Occupancy-Aware Reasoning for Safe Quadrotor Navigation: Perception-Aware MPPI

Yifan Zhai, Rudolf Reiter, and Davide Scaramuzza

***Abstract*—Safe robot autonomy in unstructured environments demand geometric sensing, and, importantly, robots must *reason* about what regions of the environment are traversable, occupied, or simply unknown. We present PA-MPPI (Perception-Aware Model Predictive Path Integral), a real-time quadrotor controller that tightly couples occupancy-aware map reasoning with sampling-based optimal control. PA-MPPI maintains a three-state occupancy grid (*free / occupied / unknown*) and introduces a novel perception cost that steers optimized trajectories toward unknown frontiers aligned with the goal, enabling exploration-driven navigation without external planners or reference trajectories. Running at $50\,\mathrm{Hz}$, PA-MPPI performs on par with the state-of-the-art safety-assured planner SUPER [1] across challenging cluttered scenes, while achieving up to 34% lower energy consumption. We further demonstrate that PA-MPPI serves as a safe and robust action policy for navigation foundation models, compensating for their lack of 3-D geometric awareness.**

***Index Terms*—Quadrotor navigation, occupancy mapping, model predictive path integral, perception-aware control, safe autonomy, foundation models.**

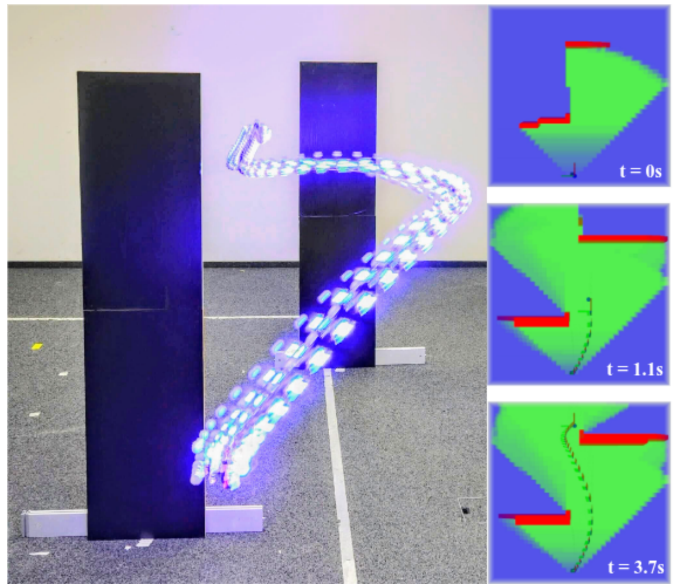


Fig. 1: Perception Aware Navigation.

I. Introduction

Reliable robot autonomy in real-world environments requires, beyond low-level sensing and localisation, to understand *what* different spatial regions imply for the robot, whether an area is safely traversable, blocked by an obstacle, or simply unobserved. This semantic distinction between *free*, *occupied*, and *unknown* space underpins safe motion planning, and exploiting it at the control level remains an open challenge relevant for environment understanding and safe interaction.

For quadrotor navigation in unknown environments, three key difficulties must be addressed simultaneously: *(i)* obstacle avoidance in non-convex free space; *(ii)* satisfaction of quadrotor-specific dynamics and actuation constraints; and *(iii)* goal-directed exploration of unmapped regions to find feasible paths. Classical hierarchical architectures [2], [3] decouple planning from control, introducing conservatism and making it hard to account for orientation-dependent perception objectives such as aligning the camera with informative frontiers.

Model Predictive Path Integral (MPPI) control [4] has recently emerged as a powerful alternative: its sampling-based optimization handles non-smooth, non-convex objectives and directly considers full quadrotor dynamics [5]. However, existing MPPI formulations for UAVs are confined to trajectory tracking—optimizing in a small neighbourhood of a reference—and lack reasoning about unknown space. When large obstacles block the direct path, they stall.

We build on PA-MPPI [6], which extends MPPI with a semantic perception cost derived from an online-updated 3-D occupancy map. The key insight is treating the three map states as *semantic labels* that carry actionable meaning: *unknown* voxels along the goal direction represent exploration opportunities; *occupied* voxels prohibit traversal. PA-MPPI's perception cost encodes this reasoning directly into the trajectory optimizer, enabling reference-free, map-grounded navigation at $50\,\mathrm{Hz}$. Fig. 1 shows PAMPPI navigating around obstacles and exploring unknown space.

We present three key contributions:

- A novel **semantic perception cost** for MPPI that distinguishes free/occupied/unknown space to guide goal-directed exploration.
- An **integrated framework** coupling depth sensing, ROG-Map occupancy mapping [7], and GPU-accelerated MPPI at $50\,\mathrm{Hz}$.
- **Hardware-in-the-loop validation** showing performance on par with the state-of-the-art planner SUPER [1] and safe deployment as the action policy for the navigation foundation model NoMaD [8].

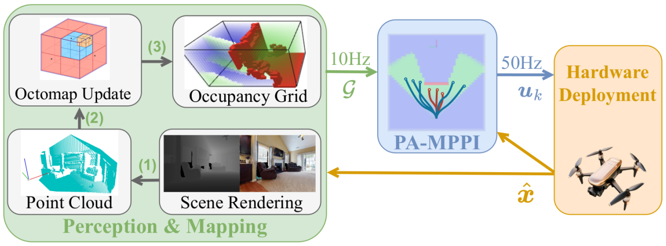


Fig. 2: Modules involved in the perception-aware navigation algorithm PA-MPPI.

II. Related Work

A. Obstacle Avoidance and Occupancy Mapping

Classical planners [2], [3] rely on discrete search spaces, separating perception from control. Optimization-based approaches [9], [10] reason jointly over trajectory and obstacle geometry but require smooth, differentiable representations that are difficult to obtain from raw depth images in cluttered scenes. Learned depth-based policies [11] improve closed-loop cost but lack explicit constraint handling. PA-MPPI addresses all three gaps by operating directly on the voxel-state occupancy grid without requiring a differentiable environment model.

B. Perception-Aware Planning

Prior work integrates perception objectives to maintain target visibility [12], [13] or manage localisation uncertainty [14]. SUPER [1] simultaneously plans exploratory and backup trajectories guided by A* search in free and unknown space. These approaches treat exploration as a separate planning layer. PA-MPPI unifies semantic map reasoning and low-level control in a single optimization loop, removing the planner–controller interface.

C. Sampling-Based MPC

MPPI [15], [4] uses importance-weighted Monte Carlo rollouts and has been applied to agile UAVs [5] and whole-body locomotion [16]. However, these works depend on external reference trajectories. PA-MPPI is the first MPPI formulation to achieve reference-free navigation via semantic environment reasoning.

III. Method

The method involves several modules outlined in the following and in Fig. 2.

A. Semantic Occupancy Map

A depth sensor streams point clouds into a ROG-Map [7] at $50\,\mathrm{Hz}$, maintaining a 3-D occupancy grid $\mathcal{G} \in \{-1, 0, 1\}^{X\times Y\times Z}$, where $1 =$occupied, $0 =$free, $-1 =$unknown. These three states are the *semantic labels* that are use to define the cost. The map covers a $5{\times}5{\times}2\,\mathrm{m}$ window at $0.1\,\mathrm{m}$ voxel resolution and is converted to a JAX array for GPU-accelerated access during MPPI rollouts.

B. MPPI Formulation

At each control step, $N = 17\,500$ trajectories of $H = 15$ steps are sampled by perturbing the nominal control sequence with multivariate Gaussian noise. Each trajectory is rolled out using a full quadrotor rigid-body model (position, quaternion, linear and angular velocity) with motor thrust constraints applied via clipping. The optimized control is the importance-weighted average

$$\mathbf{u}^* = \sum_{j=1}^{N} w^j\, \mathbf{u}^j, \quad w^j \propto \exp\Big(-\tfrac{\mathcal{L}^j - \mathcal{L}^{\min}}{\lambda}\Big), \tag{1}$$

where $\lambda = 0.02$ is the temperature parameter, $\mathcal{L}^{\min}$ is the lowest summed cost out of the N rollouts and the per-step cost $\mathcal{L}^j = \sum_{i=k}^{k+H-1} \ell(x_i^j, u_i^j) + \bar{V}(x_H^j)$.. The prediction step is $\Delta t_{\text{pred}} = 0.1\,\mathrm{s}$ while control executes at $\Delta t_{\text{ctrl}} = 0.02\,\mathrm{s}$, decoupling prediction horizon from control frequency [5].

C. Semantic Perception Cost

The stage cost combines goal progress, actuation regularisation, collision avoidance, and the novel **semantic perception term** $\ell_{\text{perception}}$ to the final cost $\ell = \ell_{\text{goal}} + \ell_{\text{goal},H} + \ell_{\text{act}} + \ell_{\text{collision}} + \ell_{\text{perception}}$. Two phases govern its activation:

Phase 1 – Goal occluded. When no free line of sight exists from the quadrotor to the goal, $\ell_{\text{perception}}$ activates two mechanisms: *(a)* a point-of-interest alignment term penalising deviation of the camera x-axis from the goal direction; and *(b)* a ray-tracing term that casts a ray toward the goal and evaluates the first voxel where the ray exits free space. If this voxel is *unknown*, a negative reward is given ($c_{\text{unknown}} = -4.0$), encoding that an exploration opportunity exists in the goal direction. If it is *occupied*, a positive penalty ($c_{\text{occupied}} = 2.0$) discourages trajectories facing walls.

Phase 2 – Direct line of sight. Once the ray reaches the goal through free space, exploration is unnecessary. $\ell_{\text{perception}}$ is deactivated and a high-weight goal cost ($c_{\text{goal}} = 5.0$) steers the quadrotor directly to the goal. Ray-tracing uses the 3-D Digital Differential Analyser (DDA) algorithm [17] and is evaluated at the last horizon step $H-1$ to manage computational cost.

The collision cost $\ell_{\text{collision}}$ is a large binary penalty ($c_{\text{collision}} = 15.0$) that fires whenever the quadrotor occupies a non-free voxel, enforcing that the vehicle never enters unknown space without mapping it first—a key safety invariant for navigation in unknown environments.

IV. Experiments

A. Experimental Setup

Experiments use Flightmare [18] simulation and hardware-in-the-loop (HIL) with a motion-capture system for ground-truth state and real-time depth rendering [19]. PA-MPPI is implemented in JAX and runs on an Intel i7-13800H CPU with an NVIDIA A1000 GPU ($6\,\mathrm{GB}$ VRAM). The quadrotor weighs $210\,\mathrm{g}$ with arm length $19.4\,\mathrm{cm}$ and a thrust-to-weight ratio of 6.8. SUPER [1] is used as the primary baseline, tuned for maximum speed without collisions in each test scene. In contrast, PA-MPPI uses identical hyperparameters across all scenes without per-scene tuning.

TABLE I: Navigation results for the hardest configuration of each scene. All methods achieve 100% success; PA-MPPI achieves higher average velocity and lower energy consumption.

Scene	Method	Time (s)	Avg. Vel. (m/s)	Energy (J)
C-wall (w=3.0 m)	SUPER	6.00 ± 0.84	1.08 ± 0.03	286.8 ± 42
	PA-MPPI (sim)	5.14 ± 0.22	1.14 ± 0.06	$\mathbf{249.5} \pm 9$
	PA-MPPI (HIL)	$\mathbf{5.13} \pm 0.18$	$\mathbf{1.15} \pm 0.06$	—
Hole (d=1.0 m)	SUPER	4.26 ± 0.56	0.89 ± 0.08	203.4 ± 27
	PA-MPPI (sim)	2.79 ± 0.35	1.24 ± 0.08	$\mathbf{135.8} \pm 17$
	PA-MPPI (HIL)	$\mathbf{2.67} \pm 0.30$	$\mathbf{1.32} \pm 0.10$	—
4-Wall (w=1.5 m)	SUPER	7.57 ± 0.99	0.72 ± 0.07	364.3 ± 50
	PA-MPPI (sim)	$\mathbf{5.78} \pm 0.23$	$\mathbf{1.05} \pm 0.07$	$\mathbf{281.7} \pm 11$
	PA-MPPI (HIL)	6.75 ± 0.75	0.88 ± 0.10	—

B. Synthetic Scene Results

Three obstacle types of increasing difficulty are tested: a C-shaped wall, a wall with a traversal hole, and four walls with narrow gaps. Each scene is evaluated at multiple difficulty levels, five trials per configuration. Both PA-MPPI and SUPER achieve 100% success across all scenes; PA-MPPI consistently produces faster and more energy-efficient trajectories, as summarised in Table I.

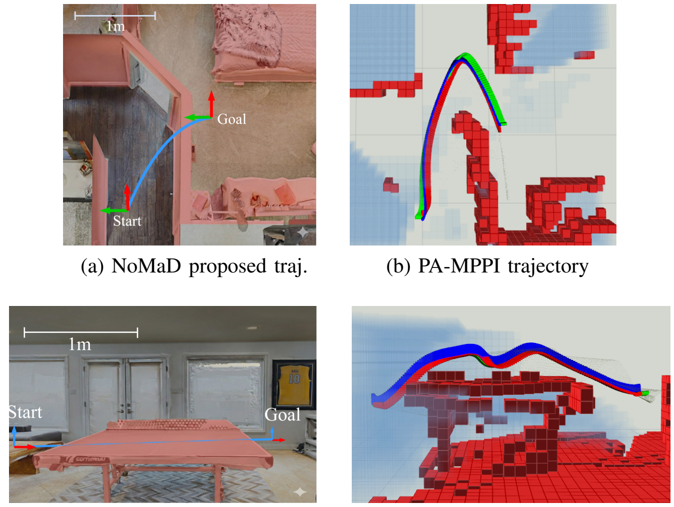
The improvement arises from PA-MPPI's occupancy-aware reasoning loop: it perceives unknown voxels in the goal direction, biases sampled trajectories toward those frontiers, maps them on approach, and transitions to direct goal-seeking once the path clears—all without a separate planner. By contrast, SUPER's A*-based front-end plans without quadrotor dynamics awareness, yielding longer, less energy-efficient trajectories.

C. Robustness Under Wind Disturbance

Wind disturbances of 1–3 $\mathrm{m\,s^{-1}}$ are injected in the xy-plane across ten episodes per magnitude in the hardest configuration of each scene. At $1\,\mathrm{m\,s^{-1}}$, PA-MPPI retains high success rates. At $3\,\mathrm{m\,s^{-1}}$, disturbances push the quadrotor into unknown regions, triggering early termination via the collision cost. While this prevents uncontrolled entry into unmapped space, it constitutes a current limitation: improving wind robustness without sacrificing the safety invariant, as well as a direct comparison with SUPER under identical disturbances, are left for future work.

D. Foundation Model Integration

PA-MPPI serves as a safe action policy for NoMaD [8], a navigation foundation model that outputs 2-D waypoints from monocular RGB images. NoMaD's proposals are scale-ambiguous and frequently collide with scene geometry. PA-MPPI receives NoMaD's waypoints as intermediate goals and plans collision-free trajectories using its occupancy map, effectively acting as a safety filter. In two Habitat-Matterport scenes [20]—one requiring navigation through a doorway, one avoiding a ping-pong table—PA-MPPI successfully reaches the proposed goals by reasoning around obstacles in the occupancy map that NoMaD cannot perceive. This demonstrates how perception-aware control can robustly ground high-level foundation model outputs, bridging the gap between image-conditioned reasoning and safe physical execution.



(a) NoMaD proposed traj. (b) PA-MPPI trajectory

(c) NoMaD proposed traj. (d) PA-MPPI trajectory

Fig. 3: Two Habitat scenes, with obstacles (walls, furniture, etc) overlaid in red and NoMaD proposed trajectory in blue (a)(c), and visualization of the PA-MPPI trajectory in the occupancy map (b)(d).

V. Discussion

A. What Environment Representation Properties Matter?

PA-MPPI demonstrates that a ternary occupancy label (free / occupied / unknown), standard in robotic mapping [21], is sufficient to enable rich navigation behaviour when tightly coupled to the control optimizer. Our finding is that differentiability is *not* required when a sampling-based framework is used, and the occupancy-state distinction between unmapped and blocked space is *critical*. Future work could enrich this layer with object-level affordances or traversability confidence score.

B. Grounding Foundation Models with Perception-Aware Control

The NoMaD integration illustrates how perception-aware control can bridge the gap between high-level, image-conditioned foundation models and safe physical execution. A natural extension is a language-conditioned semantic cost, where a large language model specifies which semantic regions to treat as hazardous (e.g. *"avoid wet floors"* translated into an elevated voxel cost).

C. Limitations and Future Directions

PA-MPPI's planning horizon is limited by compute, constraining navigable scene complexity. A richer terminal value function or hierarchical semantic reasoning could extend this. The current map does not use confidence levels. Therefore, incorporating probabilistic occupancy would allow uncertainty-weighted perception costs. Extending to multi-robot semantic sharing, richer semantic hierarchies (object identity, affordance), and longer-horizon navigation are directions for future work.

VI. Conclusion

We have presented how PA-MPPI [6], a perception-aware MPPI controller, embeds occupancy-aware environment reasoning, distinguishing free, occupied, and unknown space directly in the sampling-based cost function. This enables reference-free, safe quadrotor navigation in unknown environments at $50\,\mathrm{Hz}$, matching state-of-the-art planning performance while reducing energy consumption and providing a safe action policy for navigation foundation models. A main conclusion is that occupancy-state labels, however simple, can substantially improve reliability and safety when integrated at the control level rather than delegated to a separate planning layer.

References

[1] Y. Ren, F. Zhu, G. Lu, Y. Cai, L. Yin, F. Kong, J. Lin, N. Chen, and F. Zhang, "Safety-assured high-speed navigation for mavs," *Science Robotics*, vol. 10, no. 98, p. eado6187, 2025.

[2] T. Cieslewski, E. Kaufmann, and D. Scaramuzza, "Rapid exploration with multi-rotors: A frontier selection method for high speed flight," in *IROS*, 2017.

[3] M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart, "Path planning for motion dependent state estimation on micro aerial vehicles," in *IEEE International Conference on Robotics and Automation*, 2013.

[4] G. Williams, A. Aldrich, and E. A. Theodorou, "Model predictive path integral control: From theory to parallel computation," *Journal of Guidance, Control, and Dynamics*, 2017.

[5] M. Minařík, R. Pěnička, V. Vonásek, and M. Saska, "Model predictive path integral control for agile unmanned aerial vehicles," in *IROS*, 2024.

[6] Y. Zhai, R. Reiter, and D. Scaramuzza, "PA-MPPI: Perception-aware model predictive path integral control for quadrotor navigation in unknown environments," *IEEE Robotics and Automation Letters*, vol. 11, no. 3, pp. 3804–3811, 2026.

[7] Y. Ren, Y. Cai, F. Zhu, S. Liang, and F. Zhang, "ROG-Map: An efficient robocentric occupancy grid map for large-scene and high-resolution lidar-based motion planning," in *IROS*, 2024.

[8] A. Sridhar, D. Shah, C. Glossop, and S. Levine, "NoMaD: Goal Masked Diffusion Policies for Navigation and Exploration," *arXiv preprint*, 2023.

[9] Y. Gao, F. Messerer, N. v. Duijkeren, B. Houska, and M. Diehl, "Real-time-feasible collision-free motion planning for ellipsoidal objects," in *2024 IEEE 63rd Conference on Decision and Control (CDC)*, 2024.

[10] R. Reiter, K. Baumgärtner, R. Quirynen, and M. Diehl, "Progressive smoothing for motion planning in real-time NMPC," in *European Control Conference (ECC)*, 2024, pp. 1816–1823.

[11] A. Loquercio, E. Kaufmann, R. Ranftl, M. Müller, V. Koltun, and D. Scaramuzza, "Learning high-speed flight in the wild," *Science Robotics*, vol. 6, no. 59, p. eabg5810, 2021.

[12] D. Falanga, P. Foehn, P. Lu, and D. Scaramuzza, "PAMPC: Perception-aware model predictive control for quadrotors," in *IROS*, 2018.

[13] J. Xing, G. Cioffi, J. Hidalgo-Carrió, and D. Scaramuzza, "Autonomous power line inspection with drones via perception-aware MPC," in *IROS*, 2023.

[14] C. Papachristos, F. Mascarich, S. Khattak, T. Dang, and K. Alexis, "Localization uncertainty-aware autonomous exploration and mapping with aerial robots using receding horizon path-planning," *Autonomous Robots*, vol. 43, no. 8, pp. 2131–2161, 2019.

[15] E. A. Theodorou and E. Todorov, "Relative entropy and free energy dualities: Connections to Path Integral and KL control," in *IEEE 51st IEEE Conference on Decision and Control (CDC)*, Dec. 2012.

[16] J. Alvarez-Padilla, J. Z. Zhang, S. Kwok, J. M. Dolan, and Z. Manchester, "Real-time whole-body control of legged robots with model-predictive path integral control," in *ICRA*, 2025.

[17] J. Amanatides and A. Woo, "A fast voxel traversal algorithm for ray tracing," *Proceedings of EuroGraphics*, vol. 87, 08 1987.

[18] Y. Song, S. Naji, E. Kaufmann, A. Loquercio, and D. Scaramuzza, "Flightmare: A flexible quadrotor simulator," in *CoRL*, 2020.

[19] P. Foehn, E. Kaufmann, A. Romero, R. Pěnička, S. Sun, L. Bauersfeld, T. Laengle, G. Cioffi, Y. Song, A. Loquercio, and D. Scaramuzza, "Agilicious: Open-source and open-hardware agile quadrotor for vision-based flight," *Science Robotics*, vol. 7, 06 2022.

[20] S. K. Ramakrishnan, A. Gokaslan, E. Wijmans, O. Maksymets, A. Clegg, J. M. Turner, E. Undersander, W. Galuba, A. Westbury, A. X. Chang, M. Savva, Y. Zhao, and D. Batra, "Habitat-matterport 3d dataset (HM3d): 1000 large-scale 3d environments for embodied AI," in *55th Conference on Neural Information Processing Systems*, 2021.

[21] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, 2013.