2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

DMRP-Bench: An Integrated, Unified Multi-Robot Motion Planning Benchmark in Dynamic Environments

Zhijie Hu, Xuebo Zhang*, Runhua Wang, Yichen Li, Qingchen Bi, Hengyi Liu

***Abstract*— Multi-robot motion planning in dynamic environments poses challenges to safe, efficient coordination, yet a fair, unified testbed for evaluating diverse algorithms remains absent. We present DMRP-Bench, a comprehensive framework to address this gap. It features a layered architecture integrating global and local planners, enabling a comprehensive analysis of their combinations across both macroscopic system-level outcomes and fine-grained inter-robot interactions. High-fidelity indoor scenarios (e.g., library, mall, office) simulating varied spatial layouts and pedestrian dynamics are built within the NVIDIA Isaac Sim environment. Extensive experiments on sixteen planner combinations reveal not only critical trade-offs between trajectory efficiency and safety, but also facilitate a deeper assessment of inter-robot coordination. By correlating path execution fidelity with interaction outcomes, these experiments enable quantitative diagnosis of how local behaviors influence interactive dynamics and holistic system performance.**

I. INTRODUCTION

Multi-robot motion planning is widely applied in warehouse logistics [1]–[3], service robotics [4], [5], and smart manufacturing [6]–[8]. The core challenge lies in coordinating multiple robots to complete tasks safely and efficiently in dynamic environments, while avoiding path conflicts, resource contention, and local congestion [9].

Despite significant progress in multi-robot motion planning (MRMP), evaluating these systems in dynamic conditions faces several key limitations:

1) Existing frameworks often lack standardized interfaces for integrating global and local planners, relying on ad-hoc implementations, which increases integration costs and limits reproducibility [10], [16]–[18]. Furthermore, their benchmarks primarily focus on the global planner performance, downplaying the critical effects of local planners on real-time obstacle avoidance and velocity adjustment [10], [13], [18].
2) Evaluation metrics primarily focus on system-level outcomes like path efficiency or task completion time. They lack fine-grained quantification of inter-robot interactions, conflict density, or coordination effectiveness [10]–[13]. Specifically, they fail to capture how local planners' dynamic feedback, such as path replanning or interaction frequency, affects global path execution. This limits insights into performance variations driven by local control strategies.

*This work was supported in part by the NSFC (62293513/62293510). *(Corresponding author: Xuebo Zhang.)*

The authors are with the Tianjin Key Laboratory of Intelligent Robotics, the College of Artificial Intelligence, and the Institute of Robotics and Automatic Information System, Nankai University, Tianjin 300350, China. huzhijie@mail.nankai.edu.cn, zhangxuebo@nankai.edu.cn.

Fig. 1. A snapshot from our DMRP-Bench benchmark, showing multiple robots and dynamic pedestrians coexisting in a complex and high-fidelity library scenario rendered in NVIDIA Isaac Sim.

3) Common simulation platforms like Gazebo [14] and Flatland [15] offer limited support for realistic dynamic pedestrians and sensor-rich environments. These limitations hinder a comprehensive evaluation of multi-robot planning systems in complex environments with dynamic interactions and real-time perception.

To address these challenges, we introduce **DMRP-Bench**, a unified benchmarking framework for evaluating multi-robot motion planning strategies in dynamic environments. Built on ROS and NVIDIA Isaac Sim [19], our framework provides a reproducible, extensible, and systematic assessment of diverse planner combinations. The complete implementation is open-sourced at `https://github.com/NKU-MobFly-Robotics/DMRP-Bench`.

Our main contributions are as follows:

1) A modular benchmark architecture is developed with an enhanced global-to-local planner integration. Specifically, we introduce full-path subscription to replace goal-by-goal dispatch, enabling local controllers to track global paths responsively with real-time sensor feedback, and provide a standardized interface layer to support algorithm-level extensibility and plugin-based planner integration. This design ensures reproducible comparisons of diverse local planners and supports their integration with global planning.
2) A set of novel metrics is introduced to quantify critical, fine-grained aspects of multi-robot interaction, such as coordination risks, traffic congestion, and flow stability, aspects often overlooked by conventional evaluation. These metrics form the foundation of a systematic diagnostic approach that correlates

interaction outcomes with the execution fidelity of a planner (measured by Path Following Deviation, PFD). This correlative analysis directly enables a quantitative diagnosis of how local behaviors impact system-wide coordination, a capability largely absent in prior work.

3) High-fidelity simulation scenarios are constructed using NVIDIA Isaac Sim, covering diverse indoor layouts such as libraries, malls, and open-plan offices. The platform supports realistic modeling of dynamic pedestrians, integrated sensor streams, and physics-based interactions, enabling comprehensive evaluation of multi-robot planning strategies in dynamic, interactive settings.

Results show that the proposed framework effectively supports performance comparison and optimization across various planning strategies, providing a reproducible foundation for future research on interaction-aware multi-robot motion planning. The remainder of this paper is organized as follows: Section II reviews related work. Section III details the benchmark framework design. Section IV presents the experimental setup and analysis of global and local planners across three dynamic scenarios. Section V concludes and outlines future work.

II. RELATED WORK

Research on multi-robot motion planning has advanced rapidly in recent years, encompassing global path planning, hierarchical architectures, and benchmarking frameworks. This section reviews representative methods and their limitations, laying the foundation for our proposed framework.

A. Path Planning for Multi-Robot Systems

Multi-robot path planning requires optimizing both path length and task duration while ensuring collision-free trajectories. Centralized methods such as CBS [20] construct a constraint tree (CT) to compute globally optimal solutions, but their computational complexity increases exponentially with the number of robots [21]. To improve efficiency, ECBS [22] introduces ϵ-bounded suboptimality to reduce search complexity, while EECBS [23] further accelerates planning by estimating conflict likelihood online. SIPP [24] exploits safe intervals to reduce the search space and performs efficiently in static settings. Its multi-agent variant, SIPP-m, applies prioritized planning to avoid time conflicts but remains limited in adaptability and coordination in dynamic settings.

In contrast to centralized approaches, distributed path planning has been increasingly explored for its scalability and real-time reactivity. Among them, the Reciprocal Velocity Obstacles (RVO) [25] performs collision avoidance through velocity-space modeling and is effective in dense environments. MOPSO [26] adopts multi-objective particle swarm optimization to improve agent coordination. PRIMAL [27], fusing reinforcement and imitation learning, enables scalable multi-agent planning without explicit inter-agent communication.

Although distributed methods offer advantages in scalability and responsiveness, they lack global consistency and often involve higher implementation complexity. Given the trade-offs, our framework adopts centralized multi-robot planning to ensure global coordination, with local controllers handing real-time adaptability.

B. Hierarchical Planning and Execution

In layered path planning architectures, the global planner generates the initial path, while the local planner performs real-time adjustments in response to environmental changes. The ROS Navigation Stack [28] is a standard framework for single-robot navigation, but lacks coordination optimization for multi-robot systems. The mapf_ros package [16] supports centralized planning, but its "goal-by-goal" execution mechanism denies the local planner a holistic view of global path. Lacking this foresight, it is constrained to purely reactive adjustments, which ultimately degrades path continuity and increases latency in dynamic environments. The method by Yeh et al. [29] uses Dijkstra as a single-robot global planner, which lacks multi-robot coordination and limits its applicability in collaborative scenarios, while the reinforcement learning approach by Salimpour et al. [30] has potential but lacks full integration with global planners.

We enhance mapf_ros via full-path subscription, allowing local planners to continuously receive global paths and combine them with real-time sensor data for trajectory generation. This design improves execution continuity and responsiveness, while maintaining a centralized planning backbone and modular local execution.

C. Benchmark for Multi-Robot Navigation

Benchmarking platforms are essential for evaluating multi-robot planning strategies under realistic yet controlled conditions. However, most existing tools only partially support the full planning pipeline—lacking dynamic obstacle modeling, inter-robot interaction metrics, or physically realistic simulation. These limitations hinder the systematic study of planner performance in complex and interactive environments.

MotionBenchMaker [12] and MRPB 1.0 [17] support dataset generation and local planner evaluation for single-robot scenarios, but lack support for inter-robot coordination. Asprilo [13] offers grid-based benchmarking with task assignment for warehouse logistics, yet omits continuous-time dynamics and physical realism. MRP-Bench [10] extends evaluation to centralized multi-robot planning but assumes static environments. Open-RMF [18], while addressing fleet-level coordination, does not support dynamic obstacles or detailed evaluation of path quality and robot interaction. Recently, SMART [36] addresses scalable execution of MAPF plans using physics-based simulators, while REMROC [35] provides a platform for comparing coordination algorithms in human-shared environments. However, neither systematically examines the interaction between global and local planners, as SMART focuses on execution and REMROC on coordination strategies, and both rely on traditional high-level metrics. Collectively, these tools highlight a persistent gap in comprehensive benchmarking under dynamic and interaction-rich conditions.

III. BENCHMARK DESIGN

To address the lack of standardized evaluation protocols and limited coordination mechanisms in multi-robot planning under dynamic environments, we propose DMRP-Bench, a unified benchmarking framework built upon mapf_ros. The system features a modular, layered architecture that spans the full multi-robot navigation stack, including high-level coordination, lower-level control, and performance evaluation. It is designed to support systematic and fine-grained assessment of integrated navigation behavior under dynamic environments, emphasizing overall system performance rather than isolated module optimization.

TABLE I
COMPARISON OF MULTI-ROBOT NAVIGATION BENCHMARKS

Feature	DMRP	MRP	MotionBench	Asprilo	OpenRMF
Dyn-Obst	✓	×	×	×	×
Unif-Interface	✓	✓	×	✓	×
Inter-Metrics	✓	×	×	×	✓
Hi-Fi Sim	✓	✓	×	×	✓
Glob-Loc Int	✓	✓	×	×	✓

As Table I shows, DMRP-Bench addresses key limitations of existing platforms. While benchmarks like MRP-Bench and Open-RMF also support multi-layer planning, they lack standardized interfaces to integrate diverse local planners, which limits their ability to conduct systematic and reproducible comparisons.

A. Layered Architecture Overview

Figure 2 illustrates the overall system architecture and the data flow among its core components. The framework comprises five key modules: the Scenario Setup, the Isaac Sim simulation backend, the centralized System Manager Node, the distributed Local Control units, and the Performance Metrics Analyzer. The scenario setup module specifies the 3D environment layout, robot configurations, and dynamic pedestrian motion, which are loaded directly into the Isaac Sim stage. The Isaac Sim backend is GPU-accelerated and operates with two core engines: the Omniverse RTX Renderer and the PhysX 5 Engine. These two engines work in tandem to generate realistic sensor feedback for the local control units and to provide the detailed world model from which the global costmap is derived. Meanwhile, the ROS Bridge facilitates real-time data exchange between the simulation and the ROS-based control nodes.

At runtime, the System Manager acts as the central coordinator. It receives the global costmap from the simulator, assigns start and goal positions for each robot, and invokes the MAPF Plugins to generate conflict-free global paths. These global plans are then delivered to the corresponding Local Control units. Each unit runs a standard `move_base`[1] stack equipped with a plugin-based local controller (e.g., TEB, DWA, MPC) and a global plan subscriber for path following. Concurrently, the Performance Metrics Analyzer receives Runtime Data and logs system-level metrics, forming the basis for reproducible and quantitative benchmarking.

Together, these modules constitute a closed-loop evaluation pipeline encompassing perception, planning, execution, and analysis.

[1]`move_base` is a foundational ROS package providing a complete navigation solution for mobile robots, including global and local path planning, and obstacle avoidance.

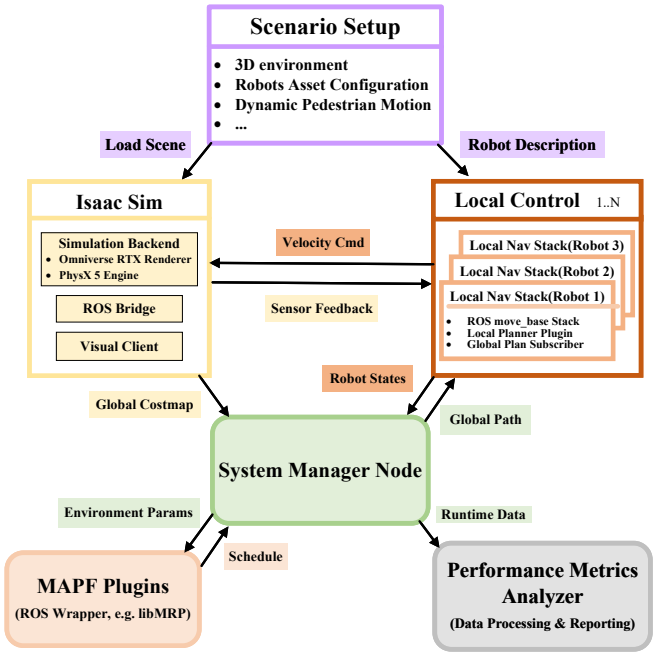


Fig. 2. System Architecture and Workflow. Illustrates the data flow and integration of the six core modules for multi-robot planning benchmark.

B. Global Planner Integration

The global planning layer is responsible for centrally coordinating all robots by computing a globally conflict-free set of paths. In our framework, we integrate four representative multi-agent path finding (MAPF) algorithms: CBS, ECBS, EECBS, and SIPP. These planners are implemented as interchangeable modules and follow a standardized interface for integration with components.

To enhance the quality of planned trajectories, we introduce an extended version of ECBS that operates on 8-connected grid maps, referred to as **ECBS-8**. Unlike the default 4-connected implementation, ECBS-8 allows both orthogonal and diagonal transitions. This modification enables the generation of smoother and shorter paths, better reflecting the continuous nature of robot motion in real-world environments.

The extension consists of the following changes: (1) expanding the motion primitives to include diagonal moves; (2) applying floating-point step costs (e.g., $\sqrt{2}$ for diagonal moves); (3) replacing the Manhattan heuristic with Chebyshev distance to remain admissible; and (4) extending the constraint model to correctly handle diagonal collisions.

The core difference in path search procedure is outlined in Algorithm 1. The rest of the ECBS pipeline, including the conflict tree and focal list logic, remains unchanged.

C. Local Controller Integration

The primary role of the local control layer is to translate a global path into real-time motion commands while adapting to dynamic obstacles and sensor uncertainties. To facilitate a systematic comparison of different control strategies, DMRP-Bench features a highly modular architecture. All controllers are implemented as plugins conforming to the standard ROS `nav_core::BaseLocalPlanner` interface, allowing for seamless substitution. We leverage this modularity to integrate four controllers that span diverse paradigms: the optimization-based Timed Elastic Band (TEB) [31] and Model Predictive Control (MPC) [33], the sampling-based Dynamic Window Approach (DWA) [32], and the E3MoP planner [34] for dynamic interactions.

Furthermore, to enhance execution quality, we fundamentally redesign the data flow between the global and local layers. We replace the default waypoint-by-waypoint dispatch in mapf_ros with a full-path subscription mechanism. This ensures the local planner always has access to the complete upcoming path, enabling smoother and more proactive control decisions for responsive navigation.

Algorithm 1 Enhanced ECBS-8 with Diagonal Conflict Resolution

Input: $\mathcal{A}$ (set of agents), start-goal pairs, map, ε
Initialize root with 8-connected plans: $\forall a_i \in \mathcal{A}$ plan with:
- Actions: $\{\uparrow, \downarrow, \leftarrow, \rightarrow, \nwarrow, \nearrow, \swarrow, \searrow\}$
- Cost: $c(s, s') = \begin{cases} 1 & \text{orthogonal} \\ \sqrt{2} & \text{diagonal} \end{cases}$
- Heuristic: $h_{\text{Chebyshev}}(s) = \max(\Delta x, \Delta y) + (\sqrt{2} - 1)\min(\Delta x, \Delta y)$
- **Diag check:** Free adjacent cells for diagonal moves

while OPEN not empty **do**
  $N \leftarrow$ OPEN.pop()
  if validate($N.solution$) **then return** $N.solution$
  $(a_i, a_j, t, \text{type}) \leftarrow$ FindEarliestConflict($N.solution$)
  for $a_k \in \{a_i, a_j\}$ **do**
    $\mathcal{C}_{a_k} \leftarrow \begin{cases} \text{Vertex: } V_{\text{Constraint}}(t, \text{loc}) \\ \text{Edge: } E_{\text{Constraint}}(t, s_i, s_j) \\ \text{Diagonal} \begin{cases} E_{\text{Constraint}}(t, s_{\text{start}}, s_{\text{end}}, a_i) \\ V_{\text{Constraint}}(t_{+1}, s_{\text{mid}}, a_j) \end{cases} \end{cases}$
    (In diagonal conflicts: a_i moves, a_j blocks)
    $N' \leftarrow \text{replan}(a_k, \mathcal{C}_{a_k})$ ▷ 8-connected WA*
    if $N' \neq \emptyset$ **then** OPEN.push(N')
end while
return *failure*

IV. Experimental Evaluation and Analysis

A. Experimental Design and Algorithm Combinations

To comprehensively evaluate planner performance under a spectrum of challenges, we designed three indoor scenarios in the Isaac Sim platform, ordered by decreasing environmental complexity and interaction density.

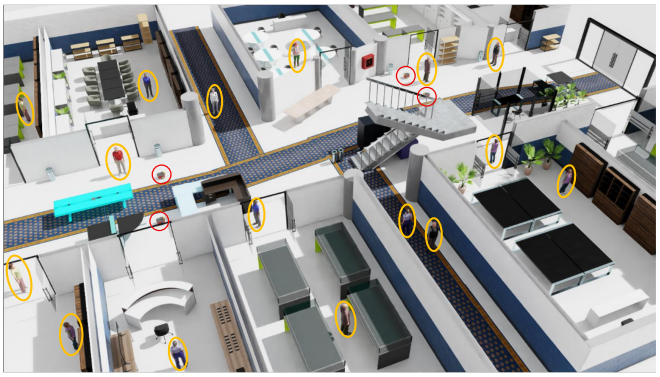


Fig. 3. The shopping mall scenario contains wide-open spaces and multiple dynamic pedestrians, designed to evaluate planner performance under high-density interference. Mobile robots and pedestrians are highlighted by red and yellow circles, respectively.

(1) Shopping Mall: A Large-Scale Commercial Thoroughfare with Dense Crowd Flow. This 40×40 m^2 scenario is designed as a large commercial space centered on a wide main thoroughfare, which is flanked by various enclosed shops. The environment is populated with decorative elements, seating areas, and stalls that create complex static obstacles. Notably, over ten dynamic pedestrians move primarily along the main thoroughfare, simulating a bustling and unpredictable commercial environment (Fig. 3). The primary challenge in this scenario is to maintain planning robustness and efficient coordination in the face of intense, stochastic crowd interference.

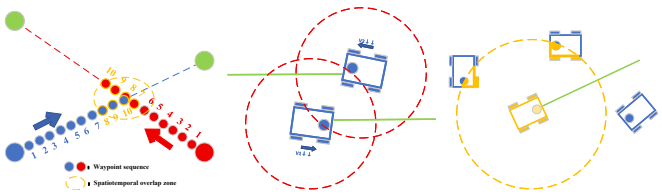
(2) Library: A Testbed for Navigation in Constrained Environments. The 25×25 m^2 library scenario features a structured grid of narrow aisles with adjacent open reading and reception areas (Fig. 4a). This constrained grid is intentionally designed to channelize all robot and pedestrian traffic, creating frequent and unavoidable head-on or overtaking interactions. The primary challenge posed by this environment is to ensure smooth traffic flow and high passage rates, while resolving potential deadlocks within these narrow corridors.

(3) Office: A Semi-Structured, Moderately Complex Scenario. The final 30×30 m^2 scenario depicts a prototypical open-plan office with expansive navigable spaces interspersed with distinct obstacle clusters, including meeting rooms, casual seating areas, and office areas (Fig. 4b). The static obstacles feature an ordered arrangement with moderate spacing, forming a less constrained environment. With only 5 pedestrians, dynamic interference remains minimal compared to other scenarios. This setting evaluates the core efficiency and adaptability of a planner in scenarios with limited dynamic disturbances.

Isaac Sim offers high-fidelity physical modeling and GPU-accelerated simulation, supporting realistic interactions among multiple robots and dynamic entities. In our experiments featuring four homogeneous robots interacting with pedestrians, pedestrian behaviors are generated using a local obstacle-avoidance-based policy, with fixed random seeds applied to ensure reproducibility and fairness. All simulations are

(a) Library **(b)** Office

Fig. 4. (a) The library scenario features a structured layout with dense shelves and corridor-based pedestrian flows, suitable for evaluating execution stability in constrained environments. (b) The Office scene presents a loosely organized, obstacle-rich layout with sparse pedestrian activity, used to assess planning adaptability under moderate complexity.



(a) STO **(b)** IBTR **(c)** LDF

Fig. 5. Visualization of Interaction-Level Metrics. (a) STO identifies planned conflict between two paths: numbered waypoints mark path sequences, and a dashed ellipse denotes the spatiotemporal overlap zone of red waypoints 7–10 and blue 8–10 (within ±2-waypoint window). (b) IBTR illustrates a congestion event where two robots mutually obstruct each other within a distance threshold, with speed indicators showing reduced velocities. (c) LDF presents the first step of its calculation: counting the number of neighboring robots within a defined radius.

executed on an NVIDIA GeForce RTX 4070 Laptop GPU (8GB VRAM) with 16GB RAM.

The system adopts a layered navigation architecture that integrates four global planners (CBS, EECBS, SIPP, and ECBS-8) and four local controllers (DWA, TEB, MPC, and E3MoP), resulting in 16 planner-controller combinations tested 20 times across three fixed scenarios. This structured design enables a comprehensive and fair comparison of planning and control performance. To ensure statistical rigor, trials with goal failures or persistent oscillations were excluded, and outliers beyond two standard deviations from the mean were removed before averaging. These failed trials are reflected in the success rates in Table III, which quantify the robustness of each combination. To ensure reproducibility, the key parameter settings for all planners are provided in the Appendix. All local planners were configured with consistent velocity and acceleration limits to enable fair comparison.

B. Evaluation Metrics: System Outcomes and Interaction

In our benchmark, we propose a multi-faceted evaluation methodology that assesses planner performance across three levels: from macroscopic system-level outcomes, to fine-grained inter-robot interactions, and finally to the underlying robot-level behaviors that cause these phenomena.

1) Macroscopic System-Level Outcomes: The first level of assessment focuses on the overall task efficiency of the system as a whole. We adopt a set of established metrics to provide a macroscopic benchmark of performance, including total path length, planning time, task completion time, and average speed [6].

2) Interaction-Level Metrics: To elucidate the internal dynamics masked by these macroscopic outcomes, we introduce a set of novel metrics specifically designed to quantify the quality of dynamic coordination among robots. These metrics are central to understanding the internal state of the system and include:

1) **Spatiotemporal Overlap (STO):** To evaluate the guidance quality and inherent safety of the initial global paths, we propose the Spatiotemporal Overlap (STO) metric. As a static metric, it analyzes the planned paths themselves to predict potential conflicts before execution begins. The STO for an individual robot is calculated as the ratio of its waypoints that are in potential conflict. A waypoint is considered to be in conflict if the path of any other robot is scheduled to pass within a distance threshold, D_{STO}, during a temporal window of $\pm k$ waypoints. The final system-wide STO presented in our results is the average of the STO values from all individual robots. STO thus serves as a crucial indicator of the coordination quality and safety of a multi-robot plan, particularly in complex and dense environments.
2) **Interaction Blocking Time Ratio (IBTR):** This metric provides a normalized measure of the temporal cost arising directly from traffic congestion. We define a "Congestion Event" as a continuous period where at least two robots mutually obstruct each other, identified when their speeds both fall below a threshold v_{thresh} while within a proximity of d_{thresh}. The event ends once this condition is no longer met for the involved robots. The final IBTR is the total duration of all "Congestion Events" divided by the total mission time, T. This ratio quantifies the fraction of time the system spent in a congested state.
3) **Local Density Fluctuation (LDF):** This metric assesses the stability of local traffic flow by quantifying its temporal variability. The LDF for an individual robot is computed in two main steps: **(1)** a time series of neighbor counts, $N(t)$, is generated by sampling the number of other robots within a radius R_{LDF} at discrete time intervals Δt; **(2)** the temporal variance of this series is calculated, where

$$\text{LDF} = \text{Var}(N) = \frac{1}{n}\sum_{i=1}^{n}(N(t_i) - \overline{N})^2.$$

To capture the worst-case instability, the final system-wide LDF is then determined by the maximum value of all individual LDFs. A high LDF score indicates significant instability in traffic flow, revealing weaknesses in coordination.

To enable a deeper performance analysis, our methodology aims to quantitatively reveal the relationship between execution behavior of a planner and its resulting impact on both interaction-level and system-level outcomes. The primary metric for quantifying execution behavior is Path Following

TABLE II

COMPARISON OF GLOBAL PLANNERS IN THREE SCENARIOS

Scenario	Planner	Path Length (m)	Planning Time (ms)	STO
Shopping	ECBS-8	**59.33**	15	0.036
	CBS	63.02	**3**	0
	SIPP	63.26	16	0.031
	EECBS	63.02	7	0
Library	ECBS-8	**74.96**	101	0
	CBS	79.72	**7**	0
	SIPP	79.78	49	0
	EECBS	79.72	10	0
Office	ECBS-8	**32.95**	33	0
	CBS	42.80	**10**	0
	SIPP	39.15	17	0
	EECBS	42.80	18	0

Deviation (PFD), defined as the time-averaged Euclidean distance between a robot's actual position and the closest point on its reference global path at each control timestep. By systematically correlating PFD values with the previously defined outcome metrics (e.g., IBTR, LDF, and Makespan), our benchmark provides a quantitative basis for assessing how local execution fidelity influences overall system performance and coordination. This is because a standalone PFD value is ambiguous; the outcome metrics provide the necessary context to interpret whether a deviation was strategically effective or detrimental.

In our experimental evaluation, the specific parameters for these novel interaction metrics were configured as follows. For Spatiotemporal Overlap (STO), the proximity threshold D_{STO} was set to 0.5 meters, and the temporal window radius k was 2 waypoints. For Interaction Blocking Time Ratio (IBTR), a low-velocity threshold of v_{thresh} was defined as 20% of the robot's average linear velocity, and the interaction distance d_{thresh} was 2.0 meters. Finally, for Local Density Fluctuation (LDF), the neighborhood radius R_{LDF} was set to 2.0 meters, with the system state sampled at a time interval of $\Delta t = 1.0$ second. These values were chosen empirically to best reflect the robot dimensions and interaction dynamics in our tested scenarios.

These values were determined based on robot dimensions, motion capabilities, and scenario configurations, to align with our experiments and enable consistent comparative analysis. We note that these thresholds are sensitive to our specific test conditions and may not apply to scenarios with drastically different robot scales or dynamics. Future work will explore a generalized threshold-setting method for diverse environments

C. Global Path Planner Performance Analysis

We evaluated four global planners (CBS, SIPP, EECBS, and ECBS-8) across the three benchmark scenarios, comparing their performance in terms of path length, spatiotemporal overlap, and planning time. Results are shown in Table II.

Our analysis of global planners revealed distinct performance profiles, highlighting a clear trade-off between path optimality, safety, and computation time. Both CBS and EECBS consistently generated paths with zero spatiotemporal overlap (STO=0), demonstrating their effectiveness in producing conflict-free plans. In contrast, while generating the most path-efficient trajectories, ECBS-8 introduced the highest overlap risk (STO=0.036 in the mall scenario), which is attributable to its search encouraging more direct paths. Similarly, SIPP also exhibited a non-zero STO. Regarding planning time, CBS, SIPP, and EECBS were efficient, while ECBS-8 incurred the highest computational cost.

Given these findings, we selected CBS and ECBS-8 for subsequent experiments as they represent two representative strategies: one prioritizing safety and predictability, the other prioritizing path length optimality.

TABLE III

COMPARISON OF LOCAL CONTROLLERS IN THE SHOPPING MALL SCENARIO

Controller	Suc (%)	L (m)	T (s)	$\bar{V}$ (m/s)	σ_V	IBTR (%)	LDF	PFD (m)
ECBS-8+MPC	91.5	59.74	46.8	0.45	0.219	0	0.074	0.056
ECBS-8+TEB	89.2	**59.65**	**43.0**	0.43	0.180	0	0.036	0.075
ECBS-8+DWA	95.3	59.98	46.0	0.42	**0.148**	0	0.117	0.039
ECBS-8+E3MoP	87.7	59.64	45.0	0.43	0.150	0	0.045	0.063
CBS+MPC	**97.3**	65.12	44.4	**0.52**	0.253	0	0	**0.012**
CBS+TEB	95.4	62.75	45.4	0.41	0.207	2.6	0	0.034
CBS+DWA	97.0	62.89	46.6	0.41	0.190	0	0	0.056
CBS+E3MoP	93.8	62.76	48.5	0.39	0.181	3.5	0	0.021

D. Local Controller Performance Analysis

Based on the CBS and ECBS-8 global paths, we evaluate the performance of four local controllers (DWA, TEB, MPC, and E3MoP) in the shopping mall scenario, focusing on execution efficiency and interaction behavior. The results presented in Table III are averaged over 20 trials, where *Suc.* denotes the task success rate, *L* is the odometry-based path length (m), and *T* is the completion time (s). Our novel interaction and fidelity metrics (IBTR, LDF, PFD) are calculated as defined in the preceding section.

As shown by the results in Table III, global path selection reveals a trade-off between robustness and trajectory efficiency. Compared to structured CBS paths, the aggressive, optimized ECBS-8 paths, while significantly shortening the actual executed trajectories, simultaneously pose a greater challenge to local planners, with the success rate dropping from 97% to 91%. Furthermore, in purely reactive navigation, sampling-based DWA and model-predictive MPC exhibit significantly greater resilience than optimization-based planners like TEB and E3MoP. This is likely because the former prioritize rapid, adaptive responses, whereas the latter's optimization process is more susceptible to failures when faced with abrupt environmental changes.

Regarding interaction performance, our novel metrics reveal the varying coordination effects demonstrated by different planner combinations.The path optimization of ECBS-8 leads to higher local robot density and thus significant LDF; however, with a limited number of agents in the scenario, this instability does not escalate into mutual gridlock, resulting in zero IBTR. Conversely, the structured nature of CBS paths minimizes LDF by design, but can create unavoidable geometric choke points. To safely navigate these

critical sections, planners like TEB and E3MoP often must decelerate, leading to brief mutual obstructions—an effect reflected in our IBTR metric. This demonstrates that different planner combinations lead to distinct interaction challenges, ranging from traffic instability (measured by LDF) to mutual blockages (measured by IBTR). A comprehensive benchmark must therefore be able to distinguish between these varied interaction patterns.

The joint analysis of PFD with these interaction metrics provides a deeper understanding of these behaviors. For example, under the challenging ECBS-8 plan, TEB pairs the highest PFD with the lowest LDF. This suggests its frequent path deviations were well-managed and resulted in stable, fluid interactions. Conversely, the rigid path adherence of DWA (low PFD) is coupled with the highest LDF, indicating that a lack of adaptive deviation can be detrimental to system-wide stability. This analysis exemplifies how our benchmark provides a quantitative assessment of how local execution fidelity directly impacts system-level coordination.

V. CONCLUSIONS AND FUTURE WORK

This paper introduced **DMRP-Bench**, an integrated and unified benchmark framework for the systematic evaluation of multi-robot motion planning systems in dynamic, high-fidelity environments. Through extensive experiments combining multiple global and local planners, we demonstrated its effectiveness in capturing a wide spectrum of performance characteristics, from macroscopic efficiency to fine-grained interaction dynamics. Furthermore, the modular architecture and standardized interfaces of the framework ensure both reproducibility and extensibility for future research.

Our results revealed clear trade-offs at the global planning level, where path-length-optimized planners like ECBS-8, while efficient, tend to induce greater instability (LDF) in local traffic. Conversely, more structured planners like CBS ensure smoother flow but can introduce susceptibility to brief, geometric blockages (IBTR). More importantly, the benchmark's core capability was demonstrated in the analysis of local controllers. By performing a joint analysis of our novel interaction metrics with path execution fidelity (PFD), our methodology provides a quantitative basis for assessing how different local planning strategies, ranging from adaptive path deviation to rigid path adherence, directly impact system-wide coordination and stability. This moves evaluation beyond simple scoring to a deeper, more insightful assessment of planner behavior.

In future work, we plan to extend the benchmark toward larger-scale systems, more complex dynamic interaction scenarios with socially-aware pedestrian models, and higher real-time requirements, as well as to develop more indicative coordination-oriented metrics. These enhancements aim to further improve framework applicability and to support the development and deployment of collaborative navigation algorithms.

REFERENCES

[1] A. Bolu and Ö. Korçak, "Adaptive Task Planning for Multi-Robot Smart Warehouse," *IEEE Access*, vol. 9, pp. 27346-27358, 2021.

[2] Y. Liu, F. Ma, and G. Li, "A Coordinated Path Planning Algorithm for Multi-Robot in Intelligent Warehouse," in *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2019, pp. 2945-2950.

[3] D. Pereira, D. Matos, P. Rebelo, F. Ribeiro, P. Costa, and J. Lima, "Multi-robot coordination for a heterogeneous fleet of robots," in *Iberian Robotics Conference*, Cham: Springer International Publishing, 2022, pp. 229-240.

[4] J. A. Gonzalez-Aguirre, R. Osorio-Oliveros, K. L. Rodríguez-Hernández, J. Lizárraga-Iturralde, R. Morales Menendez, R. A. Ramirez-Mendoza, *et al.*, "Service robots: Trends and technology," *Appl. Sci.*, vol. 11, no. 22, p. 10702, 2021.

[5] B. R. Nair, "Collaborative perception in multi-robot systems: Case studies in household cleaning and warehouse operations," in *2024 6th International Conference on Robotics and Computer Vision (ICRCV)*, 2024, pp. 195-200.

[6] P. Zheng, H. Wang, Z. Sang, R. Y. Zhong, Y. Liu, C. Liu, *et al.*, "Smart manufacturing systems for Industry 4.0: Conceptual framework, scenarios, and future perspectives," *Front. Mech. Eng.*, vol. 13, no. 2, pp. 137–150, 2018.

[7] K. L. Keung, C. K. M. Lee, and P. Ji, "A cyber-physical robotic mobile fulfillment system in smart manufacturing: The simulation aspect," *Robotics and Computer-Integrated Manufacturing*, vol. 83, p. 102578, Oct. 2023.

[8] Z. Zhao, X. Li, S. Liu, M. Zhou, and X. Yang, "Multi-mobile-robot transport and production integrated system optimization," *IEEE Trans. Autom. Sci. Eng.*, 2024.

[9] G. Kyprianou, L. Doitsidis, and S. A. Chatzichristofis, "Towards the achievement of path planning with multi-robot systems in dynamic environments," *J. Intell. Robot. Syst.*, vol. 104, no. 1, p. 15, 2022.

[10] S. Schaefer, L. Palmieri, L. Heuer, R. Dillmann, S. Koenig, and A. Kleiner, "A benchmark for multi-robot planning in realistic, complex and cluttered environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2023, pp. 9231–9237.

[11] R. Stern, N. Sturtevant, A. Felner, S. Koenig, H. Ma, T. Walker, J. Li, D. Atzmon, L. Cohen, T. Kumar, E. Boyarski, and R. Barták, "Multi-agent pathfinding: Definitions, variants, and benchmarks," in *Proc. Int. Symp. Combinatorial Search (SoCS)*, vol. 10, no. 1, pp. 151–158, 2019.

[12] C. Chamzas, C. Quintero-Pena, Z. Kingston, A. Orthey, D. Rakita, M. Gleicher, M. Toussaint, and L. E. Kavraki, "MotionBenchMaker: A tool to generate and benchmark motion planning datasets," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 882–889, 2021.

[13] M. Gebser, P. Obermeier, T. Otto, T. Schaub, O. Sabuncu, V. Nguyen, and T. C. Son, "Experimenting with robotic intra-logistics domains," *Theory Pract. Log. Program.*, vol. 18, no. 3–4, pp. 502–519, 2018.

[14] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, 2004,vol. 3, pp. 2149–2154.

[15] "Welcome to Flatland," [Online]. Available: `https://flatland.aicrowd.com/intro.html`, accessed: Mar. 1, 2025.

[16] J. Yu, "mapf_ros: A multi-agent pathfinding framework," [Online software]. Available: `https://github.com/speedzjy/mapf_ros`, accessed: Mar 1, 2025.

[17] J. Wen, X. Zhang, Q. Bi, Z. Pan, Y. Feng, J. Yuan, and Y. Fang, "MRPB 1.0: A unified benchmark for the evaluation of mobile robot local planning approaches," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2021, pp. 8230–8237.

[18] Open-RMF, "RMF demos," [Online software]. Available: `https://github.com/open-rmf/rmf_demos`, accessed: May 1, 2025.

[19] NVIDIA, "NVIDIA ISAAC SIM," [Online]. Available: `https://developer.nvidia.com/isaac-sim`, accessed: March 1, 2025.

[20] G. Sharon, R. Stern, A. Felner, N. R. Sturtevant, "Conflict-based search for optimal multi-agent pathfinding," *Artif. Intell.*, vol. 219, pp. 40–66, 2015.

[21] G. Wagner and H. Choset, "Subdimensional expansion for multirobot path planning," *Artif. Intell.*, vol. 219, pp. 1–24, 2015.

[22] M. Barer, G. Sharon, R. Stern, and A. Felner, "Suboptimal variants of the conflict-based search algorithm for the multi-agent pathfinding problem," in *Proc. Int. Symp. Combinatorial Search (SoCS)*, vol. 5, no. 1, pp. 19–27, 2014.

[23] J. Li, W. Ruml, and S. Koenig, "EECBS: A bounded-suboptimal search for multi-agent path finding," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 14, pp. 12353–12362.

[24] M. Phillips and M. Likhachev, "SIPP: Safe interval path planning for dynamic environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2011, pp. 5628–5635.
[25] J. van den Berg, M. Lin, and D. Manocha, "Reciprocal velocity obstacles for real-time multi-agent navigation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 1928–1935, May 2008.
[26] T. Sahib and M. Ali, "Multi-robot path planning based on multi-objective particle swarm optimization," *IEEE Access*, vol. 7, pp. 2138–2147, 2018.
[27] G. Sartoretti, J. Kerr, Y. Shi, G. Wagner, T. S. Kumar, S. Koenig, and H. Choset, "PRIMAL: Pathfinding via reinforcement and imitation multi-agent learning," *IEEE Robot. Autom. Lett.*, vol. 4, no. 3, pp. 2378–2385, Jul. 2019.
[28] ROS Navigation Stack, [Online software]. Available: `https://github.com/ros-planning/navigation`, accessed: March 1, 2025.
[29] Y.-W. Yeh, W.-C. Wang, and R. Chen, "Collision-free navigation for multiple robots in dynamic environment," in *Proc. IEEE/ASME Int. Conf. Mechatronic Embedded Syst. Appl. (MESA)*, pp. 107–112, Nov. 2022.
[30] S. Salimpour, J. Peña-Queralta, D. Paez-Granados, J. Heikkonen, and T. Westerlund, "Sim-to-real transfer for mobile robots with reinforcement learning: From NVIDIA Isaac Sim to Gazebo and real ROS 2 robots," *arXiv preprint arXiv:2501.02902*, 2025.
[31] C. Rösmann, W. Feiten, T. Wöesch, F. Hoffmann, and T. Bertram, "Trajectory modification considering dynamic constraints of autonomous robots," in *Proc. ROBOTIK 2012; 7th German Conference on Robotics*, 2012. pp. 74–79.
[32] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robot. Autom. Mag.*, vol. 4, no. 1, pp. 23–33, 2002.
[33] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal Trajectory Generation for Dynamic Street Scenarios in a Frenet Frame," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, pp. 987–993, 2010.
[34] J. Wen, X. Zhang, H. Gao, J. Yuan, and Y. Fang, "E³MoP: Efficient motion planning based on heuristic-guided motion primitives pruning and path optimization with sparse-banded structure," *IEEE Trans. Autom. Sci. Eng.*, vol. 19, no. 4, pp. 2762-2775, 2022.
[35] J. Yan, Z. Li, W. Kang, *et al.*, "Advancing MAPF towards the real world: A scalable multi-agent realistic testbed (SMART)," *arXiv preprint arXiv:2503.04798*, 2025.
[36] L. Heuer, L. Palmieri, A. Mannucci, *et al.*, "Benchmarking Multi-robot coordination in realistic, unstructured human-shared environments," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 14541-14547.

APPENDIX

Table IV lists the key parameters for each planner in our experiments. To ensure fair comparison, all local planners were configured with the same velocity and acceleration limits: max_vel_x = 0.8 m/s, max_vel_theta = 1.0 rad/s, acc_lim_x = 1.0 m/s^2, and acc_lim_theta = 1.0 rad/s^2.

TABLE IV
KEY PLANNER PARAMETERS

Planner	**Parameter**	**Value**
TEB	weight_obstacle	50
DWA	path_distance_bias	5.0
MPC	objective_type	minimum_time
E3MoP	weight_obstacle	50
ECBS-8	suboptimality_bound	2.0
CBS	–	(default)