Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Safe and Efficient Robot Navigation via Costmap-Guided Multi-Objective A*

hand ouelmokhtar

hand.ouelmokhtar@ummto.dz

Mouloud Mammeri University of Tizi-Ouzou

samira oussidhoum

Mouloud Mammeri University of Tizi-Ouzou

hocine belgaid

Mouloud Mammeri University of Tizi-Ouzou

zaina ait-chekdhidh

Mouloud Mammeri University of Tizi-Ouzou

yahia benmoussa

University of Boumerdes

Research Article

Keywords: Path planning, multi-objective optimization, safety costmaps, Pareto-optimal solutions, autonomous navigation

Posted Date: March 17th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8521606/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at International Journal of Intelligent Robotics and Applications on August 8th, 2026. See the published version at https://doi.org/10.1007/s41315-026-00574-3.

Safe and Efficient Robot Navigation via Costmap-Guided Multi-Objective A*

Hand OUELMOKHTAR[1*], Samira OUSSIDHOUM[2†], Hocine BELGAID[3†], Zaina AIT-CHEKDHIDH[4†], Yahia BENMOUSSA[5†]

[1]Lab: LMSS/UMBB , Dpt: Electromechanics, University Mouloud Mammeri, Tizi-ouzou, Algeria .
[2]Lab: LMSE/UMMTO , Dpt: Electromechanics, University Mouloud Mammeri, Tizi-ouzou, Algeria .
[3]Lab: LATAGE/UMMTO , Dpt: Electromechanics, University Mouloud Mammeri, Tizi-ouzou, Algeria .
[4]Lab: L2CSP/UMMTO , Dpt: Automatics, University Mouloud Mammeri, Tizi-ouzou, Algeria .
[5]Lab: LMSS/UMBB, Dpt: Electronics, University M'Hamed Bougara, Boumerdes, Algeria.

*Corresponding author(s). E-mail(s): hand.ouelmokhtar@ummto.dz;
Contributing authors: samira.oussidhoum@ummto.dz; hocine.belgaid@ummto.dz; zaina.ait-chekdhidh@ummto.dz; y.benmoussa@univ-boumerdes.dz;
†These authors contributed equally to this work.

Abstract

Autonomous mobile robots must balance path efficiency and safety when navigating cluttered environments. This paper proposes Costmap-MOA*, a multi-objective A* planner that couples a continuous obstacle-aware costmap with local Pareto front maintenance. The costmap fuses terrain relief (obstacle distance) and goal-directed turbulence to model both safety and directional guidance. The Costmap-MOA* algorithm simultaneously optimizes path length and accumulated safety cost, extending classical A* by maintaining non-dominated solutions at each node via lexicographic ordering of distance and risk. This design preserves shortest-path optimality while systematically exploring distance–safety trade-offs and avoiding redundant trajectories.

Experiments on sparse and cluttered 2D environments demonstrate that Costmap-MOA* achieves 2–4× more non-dominated solutions than NAMOA*, improves best-safety cost by 25–40%, and maintains comparable computation times. The method provides decision-makers with a diverse set of feasible trajectory alternatives, making it a practical tool for safety-critical autonomous navigation.

Keywords: Path planning, multi-objective optimization, safety costmaps, Pareto-optimal solutions, autonomous navigation

1 Introduction

Autonomous mobile robots must plan safe, efficient paths in structured environments while minimizing computational overhead. A* guarantees shortest-path optimality but ignores safety margins [1]; Artificial Potential Fields provide safety through continuous cost fields but risk local minima [2]; and multi-objective search algorithms like NAMOA* enable Pareto-optimal trade-offs but typically use discrete cost models, not continuous safety representations [3]. Safety-critical applications such as hospital logistics and warehouse automation require planners that can adapt clearance margins dynamically based on operational context (e.g., human proximity, fragile cargo), motivating the need for diverse Pareto-optimal solutions rather than a single scalar-optimized path.

This paper proposes **Costmap-MOA***, a multi-objective A* planner that couples a continuous obstacle-aware costmap with local Pareto-front maintenance to generate distance-optimal paths together with diverse, safety-aware alternatives in cluttered 2D environments. The overall processing pipeline is summarized in Fig. 1, which highlights the stages from environment modeling to multi-objective search and Pareto-front extraction.

1.1 Contributions

We propose four main contributions, aligned with the pipeline in Fig. 1:

1. **Environment and Costmap Modeling:** A continuous, obstacle-aware costmap that fuses terrain relief (distance to obstacles) and goal-directed turbulence, enabling tunable clearance around obstacles:

$$\text{costmap} = \alpha \cdot \text{relief} + \beta \cdot \text{turbulence}.$$

2. **Costmap-MOA* Search Formulation:** Integration of the costmap into a bi-objective A* framework that jointly optimizes path length g_d and accumulated safety cost g_s using a weighted evaluation:

$$f = w_d(g_d + h_d) + w_s g_s.$$

3. **Local Pareto-Front Construction:** Maintenance of non-dominated cost vectors at each node, including dominance checks and neighborhood expansion, to systematically explore distance–safety trade-offs and build the global Pareto front at the goal.
4. **Safe, Diverse Trajectory Generation:** A post-processing stage that reconstructs paths from the goal's Pareto front, removes duplicates, and returns a compact set of representative trajectories, providing multiple safety-aware alternatives instead of a single path.

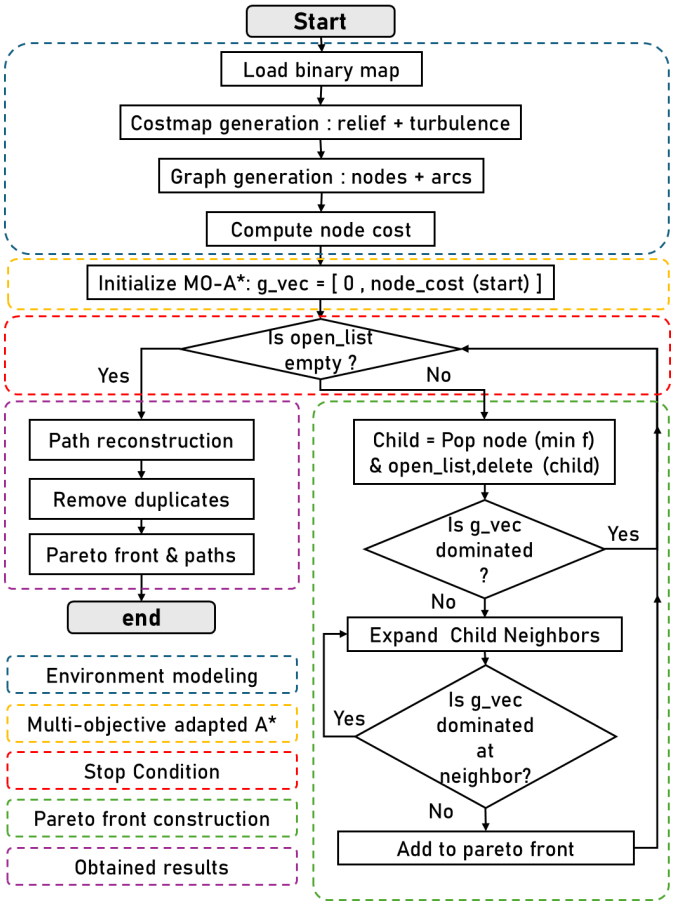


Fig. 1 Flowchart of the proposed Costmap-MOA* framework

Experiments on sparse and cluttered maps show that Costmap-MOA* yields 2–4× more non-dominated solutions than NAMOA*, achieves 25–40% lower best-safety costs, and preserves shortest-path optimality, making it suitable for safety-critical robotics where multiple trajectory alternatives are valuable.

The remainder of this paper is organized as follows. Section 2 reviews related work on potential-field-based planning, A* hybridization, and multi-objective extensions of A*. Section 3 details the Costmap-MOA* framework, including costmap construction, multi-objective search, and Pareto-front management. Section 4 reports simulation results and compares Costmap-MOA* to NAMOA* on multiple benchmark scenarios. Finally, Section 5 concludes the paper and outlines directions for future research.

2 Related Works

Trajectory planning for autonomous robots must balance path efficiency with safety, often under tight computational constraints. Existing approaches can be grouped into continuous safety fields, practical A*-based planners for robotics, and multi-objective search methods that explicitly construct Pareto-optimal trade-offs. This section reviews these three lines and motivates the need for a costmap-guided multi-objective A* such as our Costmap-MOA*.

2.1 Safety Fields and Costmaps

Artificial Potential Fields (APF) model navigation as the superposition of attractive forces toward the goal and repulsive forces from obstacles, enabling real-time obstacle avoidance and smooth motion [2]. Subsequent variants modify the potential formulation to reduce trapping and improve escape from local minima, for example by introducing lateral forces or adaptive step sizes [4]. Other works move from forces to continuous cost fields, designing APF/costmap formulations specifically to prevent deadlocks and enhance robustness in cluttered environments [5]. More recent APF-based planners extend these ideas to dynamic scenes, improving both trajectory tracking and obstacle avoidance in the presence of moving obstacles [6]. Overall, APF and costmap approaches provide intuitive continuous safety modeling, but they generally lack guarantees on global optimality or systematic coverage of distance–safety trade-offs.

2.2 A*-Based Planning for Robotics

Classical A* offers completeness and optimality for single-objective shortest paths, but raw grid-based solutions may be tortuous or computationally expensive in large maps. Many robotics planners therefore hybridize A* with local collision avoidance or trajectory smoothing. A family of works fuses improved A* variants with the Dynamic Window Approach (DWA), using A* for global routing and DWA for local obstacle avoidance in mobile robots and AGVs [7–9]. Other methods post-process A* paths using bidirectional shortcuts and perturbations to shorten and smooth trajectories while preserving feasibility [10], or adapt heuristic weights and reward terms online to accelerate convergence [11]. For non-holonomic or multi-robot scenarios, additional trajectory optimization layers are introduced on top of the global graph search to enforce kinematic constraints and coordinate multiple robots [12, 13]. These A*-based approaches improve path quality and runtime, but they still optimize a single scalar objective and thus do not provide explicit Pareto fronts for distance–safety trade-offs.

2.3 Multi-Objective Search and Applications

Extending A* to multi-objective optimization has led to algorithms that maintain non-dominated cost labels and prune dominated paths via Pareto dominance. NAMOA* is a seminal contribution: it generalizes A* to multiple objectives, introduces improved pruning strategies, and provides admissibility and completeness guarantees for the Pareto-optimal set under consistent vector heuristics [3, 14]. To reduce computational burden, NAMOA*$_{\text{dr}}$ applies dimensionality reduction [15], while BOA*-type algorithms exploit bi-objective structure for faster dominance checks [16, 17]. Parallel variants further accelerate search [18], and scalarization strategies such as weighted maximization improve Pareto coverage in non-convex regions [19].

Applications of NAMOA* and multi-objective A* address diverse objective pairs: *distance and energy* on uneven terrains [20], *distance and smoothness* for trajectory quality [21], and *path length and safety* in hazardous or dynamic environments [22]. However, most works model safety or risk using discrete node costs (e.g., clearance computed from a binary occupancy grid) rather than continuous safety fields. Costmap-MOA* addresses this gap by integrating a continuous obstacle-proximity costmap directly into multi-objective A*, enabling richer exploration of the distance–safety trade-off in cluttered 2D environments.

2.4 Positioning of Costmap-MOA*

In summary, APF and costmap-based planners offer continuous safety modeling but lack Pareto-complete guarantees; A*-based robotics planners focus on single-objective optimality and smoothing; and multi-objective A* algorithms like NAMOA* provide theoretical guarantees but are mostly studied with abstract cost vectors, not with continuous safety fields. Our Costmap-MOA* bridges these gaps by coupling a continuous, risk-aware costmap with a multi-objective A* that maintains local Pareto fronts, thereby preserving shortest-path optimality while explicitly generating a diverse set of distance–safety trade-offs suitable for safety-critical navigation.

3 Methodology

This section describes the proposed trajectory-planning methodology in detail. First, the environment representation and cost definitions are introduced, followed by the discretization of the navigation environment. Then, the employed multi-objective search algorithm and its heuristic design are presented.

3.1 Environment Modeling and Costmap Construction

The environment is first represented as a binary or grayscale map where white indicates free space and black corresponds to obstacles. This map may be provided directly or dynamically reconstructed via SLAM. To support both efficient and risk-averse trajectory planning, we transform the binary map into a continuous-valued costmap $C(x, y)$, combining two main layers:

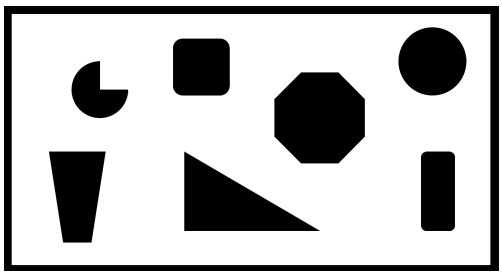


Fig. 2 Binary or grayscale map of the environment. White: free space, black: obstacles.

- **Terrain Relief** (C_{relief}): This layer encodes large-scale features such as hills and valleys using a sum of Gaussian functions:

$$G(x, y) = A \cdot \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2}{2\sigma^2}\right),$$

where A sets the amplitude, (x_0, y_0) centers the feature, and σ controls width.

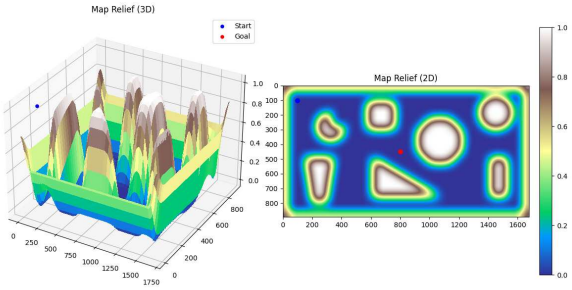


Fig. 3 Terrain relief derived from the map by summing Gaussian features.

- **Goal-Induced Turbulence** ($C_{\text{turbulence}}$): Directional cues are provided using a sinusoidal function radiating from the goal:

$$T(x, y) = \gamma \cdot \sin\left(\omega \cdot \|(x, y) - (x_g, y_g)\|\right),$$

where γ and ω are amplitude and frequency, and (x_g, y_g) is the goal position.

The composite costmap shown in Figure 5 is given by

$$C(x, y) = \alpha C_{\text{relief}}(x, y) + \beta C_{\text{turbulence}}(x, y),$$

with weights (α, β) balancing each component. This encodes both static environment difficulty and dynamic guidance. Additionally, obstacle proximity is encoded as a risk measure, biasing the planner towards safer, open regions.

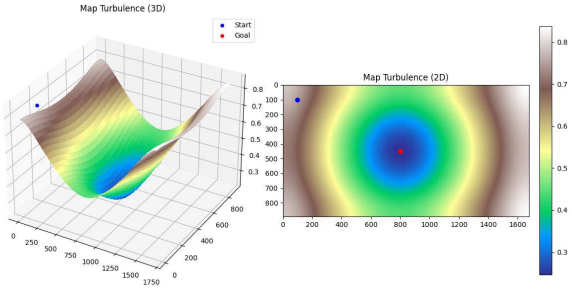


Fig. 4 Goal-induced turbulence as a sinusoidal field radiating from the goal position.

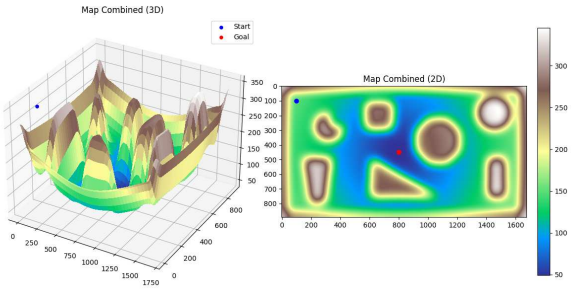


Fig. 5 Combined costmap: fusion of terrain relief and turbulence.

3.2 Grid Sampling and Navigation Graph Construction

The workspace is discretized by sampling the costmap on a regular grid (step size `GRID_SPACING`), retaining only nodes corresponding to free cells:

$$(x, y) \in \mathcal{N} \quad \Leftrightarrow \quad \text{free_mask}(x, y) = 1.$$

Each node is assigned a unique index; a lookup table allows constant-time coordinate-to-node mapping. For each node, we extract:

1. Normalized costmap value $\widehat{C}(i)$,
2. Normalized distance-to-obstacle $\widehat{d}_{\text{obst}}(i)$,
3. Normalized local connectivity ($\text{adj_count}_n(i)$): fraction of 8-connected neighboring nodes that are also free.

A multi-criteria cost is then calculated as:

$$\text{node_cost}(i) = W_c\,\widehat{C}(i) + W_o\left[1 - \widehat{d}_{\text{obst}}(i)\right] + W_{\text{conn}}\left[1 - \text{adj_count}_n(i)\right],$$

where $W_c, W_o, W_{\text{conn}}$ are tunable weights. A geometric neighborhood graph $G = (\mathcal{N}, \mathcal{E})$ is constructed by connecting each node to all others within $d_{\max} = \sqrt{2}\,\texttt{GRID_SPACING}$, so long as the connecting segment is entirely collision-free (verified by sampling intermediate points). Figure 6 illustrates the resulting navigation

graph where each node is color-coded according to its normalized cost value, with low-cost nodes (blue) representing safer regions far from obstacles and high-cost nodes (yellow/white) indicating proximity to obstacles.

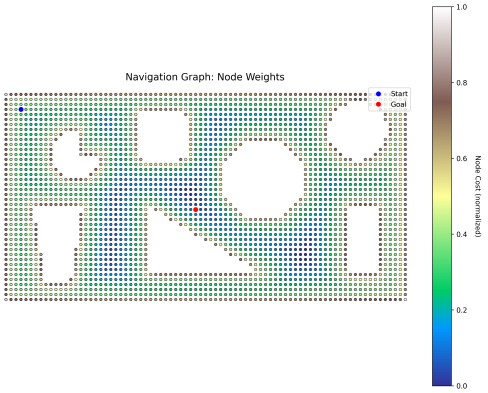


Fig. 6 Navigation graph: each node colored by its local cost.

3.3 Multi-Objective A* Path Planning

Given discrete start and goal nodes, planning seeks to optimize both distance and environmental risk. Each node i maintains a vector

$$\mathbf{g}(i) = \begin{bmatrix} g_{\text{dist}}(i) \\ g_{\text{risk}}(i) \end{bmatrix},$$

where $g_{\text{dist}}(i)$ is total path length and $g_{\text{risk}}(i)$ is cumulative node cost. The heuristic vector is

$$\mathbf{h}(i) = \begin{bmatrix} \|\mathbf{n}_i - \mathbf{n}_{\text{goal}}\|_2 \\ 0 \end{bmatrix},$$

and total cost is $\mathbf{f}(i) = \mathbf{g}(i) + \mathbf{h}(i)$. The queue is lexicographically ordered: nodes with smaller distance are prioritized; ties are broken by risk.

Each node stores a local Pareto front of non-dominated cost vectors. A child is expanded only if its vector is not dominated by previous solutions at that node. Each time the goal is reached, the current path is added to the solution set. The algorithm terminates when the open list is empty, guaranteeing complete Pareto coverage of distance-risk trade-offs. The complete procedure is formalized in Algorithm 1.

4 Results and Discussion

This section presents the simulation results of the proposed Costmap-MOA* algorithm and its comparison with the NAMOA* benchmark on multiple trajectory planning scenarios.

Algorithm 1 Multi-Objective A* (MO-A*)

Require: Graph $G = (V, E)$, start node s, goal node t, node cost $c(\cdot)$, weights w_d, w_s
Ensure: Set of Pareto-optimal paths $\mathcal{P}$ and Pareto front $\mathcal{F}$

$\mathbf{g}_s \leftarrow [0,\ c(s)]$
$f_s \leftarrow w_d \cdot h(s,t) + w_s \cdot c(s)$
OPEN $\leftarrow \{(f_s, s, \mathbf{g}_s, \text{null})\}$
for all $v \in V$ **do**
    ParetoFront$[v] \leftarrow \emptyset$
end for
add $\mathbf{g}_s$ to ParetoFront$[s]$
while OPEN $\neq \emptyset$ **do**
    $(f, v, \mathbf{g}, parent) \leftarrow$ ExtractMin(OPEN)
    let $\mathbf{g} = [g_d, g_s]$
    if $\mathbf{g}$ is dominated in ParetoFront$[v]$ **then**
        continue ▷ Skip dominated label
    end if
    if $v = t$ **then**
        continue ▷ Goal reached; keep exploring other trade-offs
    end if
    for all $u \in \text{Adj}(v)$ **do**
        $\mathbf{g}_u \leftarrow [g_d + d(v,u),\ g_s + c(u)]$
        if $\mathbf{g}_u$ is not dominated in ParetoFront$[u]$ **then**
            remove dominated vectors from ParetoFront$[u]$
            add $(\mathbf{g}_u, v)$ to ParetoFront$[u]$
            let $\mathbf{g}_u = [g_d^u, g_s^u]$
            $f_u \leftarrow w_d \cdot (g_d^u + h(u,t)) + w_s \cdot g_s^u$
            push $(f_u, u, \mathbf{g}_u, v)$ to OPEN
        end if
    end for
end while
$\mathcal{F} \leftarrow$ ParetoFront$[t]$
$\mathcal{P} \leftarrow$ ReconstructPaths$(\mathcal{F}, \text{ParetoFront})$
remove duplicate paths from $\mathcal{P}$
return $(\mathcal{P}, \mathcal{F})$

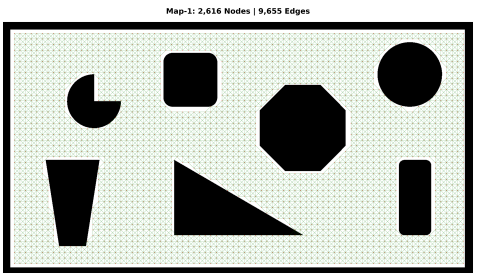


Fig. 7 Sparse environment

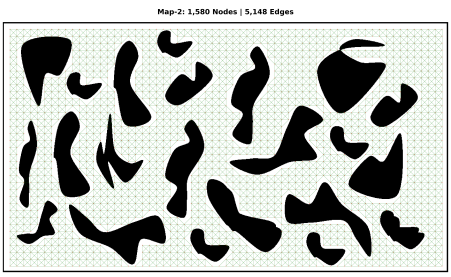


Fig. 8 Cluttered environment

4.1 Experimental Setup and Performance Evaluation

This section evaluates the proposed Costmap-MOA* planner against the NAMOA* benchmark on six start–goal scenarios across two test environments (Fig. 7 & 8). Both methods use identical start–goal configurations and the same occupancy grid for collision checking, but differ in safety modeling: Costmap-MOA* is guided by the proposed continuous costmap (relief + turbulence), whereas NAMOA* assigns node safety from obstacle clearance on the binary grid. Performance is quantified by (i) hypervolume of the normalized Pareto front, (ii) solution-set cardinality, (iii) best safety cost under shortest-path distance, and (iv) runtime.

4.2 Pareto-Front Quality and Trade-off Coverage

Tables 1 and 2 quantify the Pareto-front quality using hypervolume, spread, and spacing. Across all scenarios, Costmap-MOA* achieves higher hypervolume than NAMOA*, indicating a larger dominated region and thus a better coverage of the distance–safety trade-off space. On Map 1, Costmap-MOA* increases hypervolume from 0.187–1.033 (NAMOA*) to 1.638–1.732, while also improving spacing (e.g., 0.054 $\rightarrow$ 0.020 in S1), which reflects a more uniform distribution of solutions along the front (Table 1). On Map 2, the improvement remains consistent (e.g., 0.649 $\rightarrow$ 1.486 in S1 and 0.158 $\rightarrow$ 1.400 in S2), demonstrating that Costmap-MOA* maintains superior Pareto exploration even under tight navigational constraints (Table 2).

Table 1 Map 1 (Sparse): Pareto-front quality indicators

Scenario (Start→Goal)	Hypervolume		Spread		Spacing	
	NA MOA*	Costmap MOA*	NA MOA*	Costmap MOA*	NA MOA*	Costmap MOA*
S1 (100,100)→(800,800)	0.187	1.638	0.510	1.414	0.054	0.020
S2 (100,100)→(1000,600)	0.205	1.673	0.513	1.426	0.020	0.019
S3 (100,100)→(1600,600)	1.033	1.732	1.113	1.208	0.023	0.009

Table 2 Map 2 (Cluttered): Pareto-front quality indicators

Scenario (Start→Goal)	Hypervolume		Spread		Spacing	
	NA MOA*	Costmap MOA*	NA MOA*	Costmap MOA*	NA MOA*	Costmap MOA*
S1 (30,30)→(1200,500)	0.649	1.486	0.930	1.370	0.018	0.022
S2 (30,30)→(700,650)	0.158	1.400	0.475	1.284	0.056	0.028
S3 (30,30)→(1200,700)	0.267	1.617	0.577	1.369	0.012	0.012

4.3 Solution Set Size and Practical Diversity

Beyond quality indicators, Costmap-MOA* generates a substantially larger set of non-dominated solutions (Tables 3 and 4). In the sparse environment, Costmap-MOA* yields 26–58 solutions compared to 6–32 for NAMOA*, providing a wider range of feasible safety margins (Table 3). In the cluttered environment, Costmap-MOA* similarly expands the solution set, reaching up to 77 solutions versus NAMOA*'s maximum of 31 (Table 4). In Figs. 9 and 10, the label "Costmap-MOA*" denotes Costmap-MOA*. Overall, the richer solution set matches the more complete objective-space exploration, where Costmap-MOA* reaches high-safety regions that remain under-explored by NAMOA*.

Table 3 Map 1 (Sparse): number of solutions and runtime

Scenario	Solutions		Time (s)	
	NAMOA*	Costmap-MOA*	NAMOA*	Costmap-MOA*
S1	6	26	13.88	20.04
S2	11	36	9.60	20.04
S3	32	58	20.03	20.07

Table 4 Map 2 (Cluttered): number of solutions and runtime

Scenario	Solutions		Time (s)	
	NAMOA*	Costmap-MOA*	NAMOA*	Costmap-MOA*
S1	31	47	3.28	5.84
S2	6	25	3.16	3.74
S3	25	77	3.34	5.66

4.4 Representative Fronts and Trajectories

Figures 9 and 10 visualize representative normalized Pareto fronts. In Map 1 (Scenario 1), Costmap-MOA* generates 26 non-dominated solutions distributed across the trade-off space, whereas NAMOA* produces only 6 solutions concentrated near the minimum-distance region (Fig. 9). In Map 2 (Scenario 1), Costmap-MOA* again provides broader coverage (47 solutions) compared to NAMOA* (31 solutions), with visibly stronger exploration toward higher-safety regions (Fig. 10).

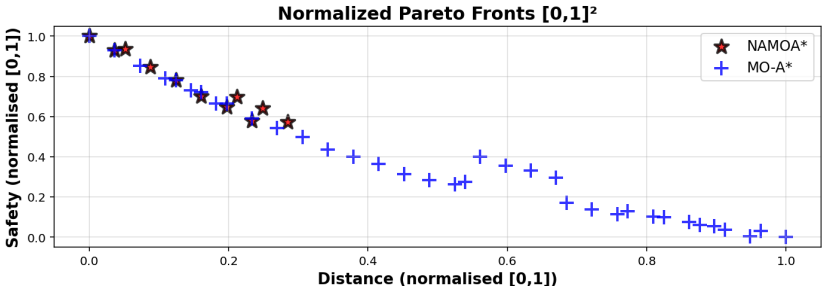


Fig. 9 Pareto front [Map 1: Scenario (100,100)→(800,800)].

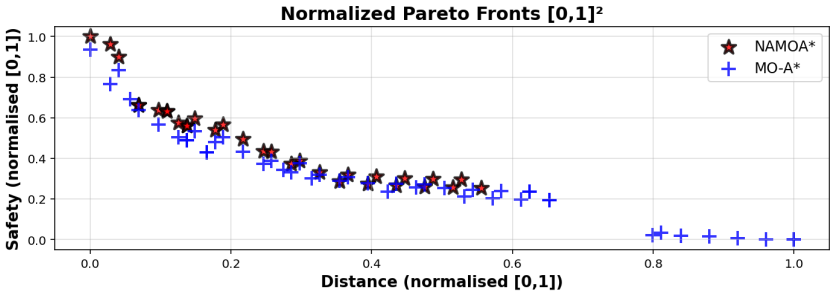


Fig. 10 Pareto front [Map 2: Scenario (30,30)→(1200,500)].

4.5 Shortest-Path Preservation and Safety Gains

A key practical property is that both planners recover the same globally shortest path length (Tables 5 and 6), i.e., Costmap-MOA* preserves distance optimality while augmenting safety trade-offs. However, Costmap-MOA* consistently achieves lower best-safety costs than NAMOA* in all scenarios (e.g., Map 1 S1: 8.91 vs. 14.82; Map 2 S1: 18.57 vs. 24.60), confirming its ability to identify safer routes without sacrificing the distance-optimal solution.

Table 5 Map 1 (Sparse): best distance and best safety costs

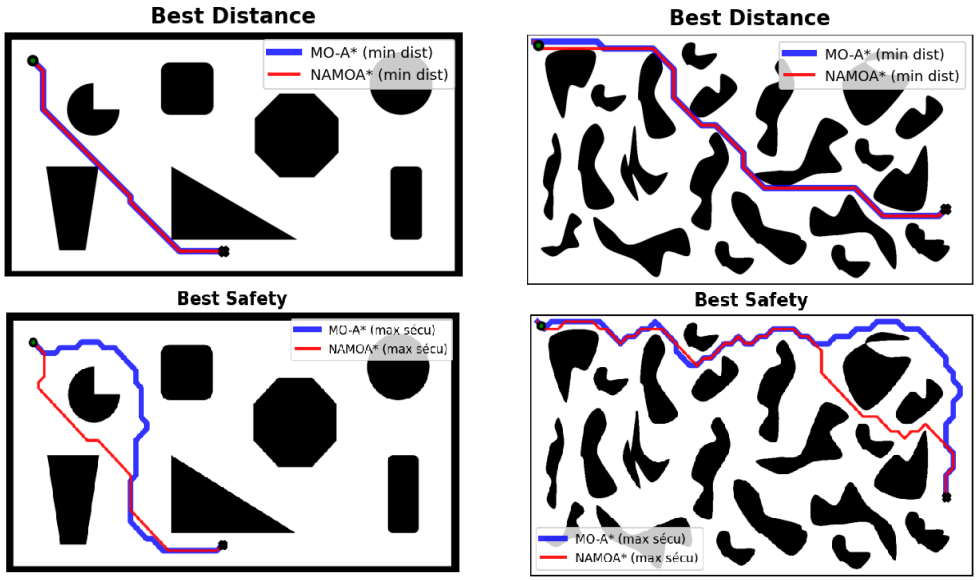
Scenario	Best distance		Best safety	
	NAMOA*	Costmap-MOA*	NAMOA*	Costmap-MOA*
S1	989.3	989.3	14.82	8.91
S2	502.1	502.1	12.18	7.28
S3	1541.7	1541.7	15.32	10.17

The qualitative trajectories in Fig. 11 corroborate the quantitative results by contrasting the distance-optimal (top row) and safety-optimal (bottom row) solutions on both Map 1 (Fig. 11.a) and Map 2 (Fig. 11.b). While the best-distance paths of

Table 6 Map 2 (Cluttered): best distance and best safety costs

Scenario	Best distance		Best safety	
	NAMOA*	Costmap-MOA*	NAMOA*	Costmap-MOA*
S1	1236.8	1236.8	24.60	18.57
S2	946.2	946.2	24.73	19.33
S3	1253.5	1253.5	24.54	18.51

Costmap-MOA* and NAMOA* are nearly identical, the safety-oriented solutions differ markedly: Costmap-MOA* selects routes that keep a consistently larger clearance from obstacle boundaries—most noticeably in the cluttered environment—whereas NAMOA* tends to follow tighter passages and remain closer to obstacles



(a) Map 1 **(b)** Map 2

Fig. 11 Best distance vs. best safety trajectories.

4.6 Computational Efficiency

Tables 3 and 4 report runtime. On Map 1, Costmap-MOA* requires approximately 20 seconds per query while returning a substantially larger solution set than NAMOA* (up to 58 solutions versus 32). On Map 2, Costmap-MOA* remains moderately slower than NAMOA* (e.g., 5.84 s vs. 3.28 s in S1), but the additional computation translates directly into a richer Pareto front and lower best-safety costs. Overall, Costmap-MOA*

provides a favorable trade-off between computation and solution quality for safety-aware navigation where multiple candidate trajectories are beneficial.

5 Conclusion and Future Work

This paper introduced a multi-objective A* (Costmap-MOA*) planner that operates on a continuous cost-map combining terrain relief, obstacle proximity and local connectivity. The cost-map formulation captures both global environment structure and local safety, allowing the planner to explicitly trade off path length against accumulated navigation risk.

At the algorithmic level, Costmap-MOA* extends classical A* to a bi-objective setting by maintaining local Pareto fronts at each node and by using a lexicographic queue ordering. This design preserves the distance-optimal solution of standard A* while systematically exploring alternative, more conservative trajectories that maximize obstacle clearance. The resulting Pareto set provides the decision-maker with a continuum of feasible options between shortest and safest paths.

Extensive simulations on sparse and cluttered environments demonstrate that Costmap-MOA* consistently outperforms the NAMOA* benchmark. The proposed method achieves higher hypervolume values, improved spread and spacing indicators, and generates a substantially larger number of non-dominated solutions. Importantly, the distance-optimal paths found by both algorithms are identical, confirming that the performance gains are achieved without sacrificing optimality in the primary objective.

Future work will focus on three main directions:

- Extend Costmap-MOA* to dynamic and partially observable environments with adaptive replanning and real-time map updates.
- Integrate the planner into ROS and realistic simulators (e.g., Gazebo) to validate performance on standard robot platforms.
- Adapt Costmap-MOA* to covering path planning, combining our continuous costmap with energy and traversability metrics, and comparing it to multi-objective evolutionary approaches for coverage tasks [23].

Declarations

Funding: No funding was provided for this research.
Conflicts of interest/Competing interests : The authors have no conflicts of interest to declare that are relevant to the content of this article
Ethics approval: No ethical approval was deemed necessary.
Availability of data and material: Data and materials developed in this research are not currently online.
Authors' contributions: All authors H.Ouelmokhtar, S.Oussidhoum, H.Belgaid, Z.Ait-Chekdhidh and Y.Benmoussa, have contributed in softwar development, writing, corrections and improvements of the paper.
Consent to participate: The authors, H.Ouelmokhtar, S.Oussidhoum, H.Belgaid, Z.Ait-Chekdhidh and Y.Benmoussa voluntarily agree to participate in this research study.

Consent for publication: The authors, H.Ouelmokhtar, S.Oussidhoum, H.Belgaid, Z.Ait-Chekdhidh and Y.Benmoussa give their consent for information about themselves to be published in the Journal of Intelligent Robotics and Applications.

References

[1] Soltani, A.R., Tawfik, H., Goulermas, J.Y., Fernando, T.: Path planning in construction sites: performance evaluation of the dijkstra, a and ga search algorithms. Advanced engineering informatics **16**(4), 291–303 (2002)

[2] Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. The international journal of robotics research **5**(1), 90–98 (1986)

[3] Mandow, L., Cruz, J.P., *et al.*: A new approach to multiobjective a* search. In: IJCAI, vol. 8 (2005)

[4] Rostami, S.M.H., Sangaiah, A.K., Wang, J., Liu, X.: Obstacle avoidance of mobile robots using modified artificial potential field algorithm. EURASIP Journal on Wireless Communications and Networking **2019**(1), 1–19 (2019)

[5] Weerakoon, T., Ishii, K., Nassiraei, A.A.F.: An artificial potential field based mobile robot navigation method to prevent from deadlock. Journal of Artificial Intelligence and Soft Computing Research **5**(3), 189–203 (2015)

[6] Di, L., Huang, N., He, J., Wu, X., Huang, H., Su, Y., Liu, T.: Trajectory tracking and obstacle avoidance in dynamic environments using an improved artificial potential field method. PLoS One **20**(7), 0326879 (2025)

[7] Guo, T., Sun, Y., Liu, Y., Liu, L., Lu, J.: An automated guided vehicle path planning algorithm based on improved a* and dynamic window approach fusion. Applied Sciences **13**(18), 10326 (2023)

[8] Su, C., Zhao, L., Xiang, D.: Adaptive a*–q-learning–dwa fusion with dynamic heuristic adjustment for safe path planning in spraying robots. Applied Sciences **15**(17), 9340 (2025)

[9] Hu, Z., Wang, X., Zhang, J.: Research on dynamic obstacle avoidance method of mobile robot based on improved a* algorithm and dynamic window method. Engineering Letters **32**(3) (2024)

[10] Li, Y., Cheng, H.: App: A* post-processing algorithm for robots with bidirectional shortcut and path perturbation. IEEE Robotics and Automation Letters **8**(11), 7775–7782 (2023)

[11] Wang, X., Li, G., Bian, Z.: Research on the a* algorithm based on adaptive weights and heuristic reward values. World Electric Vehicle Journal **16**(3), 144 (2025)

[12] Johnson, J.J., Li, L., Liu, F., Qureshi, A.H., Yip, M.C.: Dynamically constrained motion planning networks for non-holonomic robots. In: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 6937–6943 (2020). IEEE

[13] Li, J., Ran, M., Xie, L.: Efficient trajectory planning for multiple non-holonomic mobile robots via prioritized trajectory optimization. IEEE Robotics and Automation Letters **6**(2), 405–412 (2020)

[14] Mandow, L., De La Cruz, J.L.P.: Multiobjective a* search with consistent heuristics. Journal of the ACM (JACM) **57**(5), 1–25 (2008)

[15] Pulido, F.-J., Mandow, L., Pérez-de-la-Cruz, J.-L.: Dimensionality reduction in multiobjective shortest path search. Computers & Operations Research **64**, 60–70 (2015)

[16] Hernández, C., Yeoh, W., Baier, J.A., Zhang, H., Suazo, L., Koenig, S., Salzman, O.: Simple and efficient bi-objective search algorithms via fast dominance checks. Artificial intelligence **314**, 103807 (2023)

[17] Ulloa, C.H., Yeoh, W., Baier, J.A., Zhang, H., Suazo, L., Koenig, S.: A simple and fast bi-objective search algorithm. In: Proceedings of the International Conference on Automated Planning and Scheduling, vol. 30, pp. 143–151 (2020)

[18] Ahmadi, S., Sturtevant, N.R., Raith, A., Harabor, D., Jalili, M.: Parallelizing multi-objective a* search. In: Proceedings of the International Conference on Automated Planning and Scheduling, vol. 35, pp. 131–139 (2025)

[19] Wilde, N., Smith, S.L., Alonso-Mora, J.: Scalarizing multi-objective robot planning problems using weighted maximization. IEEE Robotics and Automation Letters **9**(3), 2503–2510 (2024)

[20] Ganganath, N., Cheng, C.-T., Chi, K.T.: Multiobjective path planning on uneven terrains based on namoa. In: 2016 IEEE International Symposium on Circuits and Systems (ISCAS), pp. 1846–1849 (2016). IEEE

[21] Davoodi, M., Panahi, F., Mohades, A., Hashemi, S.N.: Clear and smooth path planning. Applied Soft Computing **32**, 568–579 (2015)

[22] Lin, X., Peng, X., Liang, L.: Simultaneous multi-objective path planning with cumulative hazardous dosage constraint for mobile detection robots in complex environments. Expert Systems with Applications, 129808 (2025)

[23] Ouelmokhtar, H., Benmoussa, Y., Benazzouz, D., Ait-Chikh, M.A., Lemarchand, L.: Energy-based usv maritime monitoring using multi-objective evolutionary algorithms. Ocean Engineering **253**, 111182 (2022)