

ORIGINAL RESEARCH OPEN ACCESS

An Adaptive Whole-Body Control Approach for Dynamic Obstacle Avoidance of Mobile Manipulators for Human-Centric Smart Manufacturing

Yong Tao^{1,2}  | He Gao¹  | Donghua Tan² | Jiahao Wan² | Baicun Wang³ | Chengxi Li⁴ | Pai Zheng⁴ 

¹Research Institute of Aero-Engine, Beihang University, Beijing, China | ²School of Mechanical Engineering and Automation, Beihang University, Beijing, China | ³School of Mechanical Engineering, Zhejiang University, Hangzhou, China | ⁴Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hung Hom, China

Correspondence: Yong Tao (Taoy@buaa.edu.cn) | Pai Zheng (pai.zheng@polyu.edu.hk)

Received: 30 June 2024 | **Revised:** 23 March 2025 | **Accepted:** 28 March 2025

Handling Editor: Sahil Syed

Funding: This work is supported by National Key R&D Programme of China (No. 2022YFB4700400).

Keywords: human-robot interaction | industrial robots | manufacturing systems | mobile robots | optimal control | robot dynamics

ABSTRACT

In human-centric smart manufacturing (HCSM), the robot's dynamic obstacle avoidance function is crucial to ensuring human safety. Unlike the static obstacle avoidance of manipulators or mobile robots, the dynamic obstacle avoidance in mobile manipulators presents challenges such as high-dimensional planning and motion deadlock. In this paper, an adaptive whole-body control approach for dynamic obstacle avoidance of the mobile manipulators for HCSM is proposed. Firstly, an adaptive global path planning method is proposed to reduce planning dimension. Secondly, lateral coupling effect term and nonlinear velocity damping constraints are formulated to alleviate motion deadlock. Then, a whole-body dynamic obstacle avoidance motion controller is presented. Through simulations and real-world experiments, the planning time is reduced by 18.65% on average, and the path length by 15.94%, compared to the global RRT benchmark algorithm. The dynamic obstacle avoidance experiment simulates the obstacle combinations such as pedestrians moving in opposite direction, traversing and forming a circle during the robot operation. The proposed motion controller can adjust robot movement in real time according to the change of its relative distance from obstacles, meanwhile maintaining an average safe distance of 0.45 m from dynamic obstacles. It is assumed that the proposed approach can benefit dynamic human-robot symbiotic manufacturing tasks from more natural and efficient manipulations.

1 | Introduction

In HCSM, robots and new artificial intelligence technologies are essential for achieving safe and effective industrial human-robot collaboration. With the expansion of intelligent robot technology and its applications, the configuration of robots has undergone significant changes. Different from traditional fixed robot workstations and logistics mobile robots, composite robot configurations that combine platforms and manipulators have attracted increasing researcher's attention [1–4]. Composite-

configuration robot possess both mobility and manipulability, offering an infinite flexible workspace and bringing new possibilities to scenarios such as intelligent manufacturing and logistics. However, due to the redundancy of freedom in composite configurations, the difficulty of planning and control in such configurations has gradually increased. How to develop a safe and efficient motion planning for mobile manipulators that supports the full performance of mobile manipulator equipment has become a concern of researchers [5, 6].

This is an open access article under the terms of the [Creative Commons Attribution-NonCommercial](https://creativecommons.org/licenses/by-nc/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited and is not used for commercial purposes.

© 2025 The Author(s). *IET Collaborative Intelligent Manufacturing* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

In current applications, adoption of a loosely coupled planning method is a common approach used to address the aforementioned issues. This method decouples the motion planning of the robot chassis and manipulators. It then establishes signal connections at the hardware layer [7–9]. The key technologies involved mainly include search-based, sampling-based and optimisation-based techniques for mobile base planning. Additionally, sampling-based and optimisation-based techniques are used in manipulator motion planning. The loosely coupled method treats the mobile base and the manipulator as independent components. The two components do not interfere with each other. In the project's scheme, the mobile base moves into position, followed by the manipulator's action. This scheme can initially meet current engineering requirements but does not fully leverage the flexible advantages of mobile manipulators.

In terms of control algorithms, significant differences exist between the commonly used algorithms for robot chassis and manipulators. Search-based motion planning algorithms that are commonly used with mobile robots mainly operate in the task space. The algorithm identifies collision-free paths that satisfy constraints on the mobile robot's motion and environmental constraints. In contrast, sampling-based algorithms are commonly used for motion planning of manipulators, typically applied in joint space and task space. For the joint space, the algorithm samples to determine the state of each joint of the manipulator, thereby determining the manipulator's pose. For the task space, the algorithm simplifies the problem to a three-dimensional or six-dimensional planning problem at the end effector. Firstly, a three-dimensional path is planned for the robot's end effector. Then, the state of each joint is obtained through inverse kinematics computation. This divergence in the sampling space for planning between robot chassis and manipulator poses a significant challenge in solving planning problems for mobile manipulators.

In recent years, with the increasing demands for flexible, efficient and safe robot manipulation, unified modelling and planning of redundant systems such as mobile manipulators have become feasible. Researchers have explored the application of the tightly coupled approach in motion planning to enhance the motion performance of redundant systems [10–12].

However, the whole-body path planning or motion planning problem of mobile manipulators under tightly coupled schemes is a typical high-dimensional problem. Drawing from experiences in machine learning, methods such as principal component analysis and multidimensional scaling are employed to address the curse of dimensionality by reducing dimensionality. In the field of robot motion planning, reducing the dimensions of the sampling space is essential. Challenges such as accelerating the sampling process, modelling optimisation problems and expediting optimisation computations are also critical for the implementation of tightly coupled schemes.

To address the challenges of high-dimensional motion planning and dynamic obstacle avoidance in HCSM, an adaptive whole-body control approach for dynamic obstacle avoidance of the mobile manipulator is proposed. Firstly, an adaptive global path planning method is proposed to accelerate collision-free

path planning of mobile manipulators. Then, a whole-body dynamic obstacle avoidance controller is presented by formulating the lateral coupling effect term and nonlinear velocity constrain. The rest of the article is organised as follows: in Section 2, typical cases and methods for motion planning of mobile manipulators in complex environments are reviewed. In Section 3, an adaptive whole-body control approach for dynamic obstacle avoidance is proposed. The formulated lateral coupling effect term and nonlinear velocity constrain are integrated into the whole-body dynamic obstacle avoidance controller. In Section 4, experiments on path planning, static obstacle avoidance, and dynamic obstacle avoidance are conducted and compared using different algorithms. Finally, the conclusion and future work are presented in Section 5.

2 | Related Work

Efficient and user-friendly motion planning is vital for ensuring the safe and efficient execution of mobile manipulation. The related key technologies include adaptive global path planning and whole-body dynamic obstacle avoidance motion control.

2.1 | Adaptive Global Path Planning

Current path planning mainly use search-based, sampling-based and optimisation-based approaches. Search-based path planners are suitable for low-degree-of-freedom platforms such as mobile robots [13–15], autonomous vehicles [16], and drones [17]. They offer the advantages of fast computation and smooth path generation. However, they suffer from the curse of dimensionality when solving high-dimensional problems. Sampling-based planners, which are often used for serial and parallel robotic arm platforms [18, 19], are efficient but face challenges such as poor environmental adaptability and lack of path smoothness. Considering the high degree of freedom of wheeled mobile manipulators, development of an efficient and high-quality path planner is a key concern in this field.

To reduce the dimensionality of the search, Gochev et al. [20] first proposed the concept of adaptive dimensions as a means of reducing the search dimensionality. They integrated low-dimensional and high-dimensional search spaces to construct a hybrid search graph and developed an adaptive graph search algorithm. This algorithm, which was validated for motion planning on the PR2 robot, significantly reduced the planning time compared to the baseline. They further proposed an incremental version of the weighted A* algorithm. Building on the idea of adaptive dimensions, Pilia and Gupta [21] first used the PRM algorithm to compute the base roadmap and performed collision detection along the edges between adjacent nodes. If a collision occurred, the robot arm posture was adjusted in a way that was controlled by the HAMP algorithm. Compared to the full PRM method, use of the HAMP algorithm yielded improvements in metrics such as planning time, collision detection count, path length, and planning success rate. Thakar, Rajendran, Kim et al. [22] focused on a smaller search range, concentrating on heuristic search spaces for both the base and the arm. They developed the HS-BI-RRT algorithm,

thereby significantly improving the path planning time and path length compared to those yielded by algorithms such as BI-RRT and WS-Bi-RRT. Chen H L et al. [23], following a partition planning approach, used the A* and FTC algorithms for base path planning. They identified key areas of doors and windows based on visual cues and proposed a heuristic posture adjustment method, that is, the HMP method. Compared to the HAMP and full PRM algorithms, the HMP method produced improvements in runtime and planning time. These cases have established a clear adaptive approach that involves combining heterogeneous algorithms based on search and sampling. By partitioning the state or Cartesian space in a high-dimensional planning problem, planning dimensions are allocated to heterogeneous algorithms, aiming for higher solving efficiency and path quality.

2.2 | Whole-Body Dynamic Obstacle Avoidance

Robot dynamic obstacle avoidance methods include model-based, reactive and learning-based methods, among others. Examples include local planners [24], artificial potential fields [25], control barrier functions [26] and velocity obstacles [27–29]. In the method of separation of perception and planning, the perception system collects environmental information. The planning system then generates obstacle avoiding paths based on this information. This method allows flexible adaptation to different percept and planning needs. Combining local perception information with global path planning enables obstacle avoidance in dynamic environments. Local perception, supplemented by map updates, allows for timely updating of global paths to address sudden obstacles. Reactive methods are real-time sensor-based obstacle avoidance strategies. When a robot detects an obstacle, it reacts immediately to avoid collisions. Reactive methods often simplify the problem into convex optimisation problems [30] that they then solve using intelligent optimisation algorithms and MPC series methods. W. Li, and Xiong [31] proposed a nonlinear model predictive control method for dynamical obstacle avoidance of end-fixed constrained motion planning. Using the optimised result, the velocity control rule can drive the robot to avoid moving people while maintaining the robot's end fixed on the working position at the same time. Learning-based methods utilise machine learning, deep learning [32], reinforcement learning [33, 34] and imitation learning [35] techniques to learn obstacle avoidance behaviour from large amounts of data, demonstrations or digital twin-enabled models. Faverjon and Tournassoud [36] proposed a local motion planning method for systems with high degrees of freedom, introducing the concept of a velocity damper to model the relative distance relationships of moving objects. Haviland and Corke [37] reformulated the velocity damper into inequality constraints, integrated it into an optimisation controller, and achieved dynamic obstacle avoidance for robotic arms with multiple degrees of freedom. Similarly, Haviland Sünderhauf et al. [38] proposed a holistic approach to reactive mobile manipulation. They realised and deployed an optimisation-based whole-body controller for mobile manipulators. The verification was done in static environments. These cases are highly instructive for researchers exploring optimisation-based reactive dynamic obstacle avoidance methods [39].

For human–robot collaborative tasks [40, 41], collision-free efficient and flexible motion planning algorithms for wheeled mobile manipulators are essential. However, existing research on adaptive-dimensional motion planning typically adjusts the arm's pose based on a fixed sequence of chassis poses and fails to leverage the flexibility of whole-body planning. Most existing optimisation-based dynamic obstacle avoidance methods focus on fixed-arm applications, with limited extension to mobile manipulator systems. Therefore, for the future deployment and application of mobile manipulator in HCSM, an adaptive whole-body control approach for dynamic obstacle avoidance of the mobile manipulator is proposed.

3 | Adaptive Whole-Body Control Approach for Dynamic Obstacle Avoidance

3.1 | Overview

Unlike traditional approaches that focus on either the manipulator or the mobile base, the method adopts the idea of adaptive path planning and whole-body optimal control.

The framework includes adaptive global path planning and whole-body dynamic obstacle avoidance motion control, as shown in Figure 1. In tasks such as satellite solar wing robotic assembly and long pole parts robotic handling, the mobile manipulator first adaptively plans a collision-free global path. The motion control problem is reformulated as a quadratic programming (QP) problem. The goal is to determine the optimal whole-body trajectory of the mobile manipulator, satisfying multiple constraints.

The result of the global path planning is a collision-free path P in the robot's workspace from the initial pose to the terminal pose, represented as discrete spatial trajectory points. P can serve as target guidance for the whole-body dynamic obstacle avoidance controller. The whole-body controller formulates a multi-objective optimisation function. It aims to minimise joint rotations and maximise process manipulability while avoiding obstacles. The controller addresses obstacle avoidance, kinematics, joints limit and other constraints. Driven by the optimisation function, the controller continuously solves quadratic programming problems based on the robot's goal pose and current pose. This process obtains continuous control quantities for each joint of the mobile manipulators.

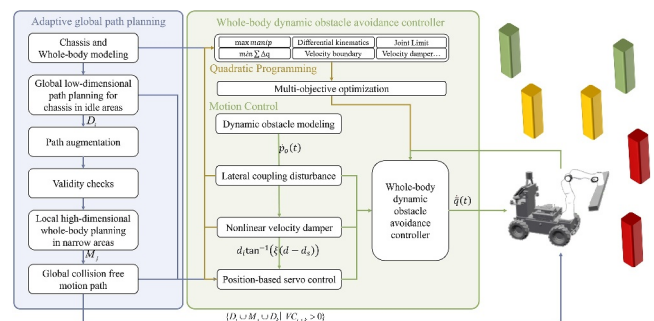


FIGURE 1 | Diagram of the adaptive whole-body dynamic obstacle avoidance.

3.2 | Adaptive Global Path Planning Method

In the operating environments of mobile manipulators, most idle areas pose low collision threats, while part of narrow spaces present high collision threats. Therefore, this method adopts an adaptive approach, planning the base in idle spaces and the whole body in narrow spaces, as illustrated in Figure 2.

The expected workspace is divided into low-dimensional and high-dimensional sampling spaces, corresponding to the base path planning and whole-body path planning tasks, respectively.

Based on defining the poses at the start and end points, this method first conducts global path planning for the chassis. Global path planners such as A*, rapidly-exploring random tree (RRT) and dijkstra can be employed. Planners generate collision-free paths B relative to the static obstacles for the chassis on a global scale.

Path augmentation is then performed by setting the initial pose of the arm joints, thereby obtaining the augmented robot global collision-free path D . When the mobile manipulators perform long pole parts handling, the material often exceeds the envelope size of the mobile manipulator itself, rendering parts of path D invalid. In the simulation environment, by presetting the robot pose, the validity evaluation list VC corresponding to path D can be easily obtained. The simulation environment is the same as the real environment.

With the help of the collision detection algorithm, the collision relationship between the robot and environmental obstacles in any posture can be realised. The collision detection in the simulation environment is implemented based on the bullet physics engine. Specifically, it employs both discrete collision detection and continuous collision detection functionalities, along with the axis-aligned bounding box (AABB) and oriented bounding box (OBB) algorithms. If there is no collision, the event is recorded as one in the VC list, that is, the path point is valid. Otherwise, it is recorded as 0, indicating that the path point is invalid. By analysing VC , the effective and ineffective segments of path D can be decoupled. Then the free area and narrow area of the environment relative to the mobile manipulator system can be analysed. The flowchart is illustrated in Figure 3.

$$P = \{D_i \cup M_j \cup D_k \mid VC_{i,j,k} > 0\} \quad (1)$$

In Equation (1), P is the list of globally valid path points. D_i represents the list of valid path points in free areas. M_j represents the list of valid path points in narrow areas.

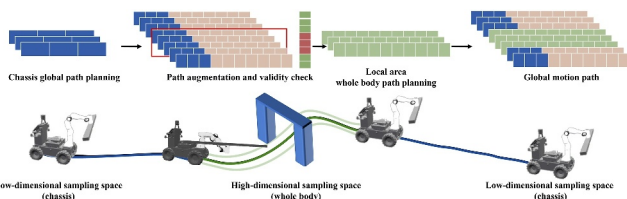


FIGURE 2 | Diagram of the adaptive global path planning method.

$D = \text{conc}(B, \text{ArmPose})$ and denotes the matrix augmentation operation. ArmPose is the initial pose of each joint of the manipulator. The i, j, k represents the index values of the node lists adjacent to free and narrow areas. VC is the path point validity check list.

In the free area, where the space is relatively wide, chassis global path planning is sufficient to meet the goal of avoiding static obstacles in the environment. Narrow areas and their corresponding D_i can meet the obstacle avoidance needs of the chassis but not those of the manipulator and the end effectors. Thus, local-regional whole-body motion planning is required. Local area path planning uses the nodes of adjacent free areas as start and end poses, defining the narrow Cartesian space representation. Subsequently, whole-body sampling planning of the mobile manipulator is conducted to obtain a collision-free path segment M_j in the narrow area. The path segment M_j ensures whole-body avoidance for the environment's static obstacles to the mobile manipulator in the local narrow area. Finally, according to the sequence index of the narrow areas in path D , the path node list of the path segments is dynamically integrated to obtain the global collision-free motion path P for the mobile manipulator.

3.3 | Whole-Body Dynamic Obstacle Avoidance Control Approach

The adaptive global path planning method for mobile manipulators sufficiently meets the requirements for static obstacles avoidance in the environment. However, in collaborative scenarios, there are often dynamic obstacles such as pedestrians and other robots. In order to optimise the dynamic obstacle avoidance performance, a whole-body dynamic obstacle avoidance motion controller is proposed. The task of the whole-body dynamic obstacle avoidance controller is carried out in two main steps.

1. the control quantities of each joint need to be determined under the guidance of global path points and motion constraints.

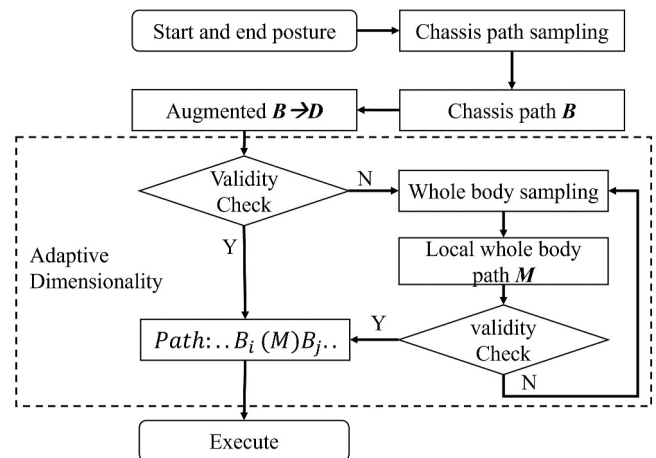


FIGURE 3 | Adaptive planning flowchart.

- the relevant dynamic obstacle motion constraints need to be modelled to address obstacle avoidance problems in dynamic environments.

Dynamic obstacle avoidance for mobile manipulators is challenging, requiring real-time adaptation to moving obstacles while maintaining whole-body motion stability and safety. The controller must ensure collision-free trajectories while satisfying motion constraints, such as joint velocity limits and maintaining the manipulator's manipulability. To address these challenges, we formulate the dynamic obstacle avoidance control problem as a quadratic programming (QP) optimisation problem. The goal is to compute joint velocities $\dot{\mathbf{q}}(t)$ that ensure collision avoidance while maintaining the desired end-effector velocity while also accounting for motion constraints and adapting to real-time changes in the relative distance between the robot and obstacles.

As illustrated in Figure 4, QP takes the desired end-effector velocity $\mathbf{v}(t)$ and real-time changes in the relative distance between the robot and obstacles $\dot{\mathbf{p}}_o(t)$ as inputs and outputs the optimal joint velocities $\dot{\mathbf{q}}(t)$ that satisfy both obstacle avoidance and whole-body motion requirements.

To enhance the controller's performance, we incorporate two key terms inspired by human behaviour: a lateral coupling effect term and a nonlinear velocity damping term. The lateral coupling effect term introduces randomised lateral movements during head-on encounters to mitigate deadlock issues, whereas the nonlinear velocity damping term modulates the robot's speed based on changes in relative distance, improving interaction safety. Both terms are incorporated as inequality constraints in the quadratic optimisation framework of the controller.

The standard form of the QP controller is as Equation (2):

$$\begin{aligned} \min_{\mathbf{x}} \quad & f_o(\mathbf{x}) = \frac{1}{2} \mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{c}^T \mathbf{x}, \\ \text{subject to} \quad & \mathbf{A}_1 \mathbf{x} = \mathbf{B}_1, \\ & \mathbf{A}_2 \mathbf{x} \leq \mathbf{B}_2. \end{aligned} \quad (2)$$

The quadratic programming (QP) problem typically includes the objective function, constraints and the solver. The optimisation outcome, in addition to the objective function, depends heavily on the constraints, which can be classified into equality constraints and inequality constraints. Equality constraints

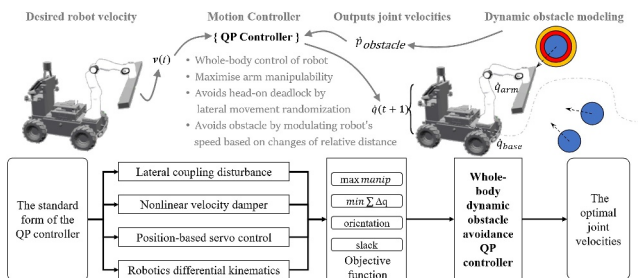


FIGURE 4 | Flowchart of the whole body control for dynamic obstacle avoidance.

$\mathbf{A}_1 \mathbf{x} = \mathbf{B}_1$ define a solution space that satisfies linear equality conditions and directly affect the optimal solution. Inequality constraints $\mathbf{A}_2 \mathbf{x} \leq \mathbf{B}_2$ define a solution space that satisfies several inequality conditions and restrict the range of optimal solutions.

The dynamic obstacle modelling in Figure 1 involves 3D obstacle modelling and relative velocity calculation. This is implemented using the Pybullet simulation environment. In real-world scenarios, a motion capture system calculates relative velocities for human-robot and multi-robot cases. The obtained real-time relative velocities from dynamic obstacle modelling are used to construct the subsequent QP controller.

3.3.1 | Lateral Coupling Effect Term

The equality constraint $\mathbf{A}_1 \mathbf{x} = \mathbf{B}_1$ is often addressed using the differential kinematics of the manipulator. This connects the robot joint velocities $\dot{\mathbf{q}}(t)$ with the end-effector motion velocities $\mathbf{v}(t)$ through the velocity Jacobian matrix. $\mathbf{v}(t)$ is calculated using a pose servo algorithm and typically correlates positively with the error in the task-space pose, as shown in Equation (3):

$$\mathbf{J}(\mathbf{q})\dot{\mathbf{q}}(t) = \mathbf{v}(t) = \beta \psi \left(\left({}^b\mathbf{T}_e \right)^{-1} * {}^b\mathbf{T}_{e^*} \right) \quad (3)$$

where $\mathbf{J}(\mathbf{q})$ is the robot velocity Jacobian matrix. $\dot{\mathbf{q}}(t)$ is the speed of each joint of the robot. $\mathbf{v}(t)$ is the spatial velocity of the end effector. $\mathbf{v}(t)$ denotes the weight for each dimension. ψ is a function that converts the homogeneous error into a six-dimensional spatial error. ${}^b\mathbf{T}_{e^*}$ represents the homogeneous matrix form of the robot's end-effector pose at the current time. ${}^b\mathbf{T}_e$ represents the homogeneous matrix of the robot's end-effector target pose. $\left(\left({}^b\mathbf{T}_e \right)^{-1} * {}^b\mathbf{T}_{e^*} \right)$ represents the transformation matrix from the current pose to the target pose, that is, the pose error.

The above formulas implement a straight-line path from the end effector's current pose to the target pose. However, if the robot, obstacles, and target point all lie in a straight line, the spatial error will only be in the relative direction. The desired velocity output will "pull" the robot towards the obstacle, and this can easily lead to conflicts with dynamic avoidance constraints, resulting in the solver's inability to find a solution and motion deadlock, as shown in Figure 5a.

To address this, a velocity control lateral coupling effect term is introduced as Equation (4). v_{LongV} and v_{LatV} represent the longitudinal velocity and the lateral velocity, respectively. $(\sin(\lambda) + \omega)$ is a trigonometric periodic factor. The term will

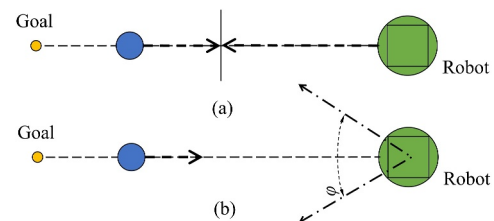


FIGURE 5 | Lateral effect terms increase turning randomness in (b) compared to the deadlock issue in (a).

cause the robot to actively attempt left and right turns while moving towards the target pose, as shown in Figure 5b.

$$v_{LatV} = ((\sin(\lambda) + \omega)) \cdot \operatorname{argmax}(v_{LongV}, v_{LatV}) \quad (4)$$

The *max* function selects the greater value between the longitudinal velocity and lateral velocity. Setting the periodic factor expands the optional range of robot end effector velocities, enhancing flexibility. The *max* function ensures that when the longitudinal error is large, the lateral velocity is influenced primarily by the longitudinal velocity. Similarly, when the lateral error is large, the controller primarily responds to the lateral error, ensuring that the robot can accurately reach the target pose.

3.3.2 | Nonlinear Velocity Damping Constraint

The optional range of the optimised controller solution is limited by the inequality constraints $A_2x \leq B_2$. d is the distance between the robot and the obstacle, v is the rate of change of d . d_i is the effective distance of the damper, and d_s is the stopping distance.

$$\dot{p}_r(t) = J_d(q)\dot{q}(t) = v - n_{or}^T \dot{p}_o(t) \quad (5)$$

In Equation (5), $\dot{p}_r(t)$ represents the velocity of a point on the robot. $J_d(q)$ represents the velocity Jacobian matrix. $\dot{q}(t)$ represents the joint velocities of the robot. n_{or} represents the unit vector originating from the point on the obstacle that is closest to the robot. $\dot{p}_o(t)$ represents the velocity of a point on the obstacle.

Three stages can be distinguished as the robot approaches dynamic obstacles, illustrated in Figure 6. In the first stage, when $d_i \leq d$, the available area for v is large, and it shrinks with decreasing distance d , with minimal impact on the robot's motion. In the second stage, as $d_s \leq d \leq d_i$, the damper begins to act, further reducing the available range of speed v . In the third stage, when $d \leq d_s$, the speed v can be negative, and the robot may either stop or retreat. In the first stage, the goal is for the robot's motion to be minimally affected. In the second stage, the goal is for the robot's motion to adjust rapidly as the distance decreases. In the third stage, when d_s is surpassed or approached, the goal is for the robot to clearly avoid or retreat.

A nonlinear velocity constraint rule is formulated to update the inequality constraints of quadratic programming as Equation (6):

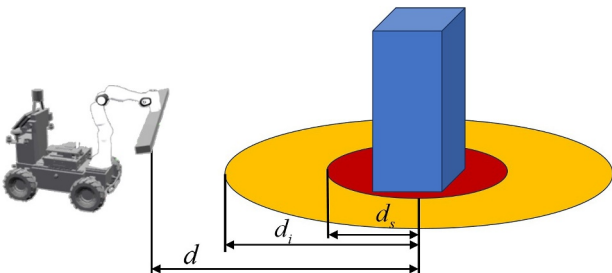


FIGURE 6 | Damper influence area.

$$v \leq d_i \tan^{-1}(\xi(d - d_s)) \quad (6)$$

Integrating the formula,

$$J_d(q)\dot{q}(t) \leq d_i \tan^{-1}(\xi(d - d_s)) - \dot{n}_{or}^T \dot{p}_o(t) \quad (7)$$

In Equation (7), ξ is the state adjustment factor. $J_d(q)\dot{q}(t)$ represents the velocity of a point on the robot. $d_i \tan^{-1}(\xi(d - d_s))$ denotes the rate of change in the relative distance between the robot and obstacle. $\dot{n}_{or}^T \dot{p}_o(t)$ is the velocity of dynamic obstacles towards the corresponding point on the robot.

Robot joint motion affects the velocity of each point on the robot. The motion velocity of each point on the robot is always less than the constant difference between the approach velocity and the obstacle motion velocity. This conclusion defines the motion planning requirements for robot obstacle avoidance with dynamic obstacles. The state adjustment factor adjusts the damper state. Larger values mean stricter velocity restrictions in the second and third stages, and vice versa. Equation (8) represents the correspondence between single-point pairs. However, in actual scenarios, there are multiple point pairs and multiple obstacle collision avoidance requirements; these are updated and organised as follows:

$$a_2 = \begin{pmatrix} J_{d_0}(q_0) & \mathbf{0} \\ \vdots & \vdots \\ J_{d_i}(q_i) & \mathbf{0} \end{pmatrix} \dot{q}(t) \leq \begin{pmatrix} d_i \tan^{-1}(\xi_0(d - d_s)) - \dot{n}_{or_0}^T \dot{p}_{o_0}(t) \\ \vdots \\ d_i \tan^{-1}(\xi_i(d - d_s)) - \dot{n}_{or_i}^T \dot{p}_{o_i}(t) \end{pmatrix} = b_2 \quad (8)$$

3.3.3 | Whole-Body Dynamic Obstacle Avoidance Controller

In the objective function, $x^T Q x$ is a term that minimises the norm of the joint velocities. $c^T x$ is a term that maximises the manipulability of the arm during motion and keeps the orientation of the base consistent with that of the end effector. The improved quadratic programming objective function, equality constraints and inequality constraints are as follows:

$$\begin{aligned} \min_x \quad & f_o(x) = \frac{1}{2} x^T Q x + \mathcal{E}^T x, \\ \text{subject to} \quad & A_1 x = B_1, \\ & A_2 x \leq B_2. \end{aligned} \quad (9)$$

where

$$x^T = (\dot{q} \quad \delta) \quad (10)$$

$$\mathcal{E} = \begin{pmatrix} \operatorname{diag}(\lambda_q) & \mathbf{0} \\ \mathbf{0} & \operatorname{diag}(\lambda_\delta) \end{pmatrix} \quad (11)$$

$$\mathcal{E}^T = (J_m + \epsilon \quad \mathbf{0}) \quad (12)$$

$$A_1 = (J(q)I) \quad (13)$$

$$B_1 = \left\{ \beta \psi \left(({}^bT_e)^{-1} * {}^bT_{e^*} \right) \middle| v_{LatV} \right\}$$

$$A_2 = \begin{pmatrix} 1_{n*n+6} \\ a_2 \end{pmatrix}; B_2^T = \begin{pmatrix} \mathbf{0}_b & \eta \frac{\rho_0 - \rho_s}{\rho_i - \rho_s} & \dots & \eta \frac{\rho_n - \rho_s}{\rho_i - \rho_s} & b_2 \end{pmatrix} \quad (14)$$

In Equations (9) through (14), x represents the control variables. $\dot{q}$ denotes the robot joint velocities. δ is the slack factor. λ_q and λ_s are the weights for minimising the velocity norm and the slack term, respectively. J_m is the manipulability Jacobian matrix. J is the Jacobian matrix. ϵ is the angle between the orientation of the mobile base and the robot arm. In addition, $\mathbf{0}$ is the zero matrix filled with matrix. $J(q)$ is the robot velocity Jacobian matrix. $\dot{q}(t)$ is the speed of each joint of the robot. $v(t)$ is the spatial velocity of the end-effector. β denotes the weight for each dimension. ψ is a function that converts the homogeneous error into a six-dimensional spatial error. bT_e represents the homogeneous matrix form of the robot's end-effector pose at the current time. ${}^bT_{e^*}$ represents the homogeneous matrix of the robot's end-effector target pose. $({}^bT_e)^{-1} * {}^bT_{e^*}$ represents the transformation matrix from the current pose to the target pose. η is the state factor. n is the number of degrees of freedom of the robot. ρ is the distance from the robot's joint to the limit. ρ_i is the influence distance of the limit damper. ρ_s is the stopping distance of the limit. A_1 and B_1 are inequality constraint terms that formulate lateral coupling effect; A_2 and B_2 are inequality constraint terms that integrate joint velocity limit damping with the nonlinear collision damping terms.

The robot system modelling and QP problem formulation describe the internal robot kinematic and the external relative distance relationships between the robot and the environment. With support from the global measurement system, the algorithm can obtain the robot's motion velocity and the relative velocity with dynamic obstacles in real time. Upon determining the current and target poses, the QP controller employs quadratic programming solvers to resolve the optimisation problem. Subject to boundary constraints, differential kinematics equality constraints, and dynamic obstacle avoidance inequality constraints, the controller conducts optimisation analysis to ascertain the optimal joint velocities that maximise the objective function's efficacy.

4 | Experiments

This section first introduces the experimental setup, the simulation environment and the robot platform. Global adaptive path planning and dynamic obstacle avoidance experiments are then conducted in simulated and real transportation scenarios. Through experiments and analysis, the hypothesis that the method helps improve the efficiency of motion planning and the safety of obstacle avoidance is verified.

4.1 | Experimental Setup

The experimental setup is shown in Figure 7. It includes an adaptive path planning experiment and a dynamic obstacle

avoidance experiment. These correspond to (1) the static obstacle avoidance planning test area and (2) the dynamic obstacle avoidance planning test area in the simulation scenario, respectively.

START and END represent the starting pose and the end pose of the robot. The static obstacle avoidance planning test area consists of an L-shaped simulation environment that includes typical obstacles such as long corridors, gateposts and crossbeams. The dynamic obstacle avoidance planning test area includes multiple uniform dynamic obstacles (red cubes) to simulate pedestrians or other robots that may be encountered during the deployment of mobile manipulators.

The mobile manipulator used in the experiment is a self-developed experimental platform that is mainly used to perform material handling and human-robot collaborative tasks, as shown in Figure 8. The robot consists of an MR2000 four-wheel differential chassis, a Franka FR3 seven-degrees-of-freedom manipulator, a sensor system and a long pole cargo to be transferred. The sensor system is composed of an industrial computer, a RoboSense lidar, an Intel D435 camera, inertial measurement unit, power supply and other components. The

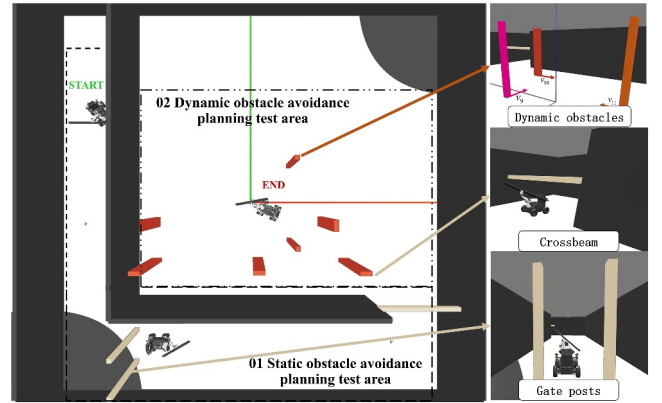


FIGURE 7 | Setup of the simulation experiment.

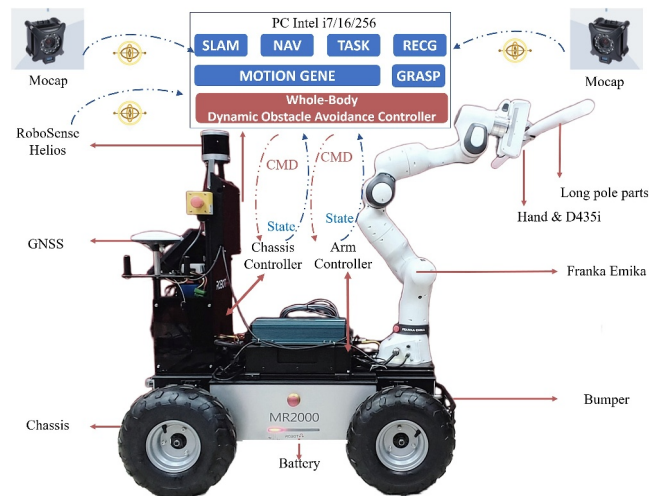


FIGURE 8 | Robot platform.

communication architecture uses the ROS-Melodic version and is deployed on the Ubuntu 18.04 system.

The PC main controller, functioning as the application layer, receives sensor data from the motion capture system (Mocap), LiDAR, and robot-onboard state sensor. It runs applications such as SLAM, navigation, recognition, grasping and motion generation. The proposed DOA controller uses external sensor measurements and robot state data to calculate and generate joint motion commands for obstacle avoidance, which are then sent to the chassis and arm controllers.

4.2 | Adaptive Global Path Planning

The hypothesis is that the idea of adaptive planning can help improve planning efficiency. The experiment was conducted in the static obstacle avoidance planning test area in a simulated environment to verify the hypothesis.

The visualisation results are shown in Figure 9a. Chassis-dominated path planning is carried out. First, the obstacle avoidance requirements of the chassis are considered, and the global collision-free base waypoint list of the chassis is obtained. The first step does not consider the manipulator's obstacle avoidance. The configuration of the manipulator in the path is fixed. There are invalid waypoints that easily cause collisions in narrow areas, as shown in subgraphs a-3 and a-4.

According to the assumption, the distribution of invalid path points determines the range of the front and rear transition areas. The transition starting point is then set to form a local whole-body planning area as shown in subgraph a-2. Full-body collision-free path planning is performed in several layout areas. The front and rear chassis collision-free path fragments are connected to obtain a global collision-free motion path, as shown in subgraphs a-5 and a-6. The global path planning results are shown in subgraph 9(b). The robot's motion follows global path points, which are sparse in idle areas and dense in narrow areas.

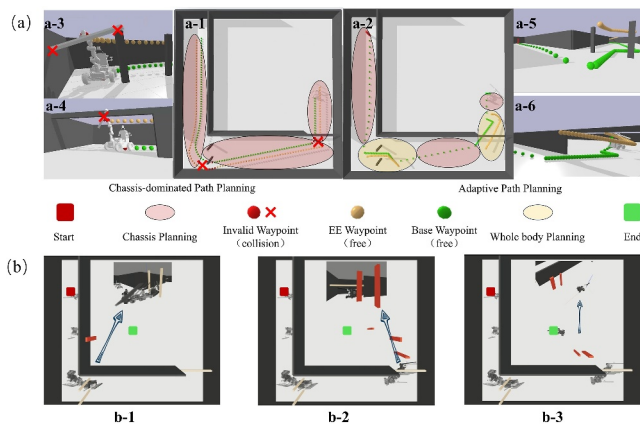


FIGURE 9 | Adaptive path planning results. (a) Comparison of chassis-dominated and adaptive path planning methods. (b) Overall robot motion demonstration.

To further quantify the effectiveness of the proposed method, a comparative experiment was performed with several sampling-based baseline methods. This experiment included test area 01 and test area 02. The experiment unified the starting point and conducted 90 sets of sampling-based global whole-body path planning. RRTConnect, BFMT, BiEST and KPIECE1 were used. The time and path length obtained using each method were recorded and counted. Using the adaptive strategies corresponding to the four methods, the adaptive path planning of 90 component areas was performed, and the above indicators were recorded and counted. The comparisons are shown in Table 1.

Data displays the mean and standard deviation (s.d.) of the planning time and path length for each strategy. These values are used to calculate the percentage of improvement (Imp_PCT) indicator. This indicator measures the improvement obtained when the adaptive strategy is used. When the adaptive strategy is used, the planning process time decreases by up to 28.26%, and the path length decreases by up to 28%.

As shown in the Figure 10, the actual experiment recorded the end-effector movement distance in global chassis-dominated path planning, adaptive path planning and global whole-body path planning cases. Result shows that in the global whole-body planning process, the end of the manipulator makes more random movements to achieve the obstacle avoidance goal. The adaptive planning strategy limits the avoidance action of the end of the manipulator to a narrow space area, which is helpful for reducing the energy consumption and ensuring its safety.

Benefiting from the work and ideas of previous researchers, a partition adaptive path planning method for wheeled mobile manipulators is proposed and verified. This method decouples the planning dimensions, proposing that whole-body high-dimensional motion planning be performed only in the required area. Such an approach is useful for achieving efficient motion planning for mobile manipulators.

4.3 | Whole-Body Dynamic Obstacle Avoidance Control

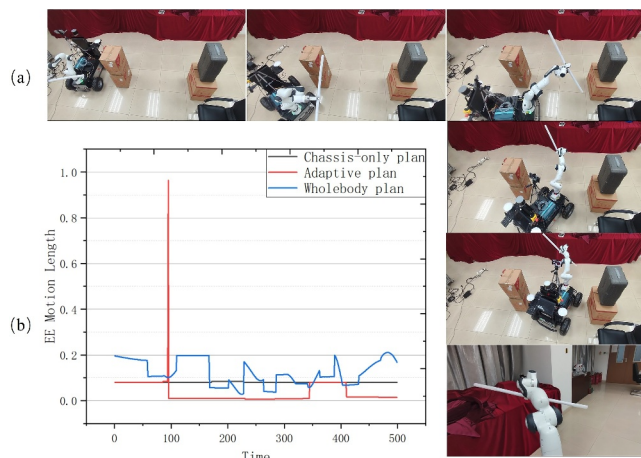
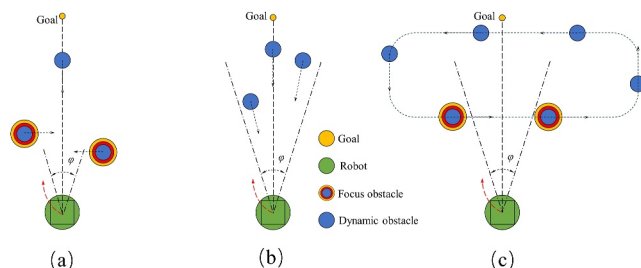
This experiment focuses on verifying the hypothesis that the improved controller can safely control the robot's dynamic obstacle avoidance and achieve the established optimisation goals.

To evaluate the performance of the dynamic obstacle avoidance method, four test scenarios are considered in both the simulation environment and the real environment, as shown in Figure 11. The test scene includes constant-speed dynamic obstacles that traverse, approach, and surround the robot's route. The starting point and the end point of the robot's motion have only longitudinal errors in the initial state. The environments provides information on the relative distance, nearest neighbour pairs and the manipulability during operation.

A screenshot of the experimental process is shown below. At the beginning of the movement, the longitudinal position error term is large. This affects the lateral coupling effect term of the speed

TABLE 1 | Static obstacle area path planning verification.

	Planner	Time (s)		Imp_PCT	Base path length (m)		Imp_PCT
		Mean	s. d.		Mean	s. d.	
1	RRTConnect	0.1375	0.3959	14.47%	35.9411	17.5103	19.64%
	RRTC_HAPP	0.1176	0.1475		28.8834	6.9278	
	BFMT	0.3057	1.5666	14.98%	9.6318	1.8067e-15	4.22%
	BFMT_HAPP	0.2599	0.2303		9.2255	1.8067e-15	
	BiEST	0.5789	0.5210	38.26%	17.3642	3.8182	13.76%
	BiEST_HAPP	0.3574	0.2039		14.9752	4.9317	
	KPIECE1	0.0488	0.0618	13.32%	34.3151	8.8431	7.88%
	KPIECE1_HAPP	0.0453	0.039		31.6123	6.2836	
2	RRTConnect	0.0499	0.0379	31.46%	25.2944	4.5178	18.43%
	RRTC_HAPP	0.0342	0.0243		20.6331	5.8282	
	BFMT	0.0231	0.0727	16.02%	28.3209	2.2853e-14	13.66%
	BFMT_HAPP	0.0194	0.0238		24.4514	1.1930e-14	
	BiEST	0.0899	0.1896	5.12%	45.4177	12.7426	28.00%
	BiEST_HAPP	0.0853	0.1875		32.7017	8.0989	
	KPIECE1	0.0650	0.0569	15.54%	96.3592	25.2615	21.95%
	KPIECE1_HAPP	0.0549	0.0306		75.2115	16.7091	

**FIGURE 10** | Moving screenshot and comparison of the end effector movement. (a) Robot movement screenshot. (b) Comparison of end-effector motion lengths for chassis-dominated, adaptive, and whole-body planning methods.**FIGURE 11** | Dynamic obstacle avoidance experimental setup. Three dynamic obstacle scenarios (traverse, approach, and surround) are set along the robot's path.

control. It is mainly influenced by the longitudinal position error, which changes according to the trend of the trigonometric function. This results in a swing action where the robot first moves to the right, then to the left, as it approaches the target point. As shown in Figure 12a–c, the introduction of swing motion increases the randomness of the orientation of the mobile manipulator and effectively avoids the phenomenon of deadlock with the opposite obstacle. In the process of movement, the robot first ensures that it maintains a warning distance from each dynamic obstacle.

In special cases, it sacrifices some flexibility to seek avoidance. The motion patterns caused by the controller to ensure that the constraints are met during the process of obstacle avoidance include parking, retreating, detouring, acceleration and deceleration. The motion data generated in the simulation environment are deployed in the real robot. The whole-body control and dynamic obstacle avoidance functions during the movement are realised, as shown in Figure 12d.

The obstacle avoidance process focuses primarily on the minimum relative distance between the robot and the dynamic obstacle. The relative distance monitoring point is set on the whole body of the entity rather than on the centroid of the parts. In Figure 13, S19, S10, S11 and B–G represent dynamic obstacle labels, and Manip1, Manip2 and Manip3 represent the manipulator manipulability monitoring values in the three scenarios.

In test scenario (a), after starting to move, the robot first attempts to move to the right front, and the relative distance to the obstacle decreases. The robot is first close to the left obstacle and reaches the warning distance, and the minimum relative distance is 0.3 m. As the left obstacle approaches, the robot retreats and rotates the end joint, seeking avoidance, and retries the right-front motion.

Then, under the control of lateral effect, the robot moves to the left front and passes through and detours between the remaining obstacles. Finally, the process accelerates to the end.

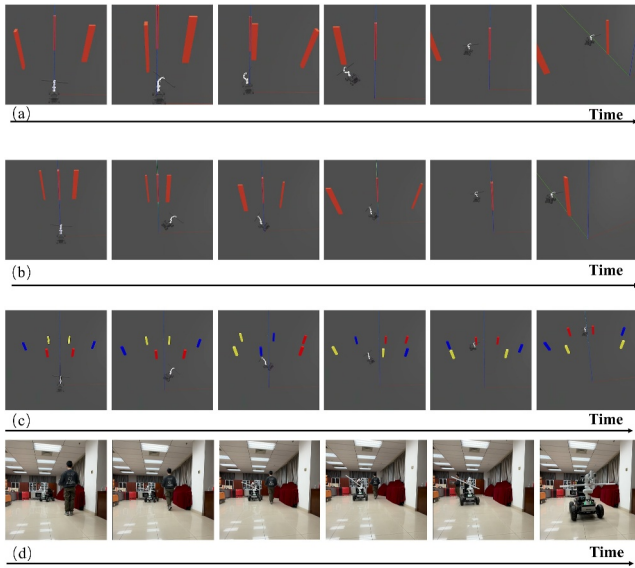


FIGURE 12 | Screenshot of the dynamic obstacle avoidance experiment. (a–c) are simulation environment screenshots. (d) are screenshots from the real-world experimental video.

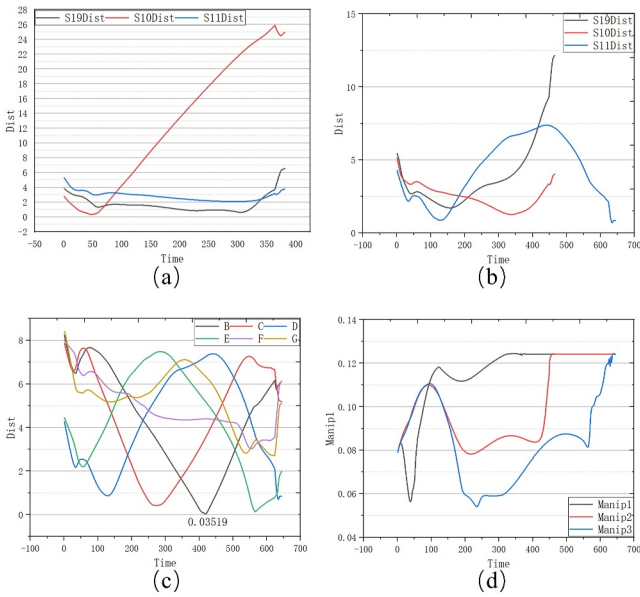


FIGURE 13 | Relative distance and manipulability during obstacle avoidance. S19, S10, S11 and B–G represent dynamic obstacle labels. Manip1, Manip2 and Manip3 represent the manipulators manipulability.

TABLE 2 | Relative distance and manoeuvrability statistics.

	Safe dist	Min dist	MD mean	Min manip	MM mean
Scenario 01	0.3	0.3099	1.0039	0.05634	0.1152
Scenario 02		0.8623	1.2786	0.07826	0.1005
Scenario 03		0.0352	1.1711	0.05408	0.0836

Abbreviations: MD, Min Dist; MM, Min Manip.

In test scenario (b), all the obstacles move towards the robot. After trying to move to the right front, the robot turns to the left front and encounters the left obstacle. The robot then stops to dodge obstacles. After pulling away to the appropriate distance while approaching the middle obstacle, it accelerates to the left and runs to the end point. In test scenario (c), six obstacles on the forward path of the robot are identified as moving around the circular path. In this scenario, there are no obstacles approaching in the middle direction. The robot does not retreat after attempting to move to the right. It is directly affected by the trigonometric function effect term and turns to the left-front direction. Subsequently, the robot stops and waits for the crossing time. After recognising the turning gap of the obstacle, the robot accelerates and enters the interior of the annular obstacle. The minimum relative distance in this process is 0.04 m. The robot stops when encountering obstacles to avoid them while passing through. During the avoidance process, the robot moves back slightly due to the approaching obstacles and then accelerates to reach the target position.

As shown in Table 2, the controller aims to satisfy obstacle avoidance and kinematic constraints during robot movement. It also seeks to maximise manipulator manipulability and optimise operating flexibility. Safe Dist represents the desired distance between the robot and dynamic obstacles. Min Dist indicates the actual minimum distance between the robot and dynamic obstacles during the experimental obstacle avoidance process. MDmean represents the average distance between the robot and dynamic obstacles during above process. Min Manip indicates the minimum manipulability of the robot. MMmean indicates the average manipulability of the robot during the experimental obstacle avoidance process. In test scenario (01), the robot maintains high flexibility but does sacrifice some flexibility when avoiding obstacles on the left in the early stage. The average manipulability is approximately 0.12, and the minimum value is 0.06. In scenario (02), the robot encounters an obstacle in the middle of its movement. Because maintenance of relative distance is primarily achieved through the movement of the manipulator, the flexibility is maintained at a low value for a short time. The average manipulability is 0.1, and the minimum manipulability is 0.08. In scenario (c), they are 0.008 and 0.05, respectively.

In the dynamic obstacle avoidance scheme described in this article, real-time position monitoring and feedback obtained in the scene are crucial to the proper function of the obstacle avoidance controller. This information is used as a known input of the scene and external sensors. During the real deployment, the robot's recognition and prediction functions of dynamic obstacle movements require additional consideration. This can be achieved through global measurement, real-time perception, and other methods.

5 | Conclusion

An adaptive whole-body control approach was proposed for dynamic obstacle avoidance of mobile manipulators for HCSM. Firstly, an adaptive global path planning method for mobile manipulators was proposed to reduce the planning dimension and accelerate motion planning. Compared with the RRT benchmark algorithm, the planning time and path length are shortened by 18.65% and 15.94%. Then a whole-body dynamic obstacle avoidance controller was presented by formulating lateral coupling effect term and nonlinear velocity constrain. The motion deadlock in dynamic obstacle avoidance of mobile manipulator was alleviated. In whole-body dynamic obstacle avoidance control experiments, the controller could optimise the manipulator's flexibility while ensuring that the robot is collision-free globally. The average relative distance between the robot and the obstacle was 0.45 m. The average manipulability of the arm was 0.1. It is hoped that the proposed approach can benefit dynamic human-robot symbiotic manufacturing tasks from more natural and efficient manipulations. The future works can be investigated on the multi-sensor fusion perception and embodied intelligence algorithm deployment, to enhance the mobile robot's capabilities.

Author Contributions

Yong Tao: conceptualisation, funding acquisition, project administration, resources, supervision. **He Gao:** investigation, methodology, software, visualisation, writing – original draft. **Donghua Tan:** software, validation, visualisation. **Jiahao Wan:** data curation, investigation, validation. **Baicun Wang:** project administration, supervision, writing – review and editing. **Chengxi Li:** supervision, writing – review and editing. **Pai Zheng:** project administration, supervision, writing – review and editing.

Consent

The authors have nothing to report.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

Data available within the article and supplementary materials.

Permission to Reproduce Materials From Other Sources

The authors have nothing to report.

References

1. Q. L. Qi, F. Tao, T. L. Hu, et al., “Enabling Technologies and Tools for Digital Twin,” *Journal of Manufacturing Systems* 58 (2021): 3–21, <https://doi.org/10.1016/j.jmsy.2019.10.001>.
2. N. Ghodisian, K. Benfriha, A. Olabi, V. Gopinath, and A. Arnou, “Mobile Manipulators in Industry 4.0: A Review of Developments for Industrial Applications,” *Sensors* 23, no. 19 (2023): 8026, <https://doi.org/10.3390/s23198026>.
3. M. Wang, Z. S. Chen, K. X. Guo, et al., “Millimeter-level Pick and Peg-In-Hole Task Achieved by Aerial Manipulator,” *IEEE Transactions on Robotics* 40 (2024): 1242–1260, <https://doi.org/10.1109/tro.2023.3338956>.

4. S. Zimmermann, R. Poranne, S. Coros, et al., *Go Fetch! - Dynamic Grasps using boston dynamics Spot with External Robotic Arm* (IEEE International Conference on Robotics and Automation (ICRA), 2021), 4488–4494.
5. Q. Huang, K. Tanie, and S. Sugano, “Coordinated Motion Planning for a Mobile Manipulator Considering Stability and Manipulation,” *International Journal of Robotics Research* 19, no. 8 (2000): 732–742, <https://doi.org/10.1177/02783640022067139>.
6. Z. M. Bi, C. M. Luo, Z. H. Miao, B. Zhang, W. Zhang, and L. Wang, “Safety Assurance Mechanisms of Collaborative Robotic Systems in Manufacturing,” *Robotics and Computer-Integrated Manufacturing* 67 (2021): 102022, <https://doi.org/10.1016/j.rcim.2020.102022>.
7. J. Xu, K. Harada, W. Wan, et al., “Planning an Efficient and Robust Base Sequence for a Mobile Manipulator Performing Multiple Pick-And-Place Tasks,” in *IEEE International Conference on Robotics and Automation (ICRA)*, 2020), 11018–11024: Online.
8. T. Sandy and J. Buchli, *Dynamically Decoupling Base and End-Effector Motion for Mobile Manipulation Using Visual-Inertial Sensing* (IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017), 6299–6306.
9. S. Thakar, P. Rajendran, A. M. Kabir, and S. K. Gupta, “Manipulator Motion Planning for Part Pickup and Transport Operations From a Moving Base,” *IEEE Transactions on Automation Science and Engineering* 19, no. 1 (2022): 191–206, <https://doi.org/10.1109/tase.2020.3020050>.
10. K. Jang, S. Kim, and J. Park, “Motion Planning of Mobile Manipulator for Navigation Including Door Traversal,” *IEEE Robotics and Automation Letters* 8, no. 7 (2023): 4147–4154, <https://doi.org/10.1109/lra.2023.3279612>.
11. M. N. Finean, W. Merkt, and I. Havoutis, “Where Should I Look? Optimised Gaze Control for Whole-Body Collision Avoidance in Dynamic Environments,” *IEEE Robotics and Automation Letters* 7, no. 2 (2022): 1095–1102, <https://doi.org/10.1109/lra.2021.3137545>.
12. Y. Q. Wu, E. Lamon, F. Zhao, W. Kim, and A. Ajoudani, “Unified Approach for Hybrid Motion Control of Moca Based on Weighted Whole-Body Cartesian Impedance Formulation,” *IEEE Robotics and Automation Letters* 6, no. 2 (2021): 3505–3512, <https://doi.org/10.1109/lra.2021.3062316>.
13. Z. Y. Lu, Z. C. Liu, M. Campbell, and K. Karydis, “Online Search-Based Collision-Inclusive Motion Planning and Control for Impact-Resilient Mobile Robots,” *IEEE Transactions on Robotics* 39, no. 2 (2023): 1029–1049, <https://doi.org/10.1109/tro.2022.3211131>.
14. A. T. Le, T. D. Le, and R. Roka, *Search-based Planning and Replanning in Robotics and Autonomous Systems* (Advanced Path Planning for Mobile Entities, 2018), 63–89.
15. Y. Tao, H. Gao, Y. F. Wen, L. Duan, and J. Lan, “Glass Recognition and Map Optimization Method for Mobile Robot Based on Boundary Guidance,” *Chinese Journal of Mechanical Engineering* 36, no. 1 (2023): 74, <https://doi.org/10.1186/s10033-023-00902-9>.
16. Z. Ajanovic, B. Lacevic, B. Shyrokau, et al., *Search-based Optimal Motion Planning for Automated Driving* (IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2018), 4523–4530.
17. S. K. Liu, K. Mohta, N. Atanasov, and V. Kumar, “Search-based Motion Planning for Aggressive Flight in SE(3),” *IEEE Robotics and Automation Letters* 3, no. 3 (2018): 2439–2446, <https://doi.org/10.1109/lra.2018.2795654>.
18. I. Becerra, H. Yervilla-Herrera, E. Antonio, and R. Murrieta-Cid, “On the Local Planners in the RRT* for Dynamical Systems and Their Reusability for Compound Cost Functionals,” *IEEE Transactions on Robotics* 38, no. 2 (2022): 887–905, <https://doi.org/10.1109/tro.2021.3098244>.

19. F. Burget, M. Bennewitz, and W. Burgard, *Bi2rrt*: An Efficient Sampling-Based Path Planning Framework for Task-Constrained Mobile Manipulation* (IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016), 3714–3721.
20. K. Gochev, A. Safonova, and M. Likhachev, “Incremental Planning With Adaptive Dimensionality,” in *International Conference on Automated Planning and Scheduling, Rome, Italy*, 2013), 82–90.
21. V. Pilić and K. Gupta, “A Hierarchical and Adaptive Mobile Manipulator Planner,” in *14TH IEEE-RAS International Conference on Humanoid Robots (HUMANOIDS), Madrid, Spain* (2014), 45–51.
22. S. Thakar, P. Rajendran, H. Kim, et al., *Accelerating Bi-directional Sampling-Based Search for Motion Planning of Non-holonomic Mobile Manipulators* (IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020), 6711–6717.
23. H. L. Chen, X. Z. Zang, Y. B. Liu, X. Zhang, and J. Zhao, “A Hierarchical Motion Planning Method for Mobile Manipulator,” *Sensors* 23, no. 15 (2023): 6952, <https://doi.org/10.3390/s23156952>.
24. C. Roesmann, W. Feiten, T. Woesch, et al., “Trajectory Modification Considering Dynamic Constraints of Autonomous Robots,” in *7th German Conference on Robotics (ROBOTIK)* (2012), 1–6.
25. C. W. Warren, *Global Path Planning Using Artificial Potential Fields* (IEEE International Conference on Robotics and Automation (ICRA), 1989), 316–317.
26. A. D. Ames, S. Coogan, M. Egerstedt, et al., “Control Barrier Functions: Theory and Applications,” in *18TH European Control Conference (ECC)* (2019), 3420–3431.
27. P. Fiorini and Z. Shiller, “Motion Planning in Dynamic Environments Using Velocity Obstacles,” *International Journal of Robotics Research* 17, no. 7 (1998): 760–772, <https://doi.org/10.1177/027836499801700706>.
28. J. Snape, J. Van Den Berg, S. J. Guy, and D. Manocha, “The Hybrid Reciprocal Velocity Obstacle,” *IEEE Transactions on Robotics* 27, no. 4 (2011): 696–706, <https://doi.org/10.1109/tro.2011.2120810>.
29. J. Van Den Berg, M. Lin, D. Manocha, et al., *Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation* (IEEE International Conference on Robotics and Automation (ICRA), 2008), 1928–1935.
30. Y. M. Liu, Z. H. Cao, H. Xiong, J. Du, H. Cao, and L. Zhang, “Dynamic Obstacle Avoidance for Cable-Driven Parallel Robots With Mobile Bases via Sim-To-Real Reinforcement Learning,” *IEEE Robotics and Automation Letters* 8, no. 3 (2023): 1683–1690, <https://doi.org/10.1109/lra.2023.3241801>.
31. W. Li and R. Xiong, “Dynamical Obstacle Avoidance of Task-Constrained Mobile Manipulation Using Model Predictive Control,” *IEEE Access* 7 (2019): 88301–88311, <https://doi.org/10.1109/access.2019.2925428>.
32. P. Z. Chen, J. A. Pei, W. Q. Lu, and M. Li, “A Deep Reinforcement Learning Based Method for Real-Time Path Planning and Dynamic Obstacle Avoidance,” *Neurocomputing* 497 (2022): 64–75, <https://doi.org/10.1016/j.neucom.2022.05.006>.
33. P. Zheng, C. Li, J. Fan, and L. Wang, 1, “A Vision-Language-Guided and Deep Reinforcement Learning-Enabled Approach for Unstructured Human-Robot Collaborative Manufacturing Task Fulfilment,” *CIRP Annals* 73, no. 1 (2024): 341–344, <https://doi.org/10.1016/j.cirp.2024.04.003>.
34. C. X. Li, P. Zheng, Y. Yin, Y. M. Pang, and S. Huo, “An Ar-Assisted Deep Reinforcement Learning-Based Approach towards Mutual-Cognitive Safe Human-Robot Interaction,” *Robotics and Computer-Integrated Manufacturing* 80 (2023): 102471, <https://doi.org/10.1016/j.rcim.2022.102471>.
35. P. Zheng, S. F. Li, J. M. Fan, and L. Wang, 1, “A Collaborative Intelligence-Based Approach for Handling Human-Robot Collaboration Uncertainties,” *CIRP Annals-Manufacturing Technology* 72, no. 1 (2023): 1–4, <https://doi.org/10.1016/j.cirp.2023.04.057>.
36. B. Faverjon and P. Tournassoud, “A Local Based Approach for Path Planning of Manipulators With a High Number of Degrees of Freedom,” in *IEEE International Conference on Robotics and Automation, Raleigh, North Carolina, USA*, 1987), 1152–1159.
37. J. Haviland and P. Corke, “Neo: A Novel Expeditious Optimisation Algorithm for Reactive Motion Control of Manipulators,” *IEEE Robotics and Automation Letters* 6, no. 2 (2021): 1043–1050, <https://doi.org/10.1109/lra.2021.3056060>.
38. J. Haviland, N. Sünderhauf, and P. Corke, “A Holistic Approach to Reactive Mobile Manipulation,” *IEEE Robotics and Automation Letters* 7, no. 2 (2022): 3122–3129, <https://doi.org/10.1109/lra.2022.3146554>.
39. B. Burgess-Limerick, J. Haviland, C. Lehnert, and P. Corke, “Reactive Base Control for On-The-Move Mobile Manipulation in Dynamic Environments,” *IEEE Robotics and Automation Letters* 9, no. 3 (2024): 2048–2055, <https://doi.org/10.1109/lra.2024.3354615>.
40. C. Li, P. Zheng, P. Zhou, Y. Yin, C. K. Lee, and L. Wang, “Unleashing Mixed-Reality Capability in Deep Reinforcement Learning-Based Robot Motion Generation towards Safe Human-Robot Collaboration,” *Journal of Manufacturing Systems* 74 (2024): 411–421, <https://doi.org/10.1016/j.jmsy.2024.03.015>.
41. B. C. Wang, F. Tao, X. D. Fang, C. Liu, Y. Liu, and T. Freiheit, “Smart Manufacturing and Intelligent Manufacturing: A Comparative Review,” *Engineering* 7, no. 6 (2021): 738–757, <https://doi.org/10.1016/j.eng.2020.07.017>.

Appendix A: Supplementary Data

Supplementary material related to this article can be found online at <https://github.com/GaH703/AWBDOA>.