



## Article

# Smooth and Efficient Path Planning for Car-like Mobile Robot Using Improved Ant Colony Optimization in Narrow and Large-Size Scenes

Likun Li <sup>1</sup>, Liyu Jiang <sup>3</sup>, Wenzhang Tu <sup>4</sup>, Liquan Jiang <sup>2,5,\*</sup> and Ruhan He <sup>6</sup>

<sup>1</sup> School of Management, Huazhong University of Science and Technology, 1037 Luoyu Road, Wuhan 430074, China; lilikun0725@gmail.com

<sup>2</sup> The State Key Laboratory of New Textile Materials and Advanced Processing Technologies, Wuhan Textile University, 1 Yangguang Avenue, Wuhan 430200, China

<sup>3</sup> Hubei Institute of Measurement and Testing Technology, 2 Maodianshan Middle Road, Wuhan 430223, China

<sup>4</sup> Hubei Province Fibre Inspection Bureau, 6 Gongping Road, Wuhan 430064, China; tuwenzhang@me.com

<sup>5</sup> Hubei Key Laboratory of Digital Textile Equipment, Wuhan Textile University, 1 Yangguang Avenue, Wuhan 430200, China

<sup>6</sup> School of Computer Science and Artificial Intelligence, Wuhan Textile University, 1 Yangguang Avenue, Wuhan 430200, China; heruhan@wtu.edu.cn

\* Correspondence: ljjiang@wtu.edu.cn

**Abstract:** Car-like mobile robots (CLMRs) are extensively utilized in various intricate scenarios owing to their exceptional maneuverability, stability, and adaptability, in which path planning is an important technical basis for their autonomous navigation. However, path planning methods are prone to inefficiently generate unsmooth paths in narrow and large-size scenes, especially considering the chassis model complexity of CLMRs with suspension. To this end, instead of traditional path planning based on an integer order model, this paper proposes fractional-order enhanced path planning using an improved Ant Colony Optimization (ACO) for CLMRs with suspension, which can obtain smooth and efficient paths in narrow and large-size scenes. On one hand, to improve the accuracy of the kinematic model construction of CLMRs with suspension, an accurate fractional-order-based kinematic modelling method is proposed, which considers the dynamic adjustment of the angle constraints. On the other hand, an improved ACO-based path planning method using fractional-order models is introduced by adopting a global multifactorial heuristic function with dynamic angle constraints, adaptive pheromone adjustment, and fractional-order state-transfer models, which avoids easily falling into a local optimum and unsmooth problem in a narrow space while increasing the search speed and success rate in large-scale scenes. Finally, the proposed method's effectiveness is validated in both large-scale and narrow scenes, confirming its capability to handle various challenging scenarios.

**Keywords:** car-like mobile robot; path planning; ant colony optimization; fractional-order; narrow and large-size scene



**Citation:** Li, L.; Jiang, L.; Tu, W.; Jiang, L.; He, R. Smooth and Efficient Path Planning for Car-like Mobile Robot Using Improved Ant Colony Optimization in Narrow and Large-Size Scenes. *Fractal Fract.* **2024**, *8*, 157. <https://doi.org/10.3390/fractalfract8030157>

Academic Editors: Norbert Herencsar, Kishore Bingi and Abhaya Pal Singh

Received: 31 January 2024

Revised: 4 March 2024

Accepted: 7 March 2024

Published: 10 March 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Path planning is a critical technology in the field of mobile robotics, enabling a mobile robot to efficiently navigate from its starting point to a designated target while circumventing obstacles within a given environment. It serves as a fundamental component of autonomous navigation and intelligent decision-making in mobile robot systems [1–3]. Meanwhile, Car-Like Mobile Robots (CLMRs) play an important role in the fields of warehousing and logistics, inspection, and distribution, etc. [4–6], and their chassis is equipped with a steering mechanism and suspension system [7], which makes CLMRs have good load capacity, passability, and flexibility. CLMRs are commonly utilized in environments

characterized by a combination of large-scale areas and narrow spaces. Examples include neighborhoods with narrow alleys, dynamic manufacturing plants, or wild landscapes with dense vegetation [8,9]. These types of scenes impose greater demands on the path planning capabilities of CLMRs, requiring them to efficiently and accurately plan smooth paths.

Recently, path planning methods have emerged as a highly prominent area of research, captivating the attention of scholars worldwide, and researchers have extensively explored and developed innovative techniques across a diverse range of scenarios and for various types of robots [10–12]. In general, research on path planning usually focuses on two major aspects: the construction of the robot chassis model and the optimization of path planning methods. Fortunately, the fractional-order method is commonly used for modelling and optimization, which is a mathematical tool dealing with non-integer order calculus [13–16]. It extends the traditional integer order calculus by allowing derivatives or integrals of non-integer orders to exist in the model. Fractional-order methods have gained significant attention in capturing the behavior of complex nonlinear systems. These methods offer a more accurate representation of system dynamics by incorporating fractional-order differential equations, which enable the modelling of properties like nonlocal dependence and nonsmooth behavior, allowing fractional-order models to better fit the behavior of real systems and provide more accurate predictions and analyses [17–20]. Therefore, this paper aims to utilize the fractional-order approach to extend the conventional path planning method based on an integer-order model, to devise a path planning scheme that is not only smoother but also more efficient.

In terms of robot chassis model construction, it can usually be categorized into kinetic model and kinematic model construction [21]. In path planning, kinematic model construction is widely used in mobile robot path planning because it is efficient and practical [22], unless robots involving special loads or structures need to consider kinetic models [23]. Traditional kinematic model construction assumes that the steering and drive mechanisms of the vehicle are rigid bodies and uses an integer order approach for model construction [24]. This approach simplifies the modelling and computational process but also poses the problem that once the structure and parameters of the robot chassis have been determined, the angle constraints are fixed. However, for the kinematic modelling of a CLMR, the traditional approach is not applicable because CLMRs are usually equipped with shock-absorbing suspensions on the drive and steering mechanisms to enhance passability and stability [7]. This leads to changes in the chassis structure when steering or crossing obstacles, which makes the angle constraints in the kinematic model time-varying. Only by more accurately describing the time-varying angle constraints can the CLMR's ability to move in a narrow space be improved. The accurate description of time-varying systems using fractional-order methods offers a valuable opportunity to enhance the CLMR's maneuverability in narrow spaces. In this regard, this paper aims to make a significant contribution by incorporating dynamic factors, such as chassis suspension, into the precise construction of the fractional-order kinematic model.

There are many different path planning methods available [25], mainly including graph-search-based methods (e.g., A\* algorithm [26]), stochastic path planning methods (e.g., Rapidly-Exploring Random Tree, RRT [27]), and optimization algorithms (e.g., Ant Colony Optimization, ACO [28], and Genetic Algorithm, GA [29]). The above path planning methods are usually used for global planning, but for dynamic obstacles, they are combined with local planning in practical applications, such as dynamic window approaches [30]. Graph-search-based methods utilize a heuristic function to assess the priority of nodes within a graph, enabling the identification of an optimal path by traversing the nodes. This approach is known for its high search efficiency and accuracy, making it particularly suitable for small-scale path planning problems. Besides, stochastic path planning methods employ random sampling and tree expansion techniques to swiftly explore feasible paths and gradually approach the desired goal position. These methods excel in high-dimensional environments and complex terrains but are susceptible to planning failures in narrow scenarios. Alternatively, optimization algorithms iteratively search for either the global

optimal solution or a near-optimal solution through an optimization process [31]. These algorithms aim to find the most optimized path by iteratively refining the solution. With the advantages of a global search ability, complex scene adaptability and learning ability, ACO has a strong solving ability in path planning problems and is widely used in real-world scenarios [10].

A lot of good research has been done to use ACOs in a better way, and usually, their efficiency and smoothing are the focus [32]. The pheromone concentration settings and heuristic mechanisms of ACO are the classical means around the efficiency improvement aspect. Liu et al. propose an enhanced heuristic mechanism for Ant Colony Optimization (ACO) that incorporates adaptive pheromone concentration settings and a heuristic mechanism with directional judgments, which increases the purposefulness of planned paths and reduces turn times [33]. However, ACO usually realizes real-time planning in a small search space, and its experimental scene is generally less than a  $50 \times 50$  grid map for algorithm verification [34], which still falls short of the demand for fine path planning in actual large-scale application scenarios. Path smoothing techniques commonly involve incorporating angle or path curvature constraints into the planning method and utilizing spline interpolation to refine the path. For instance, Ali et al. introduce a Markov decision process trajectory evaluation model that considers arc-length parameterization. This model effectively filters and reduces the sharpness of global paths, thereby enhancing path smoothness [35]. Tight constraints on steering angle or path curvature for the sake of smoothing can limit the robot's ability to move, especially in narrow spaces. Feng et al. put forward a path planning algorithm based on immune ACO and B-spline interpolation, which introduces a B-spline curve smoothing strategy based on the optimal solution to make the obtained path shorter and smoother [36]. Nonetheless, in narrow environments, the paths derived using spline interpolation are not necessarily usable, and they may collide with obstacles. In light of large-scale and narrow environments, further investigation of existing ACO algorithms is warranted. To address this, the integration of fractional-order models in path planning holds promise due to their advantages, including flexible and accurate parameter optimization as well as faster convergence. This paper aims to leverage fractional-order models to enhance path planning efficiency and smoothness, which represents a key highlight of the research.

Overall, the path planning performance of CLMRs in large-scale and narrow environments is still limited by inaccurate kinematic models as well as inefficient, insecure, and unsmooth planning methods. To tackle the aforementioned challenges, this paper presents fractional-order enhanced path planning for CLMRs in narrow and large-scale environments, which combines the benefits of fractional-order modelling and optimization techniques to enhance both the kinematic modelling of CLMRs and the ACO algorithm, thereby improving the efficiency of path planning and achieving smoother paths compared to traditional integer-order-based methods. The key contributions of this paper can be summarized as follows:

- (1) To enhance the accuracy of kinematic model construction for CLMRs equipped with suspension systems, an innovative fractional-order-based kinematic modelling method is proposed. This method takes into account the dynamic adjustment of angle constraints to address the issue caused by the time-varying position of the steering wheel's virtual center due to suspension changes. By considering these constraints, the proposed method improves the kinematic capabilities of CLMRs, especially in limit steering states, which lays a solid foundation for subsequent efficient and smooth path planning.
- (2) To address the issue of unsmooth and inefficient planning paths in narrow and large-scale scenes, an improved Ant Colony Optimization (ACO) based path planning method that incorporates fractional-order models is presented, which overcomes the limitations of traditional approaches by establishing a global multifactorial heuristic function, utilizing dynamic angle constraints in fractional-order-based kinematic modelling, incorporating adaptive pheromone adjustment rules, and adopting fractional-

order descriptive state-transfer models. These enhancements enable the algorithm to quickly acquire smooth paths and mitigate the problem of the algorithm getting trapped in local optima in narrow spaces, ultimately enhancing the searching speed and success rate of the algorithm in large-scale scenes.

- (3) Several experiments are conducted in narrow and large-size sceneries, and the effectiveness of the proposed path planning method is proved by comparison with advanced path planning methods.

The rest of this paper is organized as follows. In Section 2, system modelling and problem formulation are described. Section 3 gives the accurate fractional-order-based kinematic modeling of a CLMR. Then, improved ACO-based path planning using fractional-order models is introduced in Section 4. Experimental results are provided in Section 5, followed by the conclusions and future outlook in Section 6.

## 2. System Modelling and Problem Formulation

### 2.1. System Modelling

Constructing accurate kinematic models is essential as a prerequisite for effective path planning. However, to improve the passability of CLMRs, it is insufficient to treat the CLMR as a simple rigid structure. This is because the kinematic constraints imposed on CLMRs during their movement can vary significantly depending on the specific structure of its wheel system. As shown in Figure 1a,b, for CLMRs, limiting the minimum radius of curvature has now become a mainstream method of constructing kinematic constraints, and traditional kinematic models that do not consider suspension can be expressed in the following form:

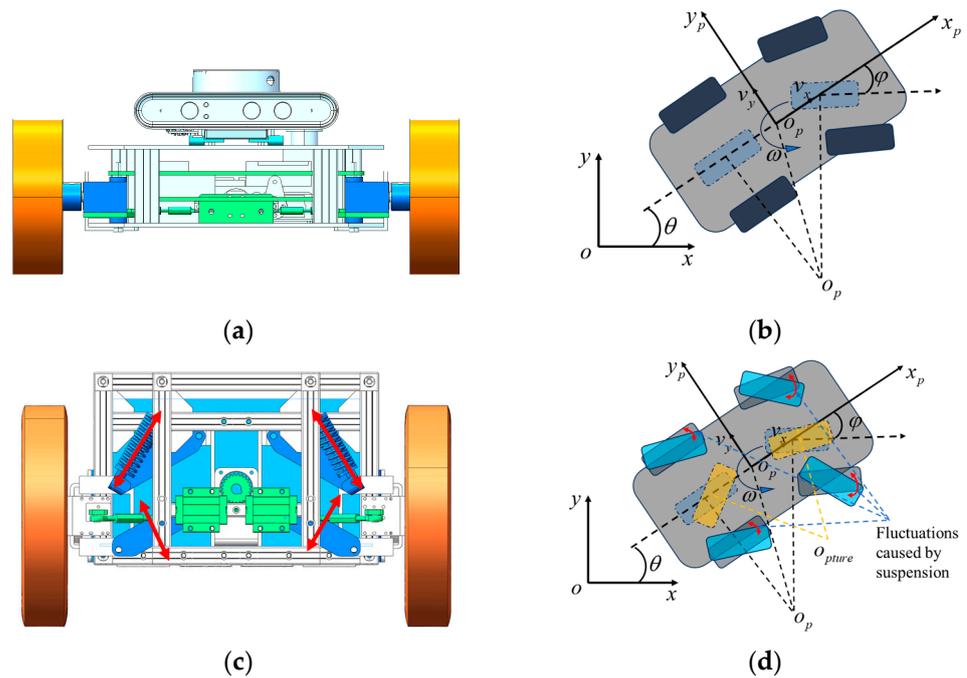
$$v_x^2 + v_y^2 - \rho_{\max} \omega^2 \geq 0 \quad (1)$$

$$v_x \sin \theta - v_y \cos \theta = 0 \quad (2)$$

$$\frac{1}{\rho_{\max}} = \frac{1}{l} \tan(\varphi_{\max}) \quad (3)$$

where  $v_x$  and  $v_y$  are the velocity components in the direction of the  $x$ - and  $y$ -axis in the global coordinate system, respectively,  $\omega$  is the angle velocity of the steering of the mobile robot,  $\rho_{\max}$  refers to the maximum curvature of the running path of the mobile robot,  $l$  is the axis distance of the robot,  $\theta$  denotes the angle of the mobile robot in the global coordinate system, and  $\varphi_{\max}$  is the maximum steering angle of the virtual wheel system.

However, the condition for the Equations (1)–(3) to hold is that the center of rotation of the kinematic is on the extension of the rear wheels. As depicted in Figure 1c,d, the four wheels of CLMRs are usually designed in independent suspension mode to ensure the abilities of obstacle crossing and shock absorption. As a result, the center of the circle of the turn is usually not on the extension line of the rear wheels, in which case the maximum steering angle and the maximum curvature are variable quantities, and the constraints of the robot need to be recalculated. Considering the one-to-one mapping relationship between the robot's direction angle and the path taken, the feasible path needs to take into account the robot's kinematic constraints. Therefore, in this paper, we will use the fractional-order technique to construct a more accurate kinematic model for a CLMR with suspension, which will be introduced in detail in Section 3.



**Figure 1.** The kinematic model of CLMRs. (a) CLMR that does not consider suspension; (b) Traditional kinematic model that does not consider suspension; (c) CLMR that considers suspension; (d) Actual kinematic model that consider suspension.

2.2. Fractional-Order Modelling

Fractional-order calculus, with its rich mathematical properties and characteristics, is an important tool for studying and analyzing complex systems. Considering that fractional order has obvious advantages in processing and modelling real data in nonlinear systems, it can be used in constructing complex kinematic models in path planning, and local characteristic constraints more accurately, and thus close to the real situation. The commonly used fractional-order definitions are the Grunwald–Letnikov definition, Riemann–Liouville definition, and Caputo definition [17–20]. Among them, the Grunwald–Letnikov definition provides an expression for the  $\alpha$  – th derivative, which allows for the consideration of the so-called short-memory principle. The Grunwald–Letnikov fractional derivative is based on discrete data points, which transform the continuity of a function into a discrete differential form. Therefore, Grunwald–Letnikov fractional derivatives apply to discrete data. This applies to the description of discrete path points in this article. Specifically, in defining the fractional-order factor  $\alpha > 0$  and continuous functions  $f(t)$ , we have the following:

$$D^\alpha [f(t)] = \lim_{h \rightarrow 0} \frac{1}{h^\alpha} \sum_{n=0}^{\infty} (-1)^n \binom{\alpha}{n} f(t - nh) \tag{4}$$

where

$$\binom{\alpha}{n} = \frac{\Gamma(\alpha + 1)}{\Gamma(n + 1)\Gamma(\alpha - n + 1)} = \frac{\alpha(\alpha - 1)(\alpha - 2) \dots (\alpha - n + 1)}{n!} \tag{5}$$

where  $D^\alpha(\cdot)$  denotes the GL fractional derivative of order  $\alpha$ ,  $\Gamma(\cdot)$  is the Gamma function,  $h$  is the time step, and  $(\alpha, n)^T$  represents binomial coefficient.

3. Accurate Fractional-Order-Based Kinematic Modeling of CLMR

As illustrated in Figure 1c,d, the CLMRs can dampen the vibration and improve the ability to cross the ditch by installing the suspension, which also leads to the unpredictability of the steering angle during the cornering process. Based on the parameters of the damping and hydraulic cylinders, the current steering angle constraints of the robot can be obtained, which provides the kinematic constraints for path acquisition. The path acquired in this

way can satisfy the obstacle avoidance while improving the tracking accuracy. To simplify the calculation process, the steering wheel can be defined as a freewheel, i.e., the wheel can rotate freely around the axle. In the calculation process, a uniform local coordinate system is defined  $x_p o_p y_p$ , with the robot center point as the origin of the local coordinate system and the direction perpendicular to the front of the vehicle as the  $x$ -axis. Then, the velocity of the virtual wheel in the global coordinate system is defined as:

$$\begin{bmatrix} v_{xp} \\ v_{yp} \end{bmatrix} = \begin{bmatrix} (-x_m \sin \theta - y_m \cos \theta)\omega + v_{oxg} \\ (x_m \cos \theta - y_m \sin \theta)\omega + v_{oyg} \end{bmatrix} \quad (6)$$

where,  $v_{xp}$  and  $v_{yp}$  are the velocities of the virtual wheel in the global coordinate system in the  $x$  and  $y$  directions, respectively,  $x_m$  and  $y_m$  denote the coordinates of the virtual wheel in the local coordinate system in the  $x$  and  $y$  directions, respectively, and  $v_{oxg}$  and  $v_{oyg}$  are the velocity of the origin of the local coordinate system in the global coordinate system. Further, the acceleration expression can be obtained as:

$$\begin{bmatrix} \dot{v}_{xp} \\ \dot{v}_{yp} \end{bmatrix} = \begin{bmatrix} (-x_m \cos \theta + y_m \sin \theta)\omega^2 + (-x_m \sin \theta - y_m \cos \theta)\dot{\omega} + \dot{v}_{oxg} \\ (-x_m \sin \theta - y_m \cos \theta)\omega^2 + (x_m \cos \theta - y_m \sin \theta)\dot{\omega} + \dot{v}_{oyg} \end{bmatrix} \quad (7)$$

According to [37], it can be known that changes in the steering angle of the wheel system can cause dynamic torque distribution. In non-rigid suspension structures, torque fluctuation can cause wheel system displacement. From the torque distribution law, the deformation of the suspension near the inner side of the arc is greater than that on the outer side of the arc, resulting in a change of angle constraint. Fortunately, onboard sensors can accurately capture the current state information during the CLMR's movement, allowing real-time constraint information to be calculated. Therefore, the variation of the virtual wheel direction angle  $\phi_c$  for the CLMR's movement is calculated as:

$$\begin{aligned} \phi_c &= \phi_m - \theta \\ \phi_m &= \arctan\left(\frac{x_{cm}}{y_{cm}}\right) \end{aligned} \quad (8)$$

Assuming that the posterior axis is fixed and parallel to the  $y$ -axis of the defined local coordinate system, there is no change in the point of the posterior axis. Consider that the velocity relation can be represented as:

$$v^2 = v_{xp}^2 + v_{yp}^2 \quad (9)$$

therefore, it can be concluded that:

$$x_v \omega = v_{xp} \sin \theta - v_{yp} \cos \theta \quad (10)$$

This leads to a general equation for the relationship between the CLMR's attitude angle, velocity, and position, and a general constraint equation for the first-order derivatives. The relationship between the effects of velocity, attitude, and steering angle on path planning should be further clarified considering that the robot moves along a curve at different velocities. The running path (the planned path is obtained in the following section) is defined as:

$$y = f(x) \quad (11)$$

Next, the slopes at the virtual wheels are calculated and the offset of the wheel system is taken into account. Conventional equations of kinematics do not correctly express the correctness of the system's kinematic process, and fractional-order models offer the

possibility of accurate system modelling. Therefore, we derive the trajectory equations and bring them into the above equation to obtain the following:

$$D^\rho \theta = \frac{1}{x_v} (\sin \theta - f'(x)) D^\alpha(x) \quad (12)$$

where  $\alpha$  and  $\rho$  are predefined fractional-order operators.

The steering angle of the CLMR imposes a constraint on the maximum curvature of the planned path, considering the dynamic characteristics of the CLMR's kinematics. However, directly calculating the curvature constraints proves challenging. From Equation (8),  $\phi_m$  can be obtained from onboard sensors. Therefore, calculating the real-time rate of  $\theta$  becomes the key to the solution. By combining Equations (11) and (12) we have the following:

$$\frac{D^\rho \theta}{D^\alpha(x)} = \frac{(\sin \theta - D^{1-\alpha}(x) f'(x) \cos \theta)}{x} \quad (13)$$

and bringing Equation (7) into Equation (8), the corner constraint can be obtained as:

$$\varphi_{max} = \arctan \left( \frac{(x_m \cos \theta - y_m \sin \theta) \frac{(\sin \theta - D^{1-\alpha}(x) f'(x) \cos \theta)}{x} + f'(x)}{(-x_m \sin \theta - y_m \cos \theta) \frac{(\sin \theta - D^{1-\alpha}(x) f'(x) \cos \theta)}{x} + 1} \right) - \theta \quad (14)$$

Considering the fluctuation of suspension in different environments, fractional-order-based kinematic modelling provides precise and dynamic angle constraints. This improves the success rate of path planning for a CLMR in narrow and difficult-to-pass scenarios.

From Figure 1, the adjustment of the angular constraints mainly relies on the suspension adjustment of the wheel system in two degrees of freedom. However, through the change of the wheel system structure, the maximum constraint angle is also changed. At this point, the circular extension of the steering is not on the rear wheel system, which is of greater relevance to the planning of the path considering the mapping of the direction angle to the path. From Equation (8), it can be seen that the wheel system angle constraint varies with the change of the wheel system angle of rotation and the initial calibration position. For computational convenience, this paper focuses on the summation constraints of the virtual wheel system to improve computational and planning efficiency. With the calculation of the maximum constraint angle and the acquisition of the current steering angle from the sensing module, we can calculate the change in the maximum constraint angle.

#### 4. Improved ACO Based Path Planning Using Fractional-Order Model

The traditional ACO usually uses the path length as the heuristic function term when solving the path planning problem; however, the environment faced during robot operation is more complex. Path planning, as a key module of mobile robot operation, plays a vital role in the safety and smoothness of robot operation. The pseudocode of the proposed ACO method is shown in Algorithm 1. In the algorithm, lines 1 to 3 are the initialization phase of the algorithm, which completes the initialization of the weight factors and pheromones. Lines 4 to 20 are the iterative part of the algorithm. Specifically, line 6 gives the initial position of the ant colony. Lines 8 to 16 are the ant colony search under the current iteration cycle, and the next moment position of the ant is obtained by transferring the probability model, recording the status of the ant colony, and determining the relationship with the target point. After the completion of the current iteration loop, the pheromone values  $\tau_{ij}(t+1)$  and path optimums  $L_k$  for the scenario are updated. In Line 21, the optimal values for each loop are compared and the optimal path  $L_n$  is selected.

**Algorithm 1** The pseudocode of the improved ACO.

---

```

1  /*Initialization*/
2  Initialize the parameters, including  $\lambda^1$   $\lambda^2$   $\varphi^1$   $L_{tr}$   $\varepsilon$   $\eta$   $\beta_1$   $\beta_2$   $\beta_3$ 
3  Calculate initialize pheromone matrix  $\tau_{ij}(0)$ 
4  /*main Loop*/
5  While iteration number  $n$  does not arrive at the target  $N_{max}$  do:
6    Place all ants at the start point;
7    /*inner loop*/
8    For  $k = 1$  to  $K$  do
9      Calculate the  $p_{ij}^k(t)$  using Formula (25) and confirm the next node
10     If Ant  $k$  reach the target point do
11       Goto step 15
12     Else
13       Goto step 9
14     End if
15     Select the optimal ant path for this round according to Equation (15)
16   End for
17   Update the  $\tau_{ij}(t + 1)$  by Formulas (23)–(25)
18    $n = n + 1$ ,  $k = 0$ 
19   Select the optimal path  $L_n$ 
20 End while
21 Return final optimal path  $L_k$ 

```

---

To obtain a safe and feasible path, the safety, smoothness, and path distance of the path need to be considered comprehensively, so the improved multi-factor heuristic function is as follows:

$$J = \varphi^1 \left\{ \lambda^1 \omega + \lambda^2 \left( D + \frac{1}{d} \right) \right\} \quad (15)$$

where  $J$  is the path planning heuristic function,  $\varphi^1$  refers to the heuristic function that ensures the safe operation of the CLMR,  $\lambda^1$  and  $\lambda^2$  denote the weighting factors, respectively,  $\omega_{ij}(k)$  refers to the curvature smoothing factor,  $D(k)$  is the modified path heuristic function, and  $d(k)$  implies the standard path heuristic function, which is required by the planning method to obtain the minimum value of the cost function.

#### 4.1. Factorization of the Cost Function with Fractional-Order Model

##### 4.1.1. Safety Functions with Local Region Preprocessing

The operational safety of the mobile robot is the first factor to be considered for path planning. As shown in Figure 2, considering the existence of tracking errors, the planning module needs to leave enough redundant space. To facilitate the process, a common approach is to uniformly inflate the static map with the CLMR's radius; however, in large-scale or highly dynamic scenarios, the optimal or relatively optimal paths are difficult to obtain and the length of the planned paths increases dramatically. Treating robots as a fixed matrix reduces the passability of a CLMR and leads to lower search efficiency. For this reason, this paper proposes a safety factor function based on ACO storage information, defined as follows:

$$\varphi_{ij}^1(k) = \begin{cases} 1, L_{tr} \times S(i, j) \cap \text{imdilate}(M_{A \times B}(i, j), L_{tr} \times S(i, j)) \\ \cap L_{pix} \times \text{mod}(\text{Dir}(i, j), 2) == 0 \neq \text{Inf}; \\ \text{Inf}, \text{Others} \end{cases} \quad (16)$$

where

$$S(i, j) = R_{C \times D} \times f(\theta) \quad (17)$$

where  $L_{tr}$  is the safety threshold constant,  $S(i, j)$  denotes the intermediate function,  $\text{imdilate}(\cdot)$  refers to the map expansion function,  $M_{A \times B}(i, j)$  implies the map information stored by the ACO in the map information,  $A \times B$  is expressed as the information dimension matrix,  $L_{pix}$

denotes the length of the map pixel point,  $mod(\cdot)$  refers to the residual function,  $Dir(i, j)$  is the searching direction raster labelling,  $R_{C \times D}$  stands for the CLMR's matrix under the CLMR's coordinate system,  $C \times D$  denotes the robot's matrix dimensions,  $f(\theta)$  is the coordinate system transfer matrix, and  $\theta$  refers to the robot direction angle in global coordinates.

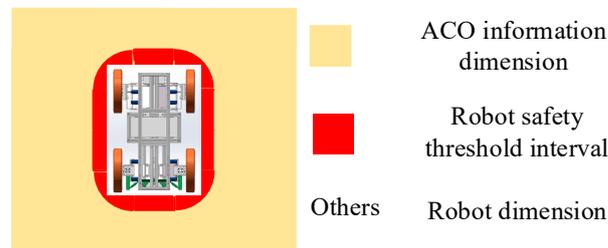


Figure 2. Planning Safety Thresholds of the CLMR.

#### 4.1.2. Smoothing Function Based on Dynamic Angle Constraints

The ACO iterates towards the final heuristic function during the planning process, while the smoothness and feasibility of the paths are not given much attention. However, the angle constraints of the robot impose new requirements on the planning of paths, while excessive corners reduce the feasibility of paths. Improving the smoothness of the path and eliminating excessive corners will help reduce the travelling time and improve the smoothness of the path. To address these issues, considering the dynamic characteristics of the dynamic angle constraints in fractional-order-based kinematic modelling, a dynamic smoothing factor is introduced to reduce the integrated angle probability and improve the comprehensive performance of the algorithm, and the corner smoothing function is:

$$\omega_{ij}(k) = \begin{cases} \varepsilon G(i, j) \frac{\varphi_{ij}(N_{\max} - N_k)}{\varphi_c N_k} & \varphi_{ij} \leq \varphi_c \\ Inf & \varphi_{ij} > \varphi_c \end{cases} \quad (18)$$

where  $\varphi_c$  denotes the computed wheel system corner constraint,  $\varphi_{ij}$  is the planning corner at point  $i$  to point  $j$ ,  $\varepsilon$  denotes the path angle adjustment factor,  $G(i, j)$  represents the robot straight travelling function,  $N_{\max}$  stands for the maximum number of iterations, and  $N_k$  refers to the current number of iterations. Further, as shown in Figure 3, the robot straight line function is expressed as:

$$G(i, j) = \begin{cases} \frac{\varphi_{m-1}(l_{m-2} + l_{m-1}) + \varphi_m(l_{m-1} + l_m) + \varphi_{m+1}(l_m + l_{m+1})}{2(l_{m-2} + l_{m-1} + l_m + l_{m+1})} & m \geq 4 \\ 1 & m < 4 \end{cases} \quad (19)$$

where  $l_{m-2}, l_{m-1}, l_m$  and  $l_{m+1}$  are the four consecutive trajectories planned by the colony at the current point, and  $\varphi_{m-1}, \varphi_m$  and  $\varphi_{m+1}$  represent the three consecutive corners consisting of these four trajectories.

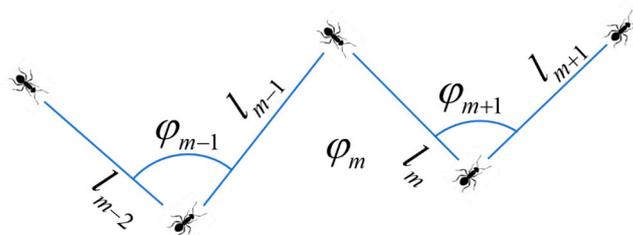


Figure 3. Planning Straight-line Constraints of the CLMR.

The feasibility of the path is improved by the smoothing function with angle constraints. In the function, the smoothing factor of the angle is added to ensure the smoothness of the planned path, which is more favorable to the operation of the CLMR.

#### 4.1.3. Path Functions by Adding Adjusting Factor

In actual operation, the length of the path is still an important factor to be considered, which is closely related to the CLMR's work efficiency and energy utilization. For the traditional ACO, at the beginning of the iteration, the very small distance difference easily causes search confusion. At the late stage of convergence, there is a certain probability of falling into a local minimum. For this reason, we need to amplify the very small factor of fluctuation in the early stage in the distance factor to accelerate the convergence speed. In the later stages of iteration, we need to reduce the influence brought by the path, so that the path obtained is comprehensively optimal. The modified path factor function is:

$$D_{ij}(k) = \begin{cases} \frac{\eta(L(\max(P_{ij}, P_{Goal})) - d(P_{ij}, P_{Goal}))}{L_{pix} + L(\max(P_{ij}, P_{Goal})) - L(\min(P_{ij}, P_{Goal}))} \frac{(N_{\max} - N_k)}{N_k} & (20) \end{cases}$$

where  $\eta$  is the path coefficient,  $L(\max(P_{ij}, P_{Goal}))$  denotes the longest path from the current point  $P_{ij}$  to the target point  $P_{Goal}$  planned by the ACO,  $L(\min(P_{ij}, P_{Goal}))$  denotes the shortest path from the current point  $P_{ij}$  to the target point  $P_{Goal}$ , and  $d(P_{ij}, P_{Goal})$  refers to the Euclidean distance from the current point  $P_{ij}$  to the target point  $P_{Goal}$ .

#### 4.2. Adaptive Pheromone Update Rules

Traditional ACO algorithms are usually set to a constant  $C$  in the initial stage, which leads to a blind search mainly relying on the heuristic function at the initial stage, and it is very easily falls into a local minimum in large scene maps. To solve this problem and improve the search efficiency, the initial pheromone is redistributed in the initial stage of the map with the help of the convergence method of the initial A\* algorithm to speed up the subsequent path replanning in large scenes. The initial pheromone is recorded as:

$$\tau_{ij}(0) = \begin{cases} nc, j \in l_p \\ c, j \in otherwise \end{cases} \quad (21)$$

In the actual operation of the CLMR, path planning is influenced by multiple factors, and the goal is to find an optimal path that considers all of these conditions collectively. Currently, efforts are focused on improving the amount of pheromone changes at different points along the path. The specific follow-up rules are as follows:

$$\tau_{ij}(t+1) = (1 - \zeta(t))\tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (22)$$

$$\Delta\tau_{ij}(t) = \begin{cases} \frac{\kappa_1 Ph}{W} + \frac{\kappa_2 Ph}{L_A}, (i, j) \in allowed \\ 0, others \end{cases} \quad (23)$$

where  $\zeta(t)$  is the dynamic volatilization factor of pheromone,  $\tau_{ij}(t+1)$  denotes the pheromone matrix at the current moment,  $Ph$  represents the pheromone concentration,  $\kappa_1$  and  $\kappa_2$  refer to the conditioning factors,  $W$  implies the mean squared deviation value of the walking angle, and  $L_A$  denotes the cumulative path length from the starting point to the target point.

The iterative values of  $\Delta\tau_{ij}(t)$  are also dynamically adjusted through the changes of angle  $W$  and distance values  $L_A$ . By setting the magnitude of the values of weight coefficients  $\kappa_1$  and  $\kappa_2$ , the acquisition of effective paths that are more compatible with the scene is facilitated.

The dynamic pheromone volatilization factor is designed as:

$$\zeta(t) = \begin{cases} a \frac{N_{\max}}{N_k} \zeta(t-1), t \neq 0, (0 < a < 1) \\ \zeta_{init}, t = 0 \end{cases} \quad (24)$$

where  $a$  and  $\zeta_{init}$  are self-defined constants, and  $\zeta(t)$  can be adjusted adaptively with the search.

From Equation (24), it can be seen that the pheromone volatilization function  $\zeta(t)$  reaches the maximum value in the pre-search period of the ACO which is in the period of the fastest change of the pheromone volatilization value. This increases the uncertainty factor in the early stage of the algorithm during the optimization process, which is more conducive to obtaining the globally optimal feasible solution. As the number of iterations increases, the pheromone volatility function  $\zeta(t)$  tends to stabilize, and the local search process is more frequent, which helps to improve the quality of the path. Therefore, the iterative process accomplishes the adaptive regulation of pheromone concentration, which facilitates the realization of rapid path planning and optimization.

#### 4.3. Fractional-Order Transfer Probability Rules

To obtain the feasible path faster and ensure the quality of the path, this paper improves the transfer probability of the algorithm. It makes the target probability increase the angle factor and distance factor. It is expected to obtain the shortest path under the premise of ensuring a smooth path. The improved state transfer probability is:

$$p_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^{\beta_1} [D^{\alpha_1} R_{ij}(t)]^{\beta_2} [D^{\alpha_2} Q_{ij}(t)]^{\beta_3}}{\sum_{s \in allowed} [\tau_{is}(t)]^{\beta_1} [D^{\alpha_1} R_{is}(t)]^{\beta_2} [D^{\alpha_2} Q_{is}(t)]^{\beta_3}}, & (25) \\ 0 & \end{cases}$$

where  $\beta_1$ ,  $\beta_2$ , and  $\beta_3$  denote the heuristic term factor, respectively, and  $R_{ij}(t)$  and  $Q_{ij}(t)$  are defined as follows:

$$D^{\alpha_1} R_{ij}(t) = D^{\alpha_1} \omega_{ij}(t) \quad (26)$$

$$D^{\alpha_2} Q_{ij}(t) = D^{\alpha_1} (D_{ij}(t) + d_{ij}(t)) \quad (27)$$

The fractional reciprocal of the angle factor and the distance factor is calculated to improve the sensitivity of the transition probability to its change, to ensure the timeliness of the path change and to improve the passability of the path.

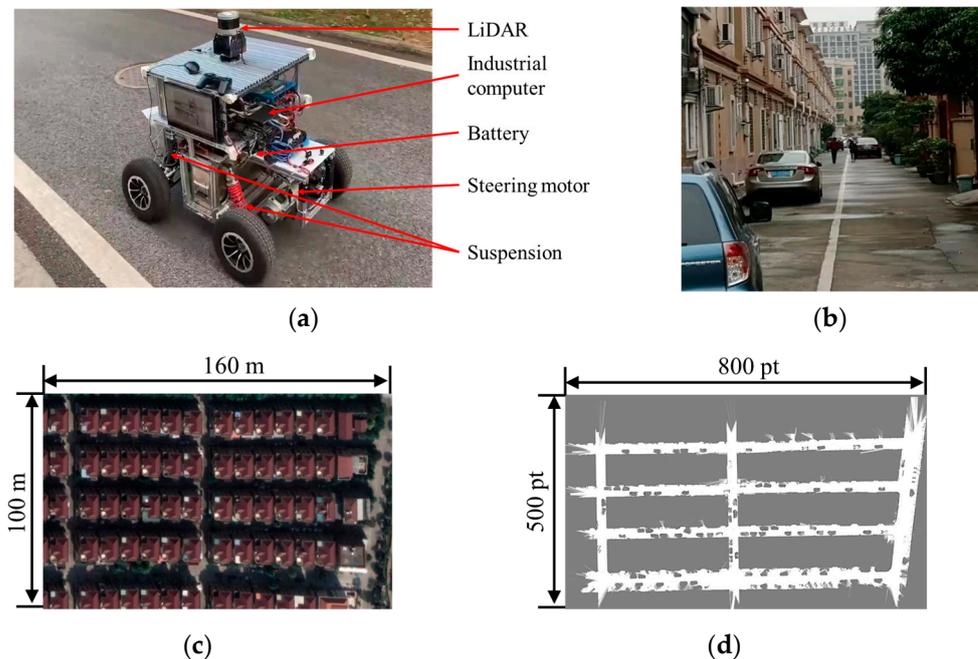
A fractional-order state-transfer model can more accurately adjust the exploration probability of ant colonies in unexplored areas, which is beneficial for ant colony algorithms to jump out of the current local optimal solution and search for the global optimal solution with a greater probability, thereby improving the success rate of the search in large-scale scenarios. At the same time, due to the high dependence of the pheromone concentration on the optimal path, the modification of the transfer model increases the search breadth and the search probability of the optimal path, avoiding the acquisition of the optimal path, accelerating the search process around the optimal path, and thus improving the convergence speed.

## 5. Experimental Validations

### 5.1. Experimental Implementation

The narrow and large-size experimental scenes and self-developed CLMRs are shown in Figure 4. The CLMR consists of an industrial computer (Intel(R) Core (TM) i7-6500U CPU @2.50 GHz, 8 GB of RAM, 64-bit operating system), LiDARs, motor encoders, and some related sensors, such as an ultrasonic transducer and IMU. More specifically, with an impressive range of 150 m and a scanning rate of 10 Hz, the Velodyne VLP-16 LiDAR provides the CLMR with a broad field of view, which guarantees that the CLMR has enough field of view to ensure safety and real-time mapping and path planning. As illustrated in Figure 4a, to improve the stability and passability of the CLMRs in complex environments, the drive and steering mechanisms are fitted with suspensions. As we can see from Figure 4b–d, the entire neighborhood is quite expansive, covering an area of sixteen thousand square meters. However, the alleyways within the neighborhood are remarkably narrow. These tight spaces are often filled with temporarily parked cars and bustling pedestrians, which severely limits the available space for CLMRs to navigate through. In particular, shown in Figure 4b, to obtain more detailed and practical paths, we

chose a grid map with a resolution of 0.2 m, making the map size  $800 \times 500$ , which is a great challenge for the path planning algorithm.



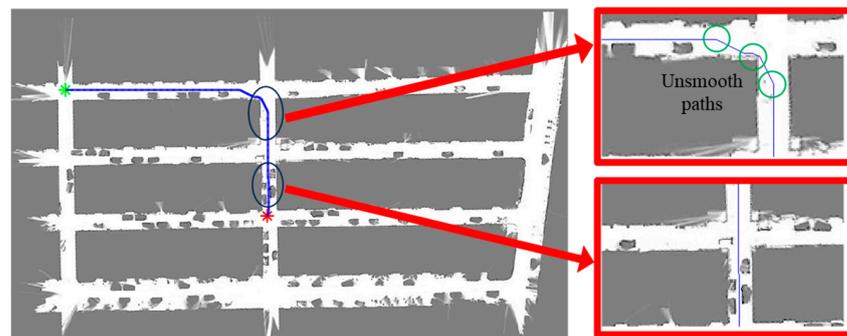
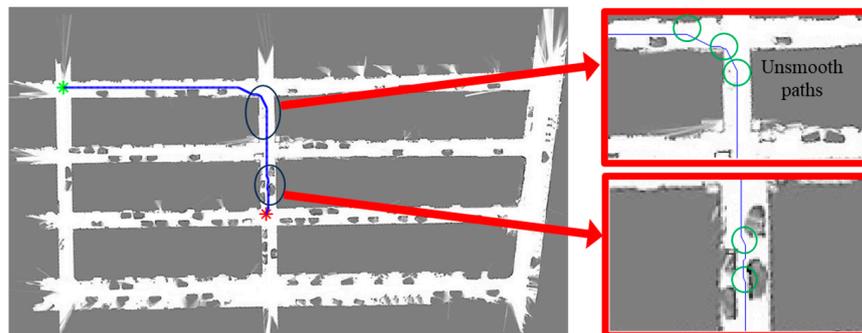
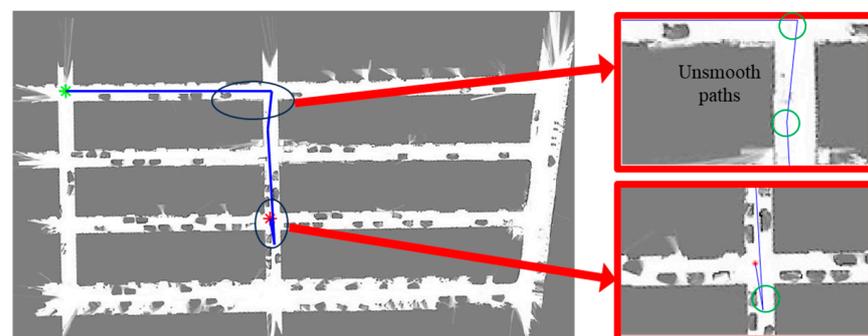
**Figure 4.** Experimental scene and platform. (a) Self-developed CLMR; (b) Real-world scene; (c) Google map of the narrow and large-size scene; (d) Grid map of the narrow and large-size scene.

### 5.2. Experimental Results and Discussions

In the experiment, it is necessary to obtain real-time vehicle positioning data and wheel steering angle data. Real-time recording and storage of experimental data was performed on the PC. The planning, calculation, allocation, and execution process are as follows: The current data are processed by the data processing unit and returned to MATLAB. Then, MATLAB is used to complete the calculation of the planning algorithm. Finally, the path instructions generated by the planning are sent to the control unit, completing the current path planning process. The initial state of the considered robot is the same, all parameters are optimally adjusted, and experiments are conducted under the same operating conditions. In the process of parameter tuning, parameter stabilization is achieved through the use of a nature-inspired optimization algorithm called Artificial Bee Colony [38], which reduces the sensitivity of the parameters to the environment and also ensures the fairness of the algorithm comparison process. The values of the parameters obtained are as follows:  $\lambda^1 = 0.6$ ,  $\lambda^2 = 0.2$ ,  $\varepsilon = 0.1$ ,  $N_{\max} = 50$ ,  $\eta = 1.1$ ,  $\kappa_1 = 0.8$ ,  $\kappa_2 = 0.2$ , and  $a = 0.4$ . The superiority of the proposed fractional-order ACO (FACO) is verified by comparing it with the traditional A\*, improved A\* (IA) [26], ACO [28], improved ACO combined with path fitting (ACOF) [39], Genetic Algorithm (GA) [29], and the GA method combined with A\* (AGA). We used different algorithms to run each of them ten times in large-scale narrow scenes. To better validate the effectiveness and advantages of the proposed method, we drew on the comparative methods in literature [40] and selected common path lengths, times, and success rates of trajectory planning (including planning failures and collisions with obstacles) to accurately describe the process of path planning. The comparison results are shown in Table 1, and the experimental results of the planned path are shown in Figures 5–11.

**Table 1.** Performance comparison of different path planning methods.

Methods	Path Length (m)	Times (s)	Success Rate (%)
A*	91.60	857.26	100
IA	93.57	591.73	100
GA	101.24	100.61	100
AGA	93.36	123.67	100
ACO	89.20	350.66	100
ACOF	94.97	153.71	0
FACO	92.98	63.74	100

**Figure 5.** Path planning result of A\*.**Figure 6.** Path planning result of IA.**Figure 7.** Path planning result of GA.

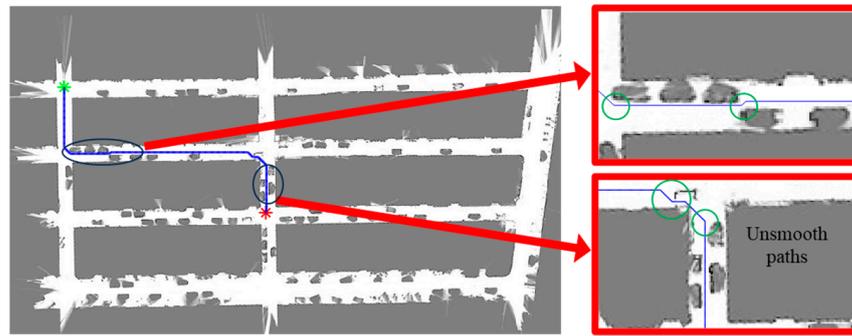


Figure 8. Path planning result of AGA.

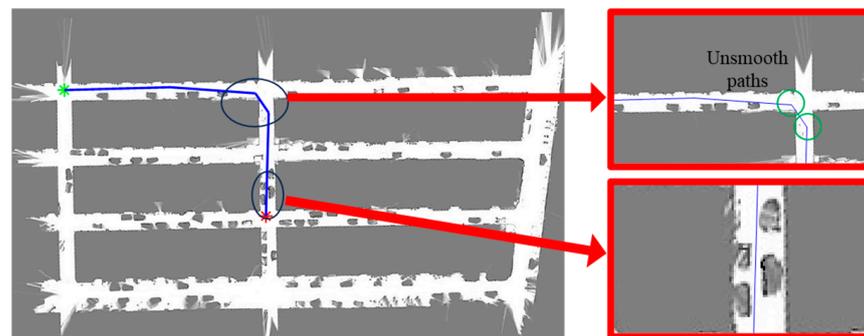


Figure 9. Path planning result of ACO.

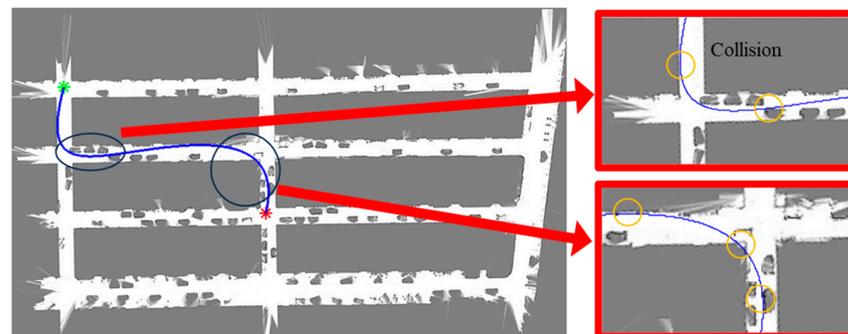


Figure 10. Path planning result of ACOF.

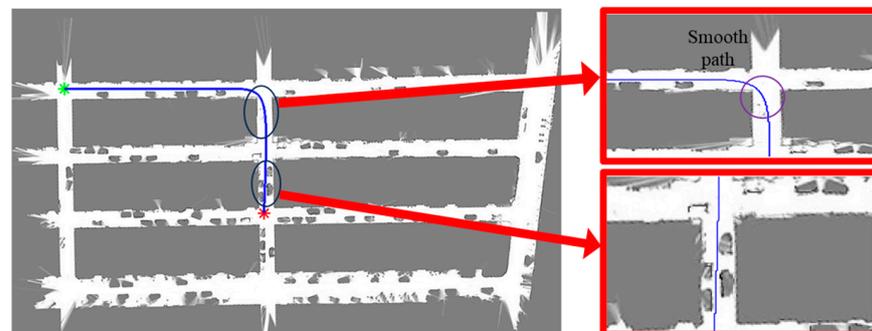


Figure 11. Path planning result of FACO.

From Table 1, it can be seen that the A\* algorithm takes the longest time, which is because the A\* algorithm completes the traversal of the surrounding space before finding the target point, and due to the high-precision attribute of the map, more and more computational resources are consumed and the computational speed decreases dramatically in the

planning process. In the improved A\* method, the search direction is guided to improve the planning efficiency, but to ensure the success rate of path planning, the efficiency improvement for a large-space search is not obvious. GA has excellent performance in the field of optimization, and in the planning process GA achieves random sampling in the space by repeated cross-compilation, which greatly reduces the planning time, and by combining it with the A\* algorithm, there is an increase in the planning time, but there is an increase in the stability of the path. The ACO-based planning algorithm demonstrates reduced dependency on parameters in path planning. However, employing pure ACO preprocessing alone increases the time required. By combining the ACO algorithm with A\*, the overall time decreases further. In the proposed method, the additional search burden caused by the large space is mitigated through local space weighting in the ACO search. This reduction in computational burden leads to improved planning efficiency. Experimental results show a significant enhancement in efficiency with the proposed method, achieving improvements of 92.56%, 57.84%, and 81.82% compared to traditional A\*, GA, and ACO methods, respectively. These improvements have great significance for real-world scenarios.

In terms of path length, the A\* method shows a greater advantage due to the objective of the method to obtain shorter paths, and with the increase in constraints, the paths of the improved methods based on A\*, GA, and ACO all increase to varying degrees. Further analysis shows that GA shows significant non-randomness of paths due to the random sampling in space and the path length appears to increase to a greater extent, whereas the ACO algorithm shows better stability of paths due to having an advanced spatial search and shorter path lengths. The proposed method needs to meet the actual operational requirements and the dynamic constraints make the planned paths increase, but the length of the planned paths decreases by 0.63%, 0.40%, and 2.10% compared to the improved methods of A\*, GA, and ACO, respectively. When comparing the various path planning, all of them show better results in terms of path length. However, the proposed method is more advantageous under the constraints.

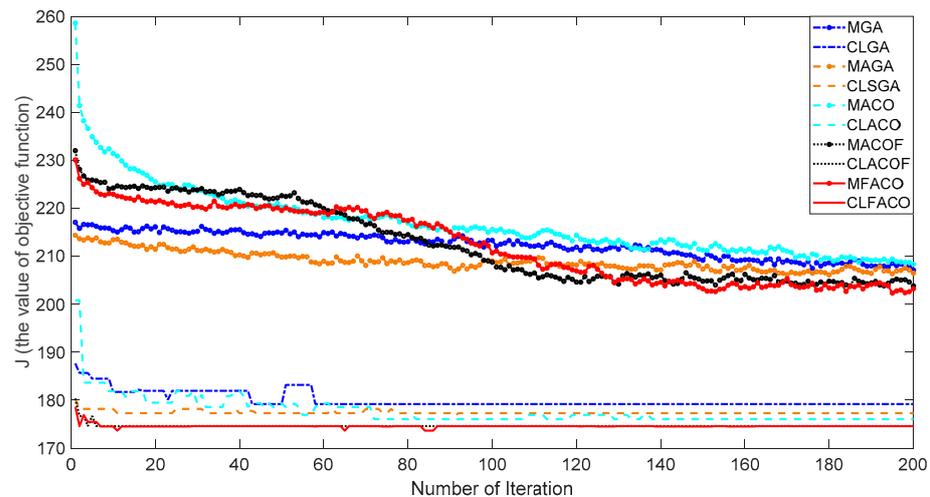
In Table 2, the smoothness indicator is given, which reflects the proportion of different steering angles between path points in path planning. It has been proven that a smoother trajectory is achieved when there is a lower proportion of large steering angles. However, optimization-based planning algorithms, such as GA, AGA, ACO, and ACOF, often prioritize a single performance improvement, resulting in a path consisting of a relatively simple finite set of points. For example, the GA algorithm represents the path using only five coordinate points, and the proportion of large steering angles reaches 66.67%. Consequently, these optimization algorithms introduce numerous angle mutation points, making it challenging to ensure a smooth path. Moreover, traditional path planning methods like A\* and IA struggle to incorporate dynamic angle constraints at corners, leading to scattered abrupt changes in the path. Although smoothing the obtained path can prevent the occurrence of mutation points, it may also result in high spatial requirements after the smoothing process. To address these challenges, we propose a novel path planning algorithm that fully leverages the constraint characteristics of angles. This algorithm performs real-time smoothing during the planning phase, avoiding abrupt changes in the path's angle, the proportion of small steering angles exceeds 99%, which means that the proposed method is essentially free of steering mutations. As a result, it offers significant benefits for planning in narrow spaces while still considering angle constraints.

**Table 2.** The proportion of different steering angles between path points.

Methods	0–5°	5–10°	>10°
A*	98.39%	1.15%	0.46%
IA	96.77%	2.30%	0.93%
GA	0	33.33%	66.67%
AGA	0	0	100%
ACO	33.33%	0	66.67%
ACOF	100%	0	0
FACO	99.4%	0.6%	0

According to the results of path planning from Figures 5–11, it is evident that both the A\* algorithm and its improved version can produce relatively smooth paths. However, it is noticeable that the paths tend to excessively prioritize shorter routes at corner nodes and articulation points. This pursuit of shorter paths increases operational risks and reduces feasibility. In the GA method, random sampling points introduce a certain possibility of encountering large corners, resulting in longer paths, and placing higher demands on the CLMR's maneuverability. To address this issue, the combination of A\* and GA methods improves the path's smoothness and provides guidance. Nonetheless, the presence of more corners and narrow areas in the path poses greater challenges for the robot's capabilities and model. In the planning of ACO, it follows the pursuit of the shortest path when the path shows better smoothness, but the passage is not considered in the method. Given this, after combining with A\*, the smoothing of the path is proposed, and here the B-spline is used for processing because the smoothing of the path is usually accompanied by a change in the path points, and it can be seen from the figure that, due to the space being relatively small, the path status quo is changed, and a larger set of points appeared to be in contact with the obstacles, which poses a greater threat to the operational safety, making it difficult for the traditional path smoothing to be sufficient. In the proposed method, a transfer model of fractional-order is established by building a fractional-order model with constraints in the method, which improves the path smoothing, avoids the path changes brought about by the additional path smoothing, and improves the path safety, while the obtained smoothed paths ensures the path feasibility. At the same time, we can see that the paths of ACOF are corrected to increase smoothness, but in a narrow space, and this late correction makes it easy for the planned paths to run into obstacles, resulting in a drastic decrease in their success rate. The proposed method, on the other hand, maintains smoothness and at the same time has a high success rate, which proves the excellent performance of the planning method based on the fractional-order model.

Figure 12 provides the mean and current lowest values of the objective function for various optimization algorithms: GA—both Mean GA (MGA) and Current Lowest GA (CLGA), AGA—both Mean AGA (MAGA) and Current Lowest AGA (CLAGA), ACO—both Mean ACO (MACO) and Current Lowest ACO (CLACO), ACOF—both Mean ACOF (MACOF) and Current Lowest ACOF (CLACOF), and FACO—both Mean FACO (MFACO) and Current Lowest FACO (CLFACO). GA is characterized by fast convergence in the initial stage; however, with the depth of the iteration, the GA is more likely to fall into premature maturity. In contrast to the ACO calculation, the method is more dependent on the initial value, and the convergence is slower in the early stage; however, with the help of the method's adaptation to nonlinear and complex problems it has an advantage in dealing with the local optimal solution. To improve the search efficiency, the traversal based on the A\* method is used in this paper: AGA, ACOF, and FACO. This shows great advantages in the fast convergence of the mean value and the optimization of the minimum value. In the iterative search for the minimum value, AGA achieves faster convergence, while ACOF and FACO can still jump out of the current local optimality conditions and further search for the global optimal solution. Therefore, the proposed method has a great advantage in realizing the optimal value of the cost function in this paper.



**Figure 12.** Comparison of the objective function value.

## 6. Conclusions and Outlook

This paper presents a novel ACO approach for the path planning of CLMRs with suspension in narrow and large-scale environments, which combines fractional-order enhanced path planning with an improved ACO algorithm to achieve smooth and efficient paths. To improve the accuracy of the kinematic model construction for CLMRs with suspension, a precise fractional-order-based kinematic modeling method is introduced. This method takes into account the dynamic adjustment of angle constraints, resulting in a more accurate representation of the CLMR's motion. Furthermore, the path planning algorithm is further enhanced by incorporating fractional-order transfer-probability modelling into the ACO framework. This extension effectively addresses the challenges associated with local optima and lack of smoothness in narrow spaces. Additionally, it improves the search speed in large-scale scenes, ensuring more efficient and optimized path planning.

It is worth noting that the proposed method adopts a fixed fractional order although it is more accurate and flexible than the integer order, but the exact value of the fractional order is still a challenge. In future work, we will introduce sensor observation, further implement variable fractional order to improve the accuracy of model construction and explore the potential application of fractional-order models in other path planning methods. In addition, we will subsequently refine the path planning methods and further investigate local planning algorithms based on the existing global path planning to cope with highly dynamic scenarios.

**Author Contributions:** Conceptualization, L.L. and L.J. (Liquan Jiang); methodology, L.L. and L.J. (Liquan Jiang); software, L.J. (Liyu Jiang) and L.J. (Liquan Jiang); validation, L.J. (Liquan Jiang); formal analysis, W.T.; investigation, L.L. and W.T.; resources, L.J. (Liquan Jiang); data curation, W.T.; writing—original draft preparation, L.L., R.H. and L.J. (Liquan Jiang); writing—review and editing, L.J. (Liquan Jiang); visualization, L.J. (Liyu Jiang) and R.H.; supervision, L.J. (Liquan Jiang); project administration, L.J. (Liquan Jiang); funding acquisition, L.J. (Liquan Jiang). All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded in part by the Hubei Provincial Engineering Research Center for Intelligent Textile and Fashion (Wuhan Textile University) under Grant no. 2023HBITF03, in part by State Key Laboratory of New Textile Materials and Advanced Processing Technologies no. FZ20230023, in part by Hubei Key Laboratory of Digital Textile Equipment no. DTL2023023 and in part by Fund of National Engineering Research Center for Water Transport Safety under Grant no. A202303.

**Data Availability Statement:** The data presented in this study may be available on request from the corresponding author.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## References

1. Liu, L.; Wang, X.; Yang, X.; Liu, H.; Li, J.; Wang, P. Path planning techniques for mobile robots: Review and prospect. *Expert Syst. Appl.* **2023**, *227*, 120254. [[CrossRef](#)]
2. Meng, J.; Wang, S.; Xie, Y.; Li, G.; Zhang, X.; Jiang, L.; Liu, C. A safe and efficient LIDAR-based navigation system for 4WS4WD mobile manipulators in manufacturing plants. *Meas. Sci. Technol.* **2021**, *32*, 045203. [[CrossRef](#)]
3. Wu, L.; Huang, X.; Cui, J.; Liu, C.; Xiao, W. Modified adaptive ant colony optimization algorithm and its application for solving path planning of mobile robot. *Expert Syst. Appl.* **2023**, *215*, 119410. [[CrossRef](#)]
4. Li, G.; Liu, C.; Wu, L.; Xiao, W. A mixing algorithm of ACO and ABC for solving path planning of mobile robot. *Appl. Soft. Comput.* **2023**, *148*, 110868. [[CrossRef](#)]
5. Ben-Asher, J.Z.; Rimon, E.D. Time optimal trajectories for a car-like mobile robot. *IEEE Trans. Robot.* **2021**, *38*, 421–432. [[CrossRef](#)]
6. Shui, Y.; Zhao, T.; Dian, S.; Hu, Y.; Guo, R.; Li, S. Data-driven generalized predictive control for car-like mobile robots using interval type-2 T-S fuzzy neural network. *Asian J. Control* **2022**, *24*, 1391–1405. [[CrossRef](#)]
7. Theunissen, J.; Tota, A.; Gruber, P.; Dhaens, M.; Sorniotti, A. Preview-based techniques for vehicle suspension control: A state-of-the-art review. *Annu. Rev. Control* **2021**, *51*, 206–235. [[CrossRef](#)]
8. Meng, J.; Wang, S.; Jiang, L.; Hu, Z.; Xie, Y. Accurate and Efficient Self-localization of AGV Relying on Trusted Area Information in Dynamic Industrial Scene. *IEEE Trans. Veh. Technol.* **2023**, *72*, 7148–7159. [[CrossRef](#)]
9. Qie, T.; Wang, W.; Yang, C.; Li, Y.; Liu, W.; Xiang, C. A path planning algorithm for autonomous flying vehicles in cross-country environments with a novel TF-RRT\* method. *Green Energy Intell. Trans.* **2022**, *1*, 100026. [[CrossRef](#)]
10. Miao, C.; Chen, G.; Yan, C.; Wu, Y. Path planning optimization of indoor mobile robot based on adaptive ant colony algorithm. *Comput. Ind. Eng.* **2021**, *156*, 107230. [[CrossRef](#)]
11. Mac, T.T.; Copot, C.; Tran, D.T.; De Keyser, R. Heuristic approaches in robot path planning: A survey. *Robot. Auton. Syst.* **2016**, *86*, 13–28. [[CrossRef](#)]
12. Song, B.; Wang, Z.; Zou, L. An improved PSO algorithm for smooth path planning of mobile robots using continuous high-degree Bezier curve. *Appl. Soft. Comput.* **2021**, *100*, 106960. [[CrossRef](#)]
13. Sang, H.; You, Y.; Sun, X.; Zhou, Y.; Liu, F. The hybrid path planning algorithm based on improved A\* and artificial potential field for unmanned surface vehicle formations. *Ocean Eng.* **2021**, *223*, 108709. [[CrossRef](#)]
14. Nassef, A.M.; Abdelkareem, M.A.; Maghrabie, H.M.; Baroutaji, A. Metaheuristic-Based Algorithms for Optimizing Fractional-Order Controllers—A Recent, Systematic, and Comprehensive Review. *Fractal Fract.* **2023**, *7*, 553. [[CrossRef](#)]
15. Jiang, L.; Wang, S.; Xie, Y.; Xie, S.; Zheng, S.; Meng, J.; Ding, H. Decoupled fractional supertwisting stabilization of interconnected mobile robot under harsh terrain conditions. *IEEE Trans. Ind. Electron.* **2021**, *69*, 8178–8189. [[CrossRef](#)]
16. Jiang, L.; Wang, S.; Xie, Y.; Xie, S.Q.; Zheng, S.; Meng, J. Fractional robust finite time control of four-wheel-steering mobile robots subject to serious time-varying perturbations. *Mech. Mach. Theory* **2022**, *169*, 104634. [[CrossRef](#)]
17. Xie, Y.; Zhang, X.; Meng, W.; Zheng, S.; Jiang, L.; Meng, J.; Wang, S. Coupled fractional-order sliding mode control and obstacle avoidance of a four-wheeled steerable mobile robot. *ISA Trans.* **2021**, *108*, 282–294. [[CrossRef](#)] [[PubMed](#)]
18. Patnaik, S.; Hollkamp, J.P.; Semperlotti, F. Applications of variable-order fractional operators: A review. *Proc. R. Soc. London Ser. A-Math. Phys. Eng. Sci.* **2020**, *476*, 20190498. [[CrossRef](#)]
19. Zhao, C.; Dai, L.; Huang, Y. Fractional-order Sequential Minimal Optimization Classification Method. *Fractal Fract.* **2023**, *7*, 637. [[CrossRef](#)]
20. Muresan, C.I.; Birs, I.; Ionescu, C.; Dulf, E.H.; De Keyser, R. A review of recent developments in autotuning methods for fractional-order controllers. *Fractal Fract.* **2022**, *6*, 37. [[CrossRef](#)]
21. Tzafestas, S.G. Mobile robot control and navigation: A global overview. *J. Intell. Robot. Syst.* **2018**, *91*, 35–58. [[CrossRef](#)]
22. Jiang, H.; Xu, G.; Zeng, W.; Gao, F. Design and kinematic modeling of a passively-actively transformable mobile robot. *Mech. Mach. Theory* **2019**, *142*, 103591. [[CrossRef](#)]
23. Mohammadpour, M.; Kelouwani, S.; Gaudreau, M.A.; Zeghmi, L.; Amamou, A.; Bahmanabadi, H.; Graba, M. Energy-efficient motion planning of an autonomous forklift using deep neural networks and kinetic model. *Expert Syst. Appl.* **2024**, *237*, 121623. [[CrossRef](#)]
24. Song, B.; Wang, Z.; Zou, L.; Xu, L.; Alsaadi, F.E. A new approach to smooth global path planning of mobile robots with kinematic constraints. *Int. J. Mach. Learn. Cybern.* **2019**, *10*, 107–119. [[CrossRef](#)]
25. Rubio, F.; Valero, F.; Llopis-Albert, C. A review of mobile robots: Concepts, methods, theoretical framework, and applications. *Int. J. Adv. Robot. Syst.* **2019**, *16*, 1729881419839596. [[CrossRef](#)]
26. Niu, C.; Li, A.; Huang, X.; Li, W.; Xu, C. Research on Global Dynamic Path Planning Method Based on Improved A\* Algorithm. *Math. Probl. Eng.* **2021**, *2021*, 1–13. [[CrossRef](#)]
27. Wang, J.; Chi, W.; Li, C.; Wang, C.; Meng, M.Q.H. Neural RRT\*: Learning-based optimal path planning. *IEEE Trans. Autom. Sci. Eng.* **2020**, *17*, 1748–1758. [[CrossRef](#)]
28. Lyridis, D.V. An improved ant colony optimization algorithm for unmanned surface vehicle local path planning with multi-modality constraints. *Ocean Eng.* **2021**, *241*, 109890. [[CrossRef](#)]
29. Nazarahari, M.; Khanmirza, E.; Doostie, S. Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm. *Expert Syst. Appl.* **2019**, *115*, 106–120. [[CrossRef](#)]

30. Li, Y.; Zhao, J.; Chen, Z.; Xiong, G.; Liu, S. A robot path planning method based on improved genetic algorithm and improved dynamic window approach. *Sustainability* **2023**, *15*, 4656. [[CrossRef](#)]
31. Aggarwal, S.; Kumar, N. Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges. *Comput. Commun.* **2020**, *149*, 270–299. [[CrossRef](#)]
32. Khedr, A.M.; Al Aghbari, Z.; Khalifa, B.E. Fuzzy-based multi-layered clustering and ACO-based multiple mobile sinks path planning for optimal coverage in WSNs. *IEEE Sens. J.* **2022**, *22*, 7277–7287. [[CrossRef](#)]
33. Liu, C.; Wu, L.; Xiao, W.; Li, G.; Xu, D.; Guo, J.; Li, W. An improved heuristic mechanism ant colony optimization algorithm for solving path planning. *Knowl. Based Syst.* **2023**, *271*, 110540. [[CrossRef](#)]
34. Ajeil, F.H.; Ibraheem, I.K.; Azar, A.T.; Humaidi, A.J. Grid-based mobile robot path planning using aging-based ant colony optimization algorithm in static and dynamic environments. *Sensors* **2020**, *20*, 1880. [[CrossRef](#)] [[PubMed](#)]
35. Ali, H.; Gong, D.; Wang, M.; Dai, X. Path planning of mobile robot with improved ant colony algorithm and MDP to produce smooth trajectory in grid-based environment. *Front. Neurobotics* **2020**, *14*, 44. [[CrossRef](#)] [[PubMed](#)]
36. Feng, K.; He, X.; Wang, M.; Chu, X.; Wang, D.; Yue, D. Path Optimization of Agricultural Robot Based on Immune Ant Colony: B-Spline Interpolation Algorithm. *Math. Probl. Eng.* **2022**, *2022*, 2585910. [[CrossRef](#)]
37. Zhang, X.; Xie, Y.; Jiang, L.; Li, G.; Meng, J.; Huang, Y. Fault-tolerant dynamic control of a four-wheel redundantly-actuated mobile robot. *IEEE Access* **2019**, *7*, 157909–157921. [[CrossRef](#)]
38. Kiran, M.S.; Hakli, H.; Gunduz, M.; Uguz, H. Artificial bee colony algorithm with variable search strategy for continuous optimization. *Inf. Sci.* **2015**, *300*, 140–157. [[CrossRef](#)]
39. Abdulakareem, M.; Raheem, F.A. Development of path planning algorithm using probabilistic roadmap based on ant colony optimization. *Engin. Techn. J.* **2020**, *38*, 343–351. [[CrossRef](#)]
40. Gao, W.; Tang, Q.; Ye, B.; Yang, Y.; Yao, J. An enhanced heuristic ant colony optimization for mobile robot path planning. *Soft Comput.* **2020**, *24*, 6139–6150. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.