Zhang, Y. & Gu, Y. A. Bi-objective green tugboat scheduling problem with the tidal Port time windows. *Transp. Res. Part D*. **110**, 103409 (2022).
13. Petris, M., Pellegrini, P. & Pesenti, R. Models and algorithms for an integrated vessel scheduling and tug assignment problem within a Canal harbor. *Eur. J. Oper. Res.* **300** (3), 1120–1135 (2022).
14. Wei, X., Jia, S., Meng, Q. & Tan, K. Tugboat scheduling for container ports. *Transp. Res. Part E*. **142**, 102071 (2020).
15. Kang, L., Meng, Q. & Tan, K. Tugboat scheduling under ship arrival and tugging process time uncertainty. *Transp. Res. Part E*. **144**, 102125 (2020).
16. Jia, S., Li, S., Lin, X. & Chen, X. Scheduling tugboats in a seaport. *Transport. Sci.* **55** (6), 1370–1391 (2021).
17. Abou Kasm, O., Diabat, A. & Ozbay, K. Vessel scheduling under different tugboat allocation policies. *Comput. Ind. Eng.* **177**, 108902 (2023).
18. Kuhn, H. W. The Hungarian method for the assignment problem. *Nav. Res. Logist.* **52**, 7–27 (2005).
19. Pentico, D. W. Assignment problems: A golden anniversary survey. *Eur. J. Oper. Res.* **176** (2), 774–793 (2007).
20. Zhang, Z. & Daigle, J. Analysis of job assignment with batch arrivals among heterogeneous servers. *Eur. J. Oper. Res.* **217**, 149–161 (2012).
21. Laalaoui, Y. & M'Hallah, R. A binary multiple knapsack model for single machine scheduling with machine unavailability. *Comput. Oper. Res.* **72**, 71–82 (2016).
22. Dimitrov, N. B., Solow, D. & Szmerekovsky, J. Emergency relocation of items using single trips: special cases of the multiple knapsack assignment problem. *Eur. J. Oper. Res.* **258**, 938–942 (2017).
23. Wang, X., Liang, Y., Wei, X., Chew, C. & Ek, P. An adaptive large neighborhood search algorithm for the tugboat scheduling problem. *Comput. Ind. Eng.* **177**, 109039 (2023).
24. Zhong, H., Zhang, Y. & Gu, Y. A. Bi-objective green tugboat scheduling problem with the tidal Port time windows. *Transp. Res. Part. D: Transp. Environ.* **110**, 103409 (2022).
25. Vakili, R., Shirazi, M. A. & Gitinavard, H. Multi-echelon green open-location-routing problem: a robust-based stochastic optimization approach. *Scientia Iranica*, **28**, 985–1000 (2020).
26. Gitinavard, H., Ghodsypour, S. H. & Shirazi, M. A. A bi-objective multi-echelon supply chain model with Pareto optimal points evaluation for perishable products under uncertainty. *Scientia Iranica*, **26**, 2952–2970 (2019).
27. Gitinavard, H., Mohagheghi, V., Shirazi, M. A. & Mousavi, S. M. Biofuel supply chain network design in competitive feedstock markets: an interactive possibilistic programming-based intelligent agent system. *Comput. Ind. Eng.*, **200**, 110849 (2025).
28. Gitinavard, H., Mohagheghi, V., Mousavi, S. M. & Makui, A. A new bi-stage interactive possibilistic programming model for perishable logistics distribution systems under uncertainty. *Expert Syst. Appl.* **238**, 122121 (2024).
29. Gitinavard, H., Shirazi, M. & Zarandi, M. A possibilistic programming approach for biomass supply chain network design under hesitant fuzzy membership function Estimation. *Scientia Iranica*. **31** (18), 1606–1624 (2024).
30. Sun, G., Xu, Z., Yu, H. & Chang, V. Dynamic network function provisioning to enable network in box for industrial applications. *IEEE Trans. Industr. Inf.* **99**, 1–1 (2020).
31. Zhou, L. et al. Numerical study of the cavitation performance of an Ice-Blocked propeller considering the free surface effect. *Water* **16**, 22 (2024).
32. Zhou, L., Zheng, S., Ding, S., Xie, C. & Liu, R. Influence of propeller on brash ice loads and pressure fluctuation for a reversing Polar ship. *Ocean Eng.* **280**, 114624 (2023).
33. Zhou, L., Gao, J., Xu, S. & Bai, X. A numerical method to simulate ice drift reversal for moored ships in level ice. *Cold Reg. Sci. Technol.* **152**, 35–47 (2018).
34. Zhou, L., Sun, Q., Ding, S., Han, S. & Wang, A. A. Machine-Learning-Based Method for Ship Propulsion Power Prediction in Ice. Journal of marine science and engineering, 11(7). (2023).
35. He, H. et al. Practical Tracking Method based on Best Buddies Similarity. Cyborg and bionic systems, 4. (2024).
36. Cao, X. et al. *Method of Human-Like Compliant Assembly Based on Variable Admittance Control for Space Maintenance*4 (Cyborg and bionic systems, 2023).
37. Xu, X. & We, Z. Dynamic pickup and delivery problem with transshipments and LIFO constraints. *Comput. Ind. Eng.* **175**, 108835 (2023).
38. Chen, Z. et al. Multi-workflow dynamic scheduling in product design: A generalizable approach based on meta-reinforcement learning. *J. Manuf. Syst.* **79**, 334–346 (2025).
39. Zou, Z., Yang, S. & Zhao, L. Dual-loop control and state prediction analysis of QUAV trajectory tracking based on biological swarm intelligent optimization algorithm. *Sci. Rep.* **14**, 19091 (2024).
40. Tejani, G., Shar, S., Mashru, N., Patel, P. & Jangir, P. Optimization of truss structures with two archive-boosted MOHO algorithm. *Alexandria Eng. J.* **120**, 296–317 (2025).
41. Tejani, G., Mashru, N., Patel, P., Sharma, S. & Celik, E. Application of the 2-archive multi-objective cuckoo search algorithm for structure optimization. *Sci. Rep.* **14**, 31533 (2024).
42. Wei, M., Yang, S., Wu, W. & Sun, B. A multi-objective fuzzy optimization model for multi-type aircraft flight scheduling problem. *Transport* **39** (4), 313–322 (2024).
43. Zhu, C., Liu, X., Ding, W. & Zhang, S. Cloud model-based multi-stage multi-attribute decision-making method under probabilistic interval-valued hesitant fuzzy environment. *Expert Syst. Appl.* **255**, 124595 (2024).
44. Mavrotas, G. Effective implementation of the ε-constraint method in Multi-Objective mathematical programming problems. *Appl. Math. Comput.* **213** (2), 455–465 (2009).
45. Al-E-Hashem, S. M. J. M., Aryanezhad, M. B. & Sadjadi, S. J. An efficient algorithm to solve a multi-objective robust aggregate production planning in an uncertain environment. *Int. J. Adv. Manuf. Technol.* **58** (5–8), 765–782 (2012).
46. Selim, H. & Ozkarahan, I. A supply chain distribution network design model: an interactive fuzzy goal programming-based solution approach. *Int. J. Adv. Manuf. Technol.* **31**, 401–418 (2008).
47. Matic, D., Kratica, J. & Maksimovic, Z. Solving the minimum edge-dilation K -center problem by genetic algorithms. *Comput. Ind. Eng.* **113**, 282–293 (2017).
48. Jamrus, T. & Chien, C. F. Extended priority-based hybrid genetic algorithm for the less-than-container loading problem. *Comput. Ind. Eng.* **96** (C), 227–236 (2016).
49. Moscato, P. & Norman, M. G. A memetic approach for the traveling salesman problem implementation of a computational ecology for combi-combinatorial optimization on message-passing systems. In: (eds Valero, M., Onate, E., Jane, M., Larriba, J. L. & Suarez, B.) Parallel Computing and Transporter Applications. Amsterdam. IOS, 177–186. (1992).
50. Moscato, P. & Cotta, C. A. *Gentle Introduction To Memetic Algorithms// Handbook of Metaheuristics*105–144 (Springer US, 2010).
51. Boudia, M. A memetic algorithm with dynamic population management for an integrated production–distribution problem. *Eur. J. Oper. Res.* **195** (3), 703–715 (2009).
52. Tavakkoli-Moghaddam, R., Safaei, N. & Sassani, F. A memetic algorithm for the flexible flow line scheduling problem with processor blocking. *Comput. Oper. Res.* **36** (2), 402–414 (2009).
53. Kim, S. S., Smith, A. E. & Lee, J. H. A memetic algorithm for channel assignment in wireless FDMA systems. *Comput. Oper. Res.* **34** (6), 1842–1856 (2007).
54. Elmekkawy, T. Y. & Liu, S. A new memetic algorithm for optimizing the partitioning problem of tandem AGV systems. *Int. J. Prod. Econ.* **118** (2), 508–520 (2009).

55. Choudhary, A., Sarkar, S. & Settur, S. A carbon market sensitive optimization model for integrated forward–reverse logistics. *Int. J. Prod. Econ.* **164** (6), 433–444 (2015).
56. Roghanian, E. & Pazhoheshfar, P. An optimization model for reverse logistics network under stochastic environment by using genetic algorithm. *J. Manuf. Syst.* **33** (3), 348–356 (2014).
57. Tari, F. G. & Hashemi, Z. A priority based genetic algorithm for nonlinear transportation costs problems. *Comput. Ind. Eng.* **96**, 86–95 (2016).
58. Afrouzy, Z. A., Nasseri, S. H. & Mahdavi, I. A genetic algorithm for supply chain configuration with new product development. *Comput. Ind. Eng.* **101**, 440–454 (2016).
59. Gupta, A., Heng, C. K. & Ong, Y. S. A generic framework for multi-criteria decision support in eco-friendly urban logistics systems. *Expert Syst. Appl.* **71** (C), 288–300 (2017).
60. Fung, R. Y. K., Liu, R. & Jiang, Z. A memetic algorithm for the open capacitated Arc routing problem. *Transp. Res. Part. E Logistics Transp. Rev.* **50** (2), 53–67 (2013).
61. Karaoglan, I. & Altiparmak, F. A memetic algorithm for the capacitated location-routing problem with mixed backhauls. *Comput. Oper. Res.* **55**, 200–216 (2015).
62. Hasani, A. & Khosrojerdi, A. Robust global supply chain network design under disruption and uncertainty considering resilience strategies: A parallel memetic algorithm for a real-life case study. *Transp. Res. Part. E Logistics Transp. Rev.* **87**, 20–52 (2016).

Author contributions

Conceptualization: Yangjun Ren, Di Liu, Mengchi Li, Botang Li. Methodology: Yangjun Ren, Yushun Lei, Di Liu, Bo-tang Li. Formal analysis and investigation: Yangjun Ren, Botang Li, Yushun Lei, Di Liu, Jianjun Tu. Writing—original draft preparation: Di Liu, Mengchi Li, Botang Li. Writing—review and editing: Yan Zhou, Botang Li, Mengchi Li. Supervision: Yan Zhou, Mengchi Li, Yushun Lei, Di Liu, Jianjun Tu.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Y.R.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by-nc-nd/4.0/.

© The Author(s) 2025