Autonomous Ground Navigation in Highly Constrained Spaces: Lessons Learned from the Fifth BARN Challenge at ICRA 2026

Competition Organizers: Xuesu Xiao[1], Zifan Xu[2], Saad Abdul Ghani[1], Daeun Song[3], Alexandyr Card[1], Nhat Le[1], and Peter Stone[2,4],
Winning Teams: Yunting Ji[5], Zihao Mao[5], Wenchao Zhang[5], Xiangyi Meng[5], Shenglin Li[5], Hahjin Lee[3], Jiyoung Park[3], Young J. Kim[3], Dun Yan Ong[6], Jin You Chan[6], Jason Wai Hao Yee[6], Wenda Chang[6], Khiem Nguyen[6], Yinuo Qu[6], and Yoonchang Sung[6]

Abstract— **The Fifth Benchmark Autonomous Robot Navigation (BARN) Challenge took place at the 2026 IEEE International Conference on Robotics and Automation (ICRA 2026) in Vienna, Austria. It continued to evaluate the performance of state-of-the-art autonomous ground navigation systems in highly constrained environments. After Philadelphia (North America), London (Europe), Yokohama (Asia), and Atlanta (North America), the challenge returned to Europe for the second time. 17 teams participated in the simulation competition, seven of which were invited to the physical competition, while four of them finally competed in Vienna. This year's challenge focused exclusively on static, highly constrained environments, without the dynamic obstacle component used previously. Second year in a row, all teams in the physical finals adopted classical navigation approaches without using machine learning. In this article, we discuss the challenge, the approaches used by the three winning teams, and lessons learned to direct future research and competitions.**

I. The Fifth BARN Challenge Overview

The Fifth BARN Challenge took place as a conference competition at ICRA 2026 in Vienna, Austria. As a continuation of The First, Second, Third, and Fourth BARN Challenges at ICRA 2022, 2023, 2024, and 2025 in Philadelphia, London, Yokohama, and Atlanta, respectively, the fifth challenge aimed to evaluate the capability of state-of-the-art navigation systems to move robots through static, highly constrained obstacle courses, an *ostensibly* simple problem even for many experienced robotics researchers, but in fact, as the results from every year's competitions suggest, a problem far away from being solved [1]–[4].

Each team needed to develop an entire navigation software stack for a standardized and provided mobile robot, i.e., a Clearpath Jackal [5] with a 2D 270°-field-of-view Hokuyo LiDAR for perception and a differential drive system with 2m/s maximal speed for actuation. The developed navigation software stack needed to autonomously drive the robot from a given starting location through a dense obstacle field to a given goal without any collision with obstacles or any human interventions. The team whose system could best accomplish this task within the least amount of time would win the competition. The Fifth BARN Challenge had two phases: a qualifying phase evaluated in simulation, and a final phase evaluated in three physical obstacle courses. The qualifying phase took place before the ICRA 2026 conference using the BARN dataset [6] (with the recent addition of DynaBARN [7]), which is composed of 300 obstacle courses in Gazebo simulation randomly generated by cellular automata. The top seven teams from the simulation phase were then invited to compete in three different physical obstacle courses set up by the organizers at ICRA 2026 in the VIECON, Vienna Congress & Convention Center.

In this article, we report on the simulation qualifier and physical finals of The Fifth BARN Challenge at ICRA 2026, present the approaches used by the top three teams, discuss lessons learned from the challenge compared against the first, second, third, and fourth BARN Challenge at ICRA 2022, 2023, 2024, and 2025, and point out future research directions to solve the problem of autonomous ground navigation in highly constrained spaces.

II. Simulation Qualifier

The simulation qualifier of The Fifth BARN Challenge started on January 1st, 2026. The qualifier used the BARN dataset [6], which consists of 300 5m $\times$ 5m obstacle environments randomly generated by cellular automata (see examples in Fig. 1), each with a predefined start and goal. These obstacle environments range from relatively open spaces, where the robot barely needs to turn, to highly dense fields, where the robot needs to squeeze between obstacles with minimal clearance. The BARN environments are open to the public, and were intended to be used by the participating teams to develop their navigation stack. Another 50 unseen environments, which are not available to the public, were generated to evaluate the teams' systems. A random BARN environment generator was also provided to the teams so that they could generate their own unseen test environments.[1]

In addition to the 300 BARN environments, six baseline approaches were also provided for the participants' reference, ranging from classical sampling-based [8] and optimization-based navigation systems [9], to end-to-end machine learning methods [10], [11], and hybrid approaches [12]. All baselines were implementations of different local planners used in conjunction with Dijkstra's search as the global planner in the ROS `move_base` navigation stack [13]. Additionally, the

[1]https://github.com/dperille/jackal-map-creation

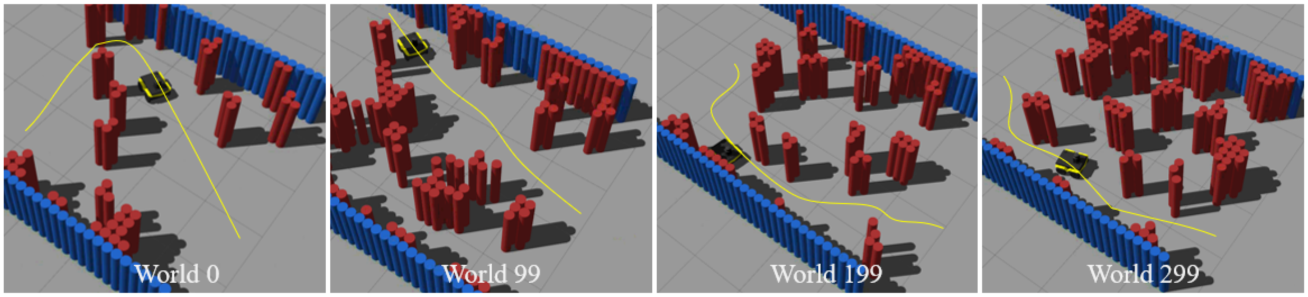

Fig. 1: Four example BARN environments in the Gazebo simulator (ordered by ascending relative difficulty level).
.

winning teams' navigation stacks from the last four competitions were also open sourced [14]. To facilitate participation, a training pipeline capable of running the standardized Jackal robot in the Gazebo simulator with ROS Noetic (in Ubuntu 20.04), with the option of being containerized in Docker or Singularity containers for fast and standardized setup and evaluation, was also provided.[2]

This year, the organizers also introduced a ROS2 version of the competition considering the growing interest in developing in ROS2. Out of the 17 simulation participants, 12 used ROS and 5 used ROS2.

A. Rules

Each participating team was required to submit their developed navigation system as a (collection of) launchable ROS node(s). The challenge utilized a standardized evaluation pipeline[3] to run each team's navigation system and compute a standardized performance metric that considers navigation success rate (collision or not reaching the goal count as failure), actual traversal time, and environment difficulty (measured by optimal traversal time). Specially, the score s for navigating each environment i was computed as

$$s_i = 1_i^{\text{success}} \times \frac{\text{OT}_i}{\text{clip}(\text{AT}_i, 2\text{OT}_i, 8\text{OT}_i)},$$

where the indicator function 1_{success} evaluates to 1 if the robot reaches the navigation goal without any collision, and evaluates to 0 otherwise. AT denotes the actual traversal time, while OT denotes the optimal traversal time, as an indicator of the environment difficulty and measured by the shortest traversal time assuming the robot always travels at its maximal speed (2m/s):

$$\text{OT}_i = \frac{\text{Path Length}_i}{\text{Maximal Speed}}.$$

The Path Length is provided by the BARN dataset based on Dijkstra's search from the given start to goal. The clip function clips AT within 2OT and 8OT in order to assure navigating extremely quickly or slowly in easy or difficult environments respectively won't disproportionally scale the score. Notice that the lower bound 2OT was reduced from the previous 4OT used in the first two challenges, considering the performance upper bound, 0.25, has been closely approached by multiple teams. Starting from the third BARN Challenge, the upper bound has been increased to 0.5 to encourage faster navigation speed. The overall score of each team is the score averaged over all 50 unseen test BARN environments, with 10 trials in each environment. Higher scores indicate better navigation performance. The six baselines score between 0.1656 and 0.4354 [14].

B. Results

The simulation qualifier started on January 1st, 2026 and lasted through a soft submission deadline (April 8th, 2026) and a hard submission deadline (May 8th, 2026). Submitting by the soft deadline will guarantee an invitation to the final physical competition given good navigation performance in simulation and leave sufficient time for invited participants to make travel arrangements to Vienna. The hard deadline is to encourage broader participation, but final physical competition eligibility will depend on the available capacity and travel arrangement made beforehand. In total, 17 teams, 13 from Asia, two from North America, one from South America, and one from Africa, submitted their navigation systems. The performance of each submission was evaluated by the standard evaluation pipeline. The results are shown in Table I with the baselines shown in the fourth column as a reference. We also evaluate the navigation systems in the DynaBARN dataset with the results shown in parenthesis in Table I, but the DynaBARN results are not considered in the final scoring and ranking. Considering the physical finals did not include the DynaBARN arena as last year, we gave the team the option not to be evaluated in the simulated DynaBARN enviornments (and hence the score 0.0000).

The top two simulation teams, IN^2BOT from Beijing Institute of Technology (BIT) and EW-Glab from Ewha Womans University, outperformed all last year's winning teams (RRSL, RobotiXX, and UVA AMR) in simulation. IN^2BOT's score, 0.4975, closely approached the performance upper bound, 0.5.

The top seven teams were invited to the physical finals at ICRA 2026 (Helios missed the hard submission deadline). Five could attend—IN^2BOT from BIT (China), EW-Glab from Ewha Womans University (South Korea), KKato, an independent participant (Japan), UFAM + INDT from Universidade Federal do Amazonas and Institute of Technology

[2] https://github.com/Daffan/ros_jackal
[3] https://github.com/Daffan/the-barn-challenge

TABLE I: Simulation Results.

Rank.	Team	Score	Baseline
1	IN^{2}BOT	0.4975 (0.0000)	
2	EW-Glab	0.4880 (0.0000)	
3	KKato	0.4856 (0.0000)	
4	UFAM + INDT	0.4814 (0.0000)	
5	ASU ROAR	0.4619 (0.0000)	
6	Team Robo	0.4515 (0.1236)	
7	Helios	0.4455 (0.0000)	LfLH [11]
8	CRAL	0.4289 (0.0000)	e2e [10]
9	PANAV	0.3682 (0.0000)	
10	Deptron	0.3392 (0.0000)	
11	eYantra	0.3119 (0.0000)	APPLR-DWA [12], E-Band [9], (Fast & Default) DWA [8]
12	LAMA	0.0517 (0.0000)	
13	LoneBot	NA	
14	VeRan	NA	
15	TUBE	NA	
16	Team K	NA	
17	PUST_BARNARS	NA	

(Brazil), and Team Robo from Nanyang Technological University (Singapore). However, KKato withdrew after failing to set up the navigation stack on the physical Jackals, leaving four to compete.

III. Physical Finals

The physical finals took place at ICRA 2026 in the VIECON, Vienna Congress & Convention Center from June 2nd to June 4th, 2026 (Fig. 2). Two physical Jackal robots with the same sensors and actuators were provided by the competition sponsor, Clearpath Robotics.

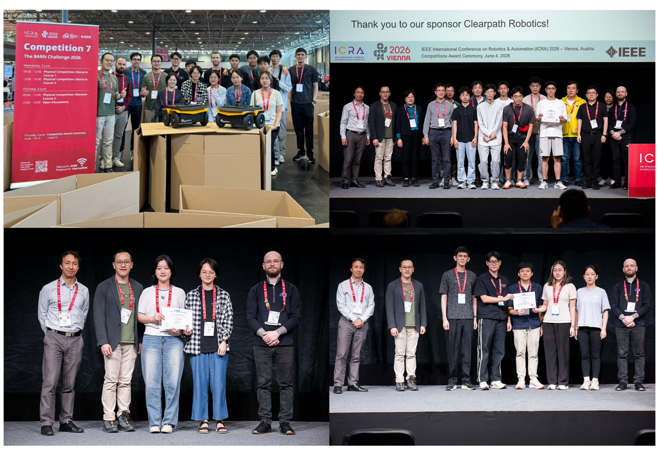

Fig. 2: Final physical competition participants and organizers at The Fifth BARN Challenge and award ceremony in Vienna, Austira.

A. Setup

Physical obstacle courses were set up using 200 cardboard boxes in the conference center. The organizers used the same guidelines to set up three obstacle courses as in the first four BARN challenges, i.e., all courses aimed at testing a navigation system's local planning and therefore had an obvious passage but with minimal clearance (a few centimeters around the robot) when traversing this passage. For logistical reasons, DynaBARN arena was not set up this year.

B. Rules

Each team had five minutes to set up their navigation systems after they received the robot without accessing each obstacle course. After the 5-minute setup, each team had a "cold trial" that required the team to navigate through the obstacle course without any fine-tuning. If the cold trial succeeded, it counted towards a bonus successful trial. This rewarded teams that could run their navigation stacks without any fine-tuning. After the cold trial, each team had the opportunity to run five timed trials (after notifying the organizers to be timed) within a 20-minute period (later expended to 25 minutes for the last two obstacle courses), similar to the previous years. In each obstacle course, the fastest three out of the five timed trials were counted, and the team that had the most successful trials would be the winner. In the case of a tie, the team with the fastest average traversal time would be declared the winner. Considering the robot traversal times were affected by the length of the obstacle courses, starting this year, a human teleoperated trial was conducted for each obstacle course, whose time was used to normalized the robot traversal time.

C. Results

The four teams' navigation performance is shown in Table II. The detailed results of all five timed trials (in seconds, only the top three were counted in the final score including the extra cold trial) are listed in the last three columns of Table II, where "X" indicates failure. The (+3) in the total runs indicate the three extra cold trials.

The winner, IN^{2}BOT, performed comparably to EW-Glab and Team Robo in the first obstacle course, but navigated the most slowly. EW-Glab showed a significant advantage in the second obstacle course by more successful trials and faster navigation. However, IN^{2}BOT caught up in the third obstacle course with three successful trials while EW-Glab had zero.

As a result, IN^{2}BOT won the competition by the most successful trials (7/9); EW-Glab came in second place with the second most successful trials (5/9); and Team Robo scored third with one-third successful trials (3/9).

IV. Top Three Teams and Approaches

In this section, we report the approaches used by the three winning teams.

A. IN^{2}BOT (BIT)

Team IN^{2}BOT developed a local-decision-centered ROS navigation stack for fast and robust traversal in narrow BARN-style environments. In narrow BARN-style environments, especially around curved passages, the key challenge is not only to find a feasible route but also to make local decisions that preserve passage clearance while enabling

TABLE II: Physical Results.

Rank.	Team	Success/Total	Norm. Avg. Time	Course 1	Course 2	Course 3
1	IN^{2}BOT	7/9(+3)	5.11	X/X/X/183/182/X	X/X/X/127/125/X	X/179/X/172/174
2	EW-Glab	5/9(+3)	2.91	X/132/137/X/X/X	112/111/109/X/111/116	X/X/X/X/X/X
3	Team Robo	3/9(+3)	2.60	X/X/X/X/138/140	X/X/84/X/X/X	X/X/X/X/X/X
4	UFAM + INDT	0/9(+3)	NA	X/X/X/X/X/X	X/X/X/X/X/X	X/X/X/X/X/X

safe acceleration and deceleration. Optimization-based local planners can be sensitive to inaccurate clearance modeling in such tight spaces, which may lead to overly conservative motion or small optimization errors that cause contact with obstacles.

To address this issue, IN^{2}BOT adopted a geometry-sampling-based reactive planning framework. An A* global path provides long-horizon decision guidance to avoid short-sighted local behavior, while a Vector Field Histogram(VFH*) planner implicitly evaluates passage clearance from the local LiDAR point cloud. Based on the selected VFH* direction, fuzzy velocity control generates smooth acceleration and deceleration, and a safety barrier control layer further constrains the final command for precise maneuvering in narrow spaces.

1) Corner-Aware Lookahead Target Selection: The global path is used as a geometric reference for local decision making. Let the path be represented by $\mathcal{P} = \{\mathbf{p}_0, \mathbf{p}_1, \dots, \mathbf{p}_n\}$, and let $\mathbf{x}_r$ be the current robot position. The planner first selects a local path point $\mathbf{p}_i$ ahead of the robot according to its distance from the current robot position,

$$i = \arg\min_j \left| \|\mathbf{p}_j - \mathbf{x}_r\| - L \right|,$$

where L is the desired lookahead distance. When a sharp turn exists ahead, the selected point is biased toward the corner region so that the robot can prepare for the turn before reaching it.

The selected path point is further converted into a control point. To estimate the local path geometry, the planner uses three neighboring path points, $\mathbf{p}_{i-1}$, $\mathbf{p}_i$, and $\mathbf{p}_{i+1}$, which define a local circumcircle with center $\mathbf{o}_i$. The outward direction of the local curve is computed as

$$\mathbf{u}_{\text{out}} = \frac{\mathbf{p}_i - \mathbf{o}_i}{\|\mathbf{p}_i - \mathbf{o}_i\|}.$$

The control point is then obtained by shifting the selected path point along this outward direction:

$$\mathbf{g} = \mathbf{p}_i + \rho_i \mathbf{u}_{\text{out}}.$$

Here, ρ_i is determined by the local curvature and the relative distance between the robot and the selected path point. A sharper curve or a closer corner produces a stronger outward shift, encouraging the robot to start turning earlier and avoid cutting into the inside of the corner. The generated control point and the VFH*-selected local driving direction are illustrated in Fig. 3.

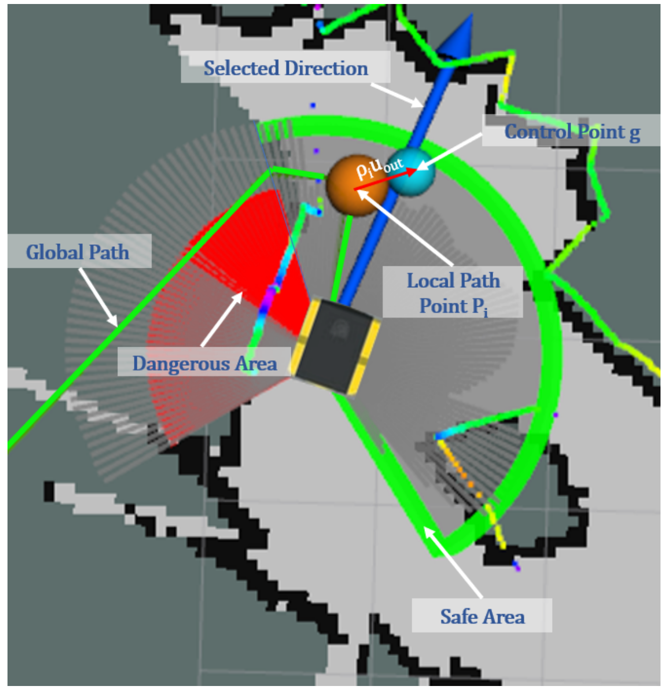

Fig. 3: **Control Point Generation and VFH* Direction Selection.** Control points are generated using the curvature of the global path at local path points. The VFH* planner selects the driving direction according to the control point orientation and the local LiDAR point cloud.

2) VFH* Local Planning and Fuzzy Velocity Control: The control target $\mathbf{g}$ is converted into a target sector in the robot frame and used by the VFH* local planner. The LiDAR scan is transformed into a polar obstacle histogram, where each angular sector represents the obstacle density and clearance in the corresponding direction. After smoothing and thresholding, consecutive free sectors are extracted as passable valleys. In this way, the VFH* representation implicitly captures the available passage width around the robot without explicitly constructing an optimization constraint for the corridor boundary.

For each passable valley, the planner generates candidate heading sectors. Centers of the narrow valleys are used to improve stability, while for wide valleys a heading closer to the target sector is selected after reserving a safety margin near the valley boundaries. The final heading is selected by balancing three factors: alignment with the control target, consistency with the previously selected heading, and local obstacle clearance. This allows the robot to follow the global decision guidance while still reacting to newly observed obstacles.

The selected VFH* heading is then passed to a fuzzy velocity controller. The angular velocity is determined by the heading error, while the linear velocity is adjusted according to the frontal obstacle distance and the required turning angle. As a result, the robot slows down before sharp turns or narrow gaps and accelerates in open regions. On top of this nominal command, the safety barrier control layer monitors a speed-dependent frontal safety region. If an obstacle enters this region, the controller suppresses forward motion and rotates toward the side with larger clearance, ensuring safe and precise maneuvering in narrow passages. Fig. 4 shows the VFH*-selected obstacle avoidance direction during a physical run.

Fig. 4: **Physical experiment of VFH*-based local obstacle avoidance.** The green line represents the global path, and the blue arrow indicates the local obstacle avoidance direction selected by VFH*.

B. EW-Glab (Ewha Womans University)

Team EW-GLAB from Ewha Womans University introduced an optimization-based local planning method using guidance points for safe navigation in cluttered environments. The method first generates a global plan using the A* algorithm to provide a reference route. Guidance points are then generated by adjusting waypoints derived from the global plan using a separation direction and line search. These guidance points are used as a constraint term in the local planner to guide the path toward collision-free regions with sufficient clearance from obstacles. This enables the robot to navigate safely through highly cluttered spaces.

1) Formulation: The objective of the local planning method is to attract the local trajectory toward collision-free regions using guidance points. To compute these points, a signed distance field $\phi(c)$ is constructed first on the grid map $\mathcal{M}$ for all grid cells $c \in \mathcal{M}$ based on the Chamfer distance. For a robot pose $\mathbf{p} = (\mathbf{x}, \theta) \in \mathbb{SE}(2)$, let $c_{\mathbf{x}} \in \mathcal{M}$ be the grid cell containing $\mathbf{x}$. For local planning, we adopt the Timed Elastic Band (TEB) framework [15], [16]. TEB computes a time-optimal trajectory $\mathcal{T}^*$ considering various constraints, including clearance from obstacles, velocity, and acceleration. These constraints are represented as weighted cost terms in the optimization objective.

2) Pose Correction: To provide safe guidance for local planning, uniformly sampled waypoints from the global plan are first adjusted toward regions with higher clearance. Following Lee and Kim [17], guidance points are generated by correcting global waypoints using separation direction and line search. For given waypoints $\mathbf{g}$, the separation direction $\mathbf{v}$ is determined from $\phi(c_{\mathbf{g}})$. Intuitively, this direction corresponds to the shortest displacement required to move $\mathbf{g}$ into free space (i.e., penetration depth). If $\mathbf{g}$ lies inside or near an obstacle, $\mathbf{v}$ is determined from the relative position between $\mathbf{g}$ and the nearest obstacle boundary point, so that it points away from the obstacle. If $\mathbf{g}$ lies on the boundary of the obstacle, $\mathbf{v}$ follows the local gradient of ϕ.

After determining the separation direction $\mathbf{v}$, each guidance point is obtained by shifting the corresponding waypoint along $\mathbf{v}$ to obtain sufficient collision clearance. From $\mathbf{g}$, a ray of length $d_{\max}$ is cast and rasterized into grid cells C. Then, directional hill climbing over C is performed by selecting cells with increasing distance values. The search terminates when C is exhausted, or no further increase in clearance is observed.

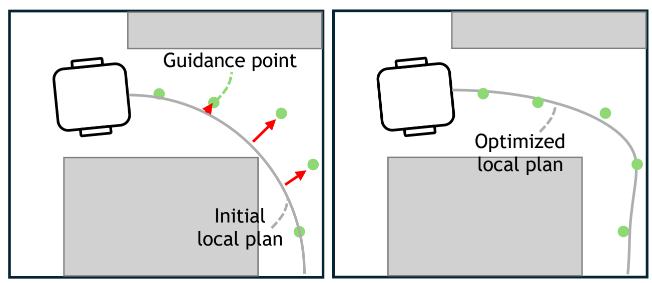

Fig. 5: **Guidance Point Based TEB Optimization.** Guidance points are generated to attract the initial local plan toward regions that provide sufficient clearance from obstacles.

3) Guidance Point Based TEB Optimization: Before optimization, the local segment of the global plan is time-parameterized to form the initial local plan. The guidance points obtained from the pose correction process are then incorporated into the TEB optimization to guide this initial local plan toward high-clearance regions. Each guidance point is matched to the closest TEB waypoint, and an attraction cost is applied to penalize the deviation of the corresponding TEB waypoint from the guidance point. This encourages the optimized local trajectory to remain close to the guidance points and move toward regions with higher obstacle clearance while satisfying the other TEB constraints, as shown in Fig. 5.

C. Team Robo (NTU)

Team Robo's key motivation is to build a navigation stack that remains useful beyond the specific static courses of BARN. The team presented a real-time Nonlinear Model Predictive Control (NMPC) framework for universal obstacle avoidance, balancing static and dynamic obstacles within a unified architecture. Unlike conventional local planners that prioritize static obstacle avoidance, a well-understood problem with mature solutions, Team Robo's approach leverages the computational efficiency of ACADOS [18] to incorporate both obstacle types directly into the constraint formulation, addressing the more challenging, often-neglected dynamic case without compromising static performance.

1) Formulation: The robot's state variables is defined as

$$\mathbf{x} = [x, y, \theta, v_r, v_l]^T \in \mathbb{R}^5,$$

where (x, y) denotes the planar position, θ the heading, and (v_r, v_l) velocities of the pairs of the right and left wheels, respectively. The control input

$$\mathbf{u} = [a_r, a_l]^T \in \mathbb{R}^2,$$

represents accelerations of the right and left wheels. The continuous-time dynamics are discretised using multiple shooting over a receding horizon of $T_f = 2.0$ seconds partitioned into $N = 20$ stages, yielding $\Delta t = 0.1$ s. The cost function minimized at each control cycle is

$$\begin{aligned} \min \quad & \sum_{k=0}^{N-1} \left((x_k - x_{\text{ref},k})^T Q (x_k - x_{\text{ref},k}) + u_k^T R u_k \right) \\ & \quad + (x_N - x_{\text{ref},N})^T Q (x_N - x_{\text{ref},N}) \\ \text{subject to} \quad & x_{k+1} = x_k + \Delta t\, f(x_k, u_k), \forall k = 0, \ldots, N-1 \end{aligned}$$

where Q and R are weighting matrices for tracking error and control effort.

2) Implementation: The global trajectory is generated by ROS `move_base` using the `NavfnROS` planner. Heading references derived from the global plan are then smoothed using a lookahead-based low-pass filter to avoid aggressive wiggles during traversals while retaining minor oscillations that aid laser scan excitation for dynamic obstacle tracking. Obstacle information is obtained from the local costmap of `move_base` and live laser scans from the Jackal, as shown in Fig. 6. A KD-tree is used to select the closest left and right static obstacles at each stage. In addition, an obstacle detection algorithm based on the same laser scans tracks up to ten dynamic obstacles with estimated positions and velocities, and propagates them over the prediction horizon. Based on the estimated velocities, the NMPC triggers a rear-side detour when $\|v_{\text{obs}}\| > \|v_{\text{robot}}\|$, as shown in Fig. 7. If the separation distance to the obstacle is insufficient for a forward lateral maneuver, the NMPC executes a reversal to prevent collision. The complete obstacle set is encoded into a 24-dimensional parameter vector and passed to the ACADOS solver at each control cycle.

V. Discussion

We discuss new findings and lessons from the Fifth BARN Challenge, not only from the technical perspective but also from the competition organization side.

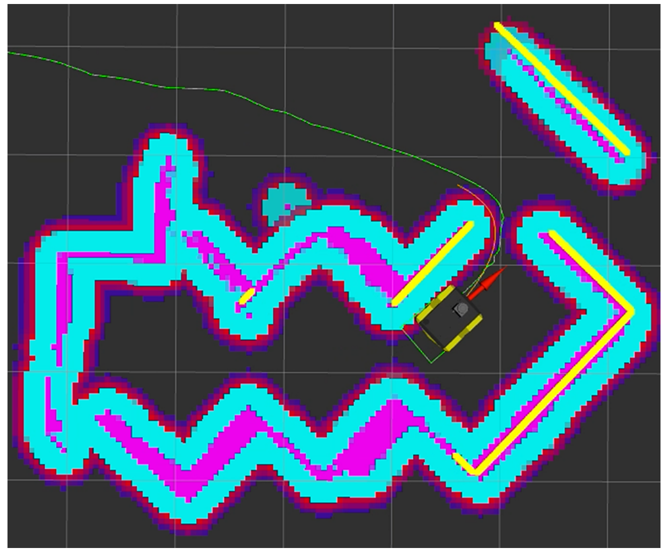

Fig. 6: **RViz Visualization of Static Obstacles.** Gradient of colors from blue to red represents the costmap. Yellow squares: laser scan

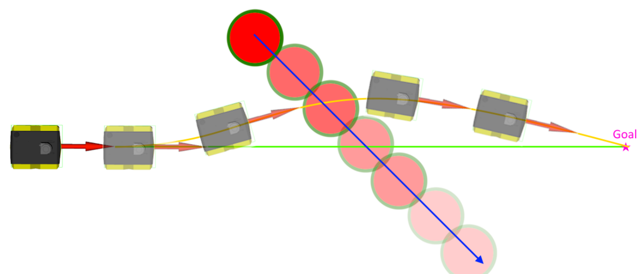

Fig. 7: **Visualization of Dynamic Obstacles Avoidance.** Green line is the global plan. Orange line is the local plan. Blue arrow is the velocity of the obstacle.

A. Simulation Result Approaches Performance Upper Bound Again

Since the Third BARN Challenge at ICRA 2024 in Yokohama, the simulation performance upper bound was raised from 0.25 to 0.5, after 0.25 had been closely approached by multiple teams. This year, the new 0.5 upper bound has itself been closely approached by at least one team: IN^{2}BOT, who achieved 0.4975 out of 0.5 and also went on to win the physical finals. EW-Glab finished second in both the simulation qualifier and the physical finals, marginally surpassing last year's simulation winner FSMT (0.4880 vs. 0.4878) while also improving on their own prior submission (0.4880 vs. 0.4736). The organizers are considering raising the performance upper bound once more if a comparable number of teams approach 0.5 in the coming year.

B. Increased Physical Difficulty Increases the Need for Fine-Tuning

This year, beginning from the first obstacle course, the physical environments were designed to be significantly more challenging than in previous years, both in terms of course length and obstacle clearance, as evidenced by the longer traversal times in Table II. As a result, fine-tuning became increasingly necessary. Across all 12 cold trials in the three obstacle courses with the four teams, only a single cold trial was successfully completed—by EW-Glab in the second obstacle course, where they also achieved notably strong performance overall (only one failed timed trial).

Autonomous navigation stacks that require no fine-tuning to adapt to different obstacle configurations remain a desirable but unrealized goal. Echoing the recommendation from previous years, researchers are encouraged to pursue parameter-free navigation systems, systems whose performance across different environments is insensitive to parameterization, or autonomous parameter tuning techniques that do not depend on human expertise [12], [19]–[24].

C. Second Consecutive Year With Classical Systems Across All Winning Teams

As in the Fourth BARN Challenge, all three winning teams this year again employed classical approaches without any machine learning. This trend stands in notable contrast to the growing prevalence of learning-based methods in robotics—particularly in manipulation and autonomous driving, where vision-language-action models and robot world models have drawn considerable recent attention. While learning-based navigation approaches, including reinforcement [25]–[29] and imitation [30]–[35] learning, have also been applied to navigation, the highly precise maneuvering demanded by the extremely constrained BARN obstacle courses continues to favor classical systems for their robustness, explainability, verifiability, debuggability, and, critically, their amenability to fine-tuning.

The organizers also wish to acknowledge that navigation challenges beyond geometric obstacle avoidance—such as visual inputs [25], [26], [36]–[41], off-road conditions [31], [32], [34], [35], [42]–[52], social contexts [53]–[63], kinodynamic constraints [64]–[66], and multi-robot navigation [67]–[70]—may better motivate the use of learning-based methods. Nevertheless, obstacle avoidance and goal-reaching remain the two most fundamental navigation capabilities of autonomous mobile robots, and ones that already underpin real-world deployments such as food delivery robots in restaurants, hotels, and neighborhoods, and logistics robots in factories and warehouses.

D. Fuzzy Logic Shines Again

A noteworthy recurrence this year is that the winning team, IN^{2}BOT, leveraged fuzzy logic in their navigation system—the same paradigm adopted by last year's winner, RRSL [71]. RRSL used fuzzy logic to directly generate motion commands (linear and angular velocities) via a two-stage optimized fuzzy controller, augmented by a search-smooth-safeguard procedure. IN^{2}BOT similarly employed fuzzy logic to regulate angular velocity based on heading error from the local planner and to modulate linear velocity according to frontal obstacle distance and required turning angle, with an additional safety barrier control layer applied on top of the fuzzy controller output. The back-to-back success of fuzzy-logic-based systems in the two most recent BARN Challenges suggests that fuzzy inference is a particularly well-suited paradigm for the fine-grained, uncertainty-aware control that tight obstacle courses demand.

E. Most Teams Optimize for Static Obstacles

The vast majority of teams continued to focus their navigation systems on static obstacle avoidance. Only one team, Team Robo, explicitly targeted dynamic obstacles in their design. This reinforces the finding from the Fourth BARN Challenge that dynamic obstacle avoidance remains an underdeveloped capability across the field.

F. Extensive Wireless Interference Causes Disruptions

During the physical finals, wireless interference was considerably more severe than in previous years and repeatedly disrupted teams' attempts to connect remotely to the robots. The likely primary source was the large number of competing teams in the vicinity of the BARN Challenge area, particularly participants in the RoboRacer competition. A further contributing factor was the RoboRacer audience: interference levels rose and fell in correlation with that competition's race schedule, peaking during races when spectators gathered in large numbers near the shared competition space.

Consequently, teams were affected to varying degrees. In the first obstacle course, the organizers paused the 20-minute countdown intermittently to mitigate the most severe disruptions—for instance, at times when connection quality degraded to the point that teams could not remotely adjust parameters, open RViz, or even set navigation goals. Maintaining a consistently fair standard for when to pause the countdown proved difficult and led to some contentious exchanges between teams and organizers. Beginning with the second obstacle course, all teams agreed to extend the session to 25 minutes with the countdown running continuously under all circumstances. While this provided a uniform standard, it exposed teams to unequal communication conditions depending on their time slot. For example, during the third obstacle course, IN^{2}BOT competed during a period of low RoboRacer spectator activity and experienced no wireless interruptions throughout their 25-minute session. EW-Glab, by contrast, competed during one of the RoboRacer semi-final races and was unable to connect to the robot for the majority of their session. The organizers will investigate technical and logistical options to reduce the impact of wireless interference at future competitions.

G. ROS Teams Significantly Outperform ROS2 Teams

As noted above, this year marked the first time ROS2 submissions were formally supported in the competition pipeline, and five of the 17 participating teams submitted ROS2-based systems (LAMA, LoneBot, VeRAN, Team K, and PUST_BARNARS). Of these, only one (LAMA) could be successfully evaluated using the standard ROS2 pipeline. In contrast, only one of the 12 ROS submissions (TUBE) failed to be evaluated. This disparity suggests that ROS remains the more mature and reliable platform for the BARN Challenge. It also likely reflects a demographic difference: ROS2 adopters tend to be newer entrants to the field building systems from scratch, whereas teams using ROS often maintain established code bases with legacy dependencies that anchor them to that ecosystem. One team, CRAL, used ROS for the simulation qualifier but intended to switch to ROS2 for the physical competition, illustrating the transitional nature of the current moment in the community's migration between the two frameworks.

VI. Future Plans

After five BARN Challenges since 2022, the competition rules have converged to a relatively stable state compared to the early iterations of the event. Accordingly, the core structure will be carried forward into the next competition: the simulation qualifier, the five-minute setup period, the cold trial as a bonus incentive for fine-tuning-free navigation, and the 20-minute (later expanded to 25 minutes) timed session will all be retained.

Given the consistently low engagement with dynamic obstacles across participating teams, the organizers will shift focus exclusively to static obstacle navigation for the foreseeable future. Rather than continuing to allocate competition time and scoring infrastructure to a dimension that few teams meaningfully optimize for, concentrating on static obstacle avoidance will allow the competition to more precisely benchmark the core capability it was designed to evaluate.

The most pressing organizational challenge for the next competition is developing a principled and fair policy for handling wireless interference during the physical finals. The current approach—extending the session length while running the countdown continuously regardless of connection conditions—provides consistency but cannot compensate for the unequal impact of interference on teams competing at different times. The organizers will explore options such as dedicated wireless channels, physical separation from neighboring competitions, or a structured and transparent protocol for pausing the countdown under objectively defined degradation conditions, with the goal of ensuring that results reflect navigation performance rather than communication fortune.

References

[1] X. Xiao, Z. Xu, Z. Wang, Y. Song, G. Warnell, P. Stone, T. Zhang, S. Ravi, G. Wang, H. Karnan *et al.*, "Autonomous ground navigation in highly constrained spaces: Lessons learned from the benchmark autonomous robot navigation challenge at icra 2022 [competitions]," *IEEE Robotics & Automation Magazine*, vol. 29, no. 4, pp. 148–156, 2022.

[2] X. Xiao, Z. Xu, G. Warnell, P. Stone, F. G. Guinjoan, R. T. Rodrigues, H. Bruyninckx, H. Mandala, G. Christmann, J. L. Blanco-Claraco *et al.*, "Autonomous ground navigation in highly constrained spaces: Lessons learned from the second barn challenge at icra 2023 [competitions]," *IEEE Robotics & Automation Magazine*, vol. 30, no. 4, pp. 91–97, 2023.

[3] X. Xiao, Z. Xu, A. Datar, G. Warnell, P. Stone, J. J. Damanik, J. Jung, C. A. Deresa, T. D. Huy, C. Jinyu *et al.*, "Autonomous ground navigation in highly constrained spaces: Lessons learned from the third barn challenge at icra 2024 [competitions]," *IEEE Robotics & Automation Magazine*, vol. 31, no. 3, pp. 197–204, 2024.

[4] X. Xiao, Z. Xu, S. A. Ghani, A. Datar, D. Song, P. Stone, A. Mazen, K. Yazdipaz, I. Mateyaunga, M. Faied, M. Krishnan, Y. Lu, T. Xu, N. Mohammad, W. Kim, J. Reasoner, R. Chandra, and N. Bezzo, "Autonomous ground navigation in highly constrained spaces: Lessons learned from the fourth barn challenge at icra 2025," *IEEE Robotics & Automation Magazine*, vol. 33, no. 1, pp. 186–196, 2026.

[5] "Jackal ugv - small weatherproof robot - clearpath," https://clearpathrobotics.com/jackal-small-unmanned-ground-vehicle/, accessed: 2022-07-21.

[6] D. Perille, A. Truong, X. Xiao, and P. Stone, "Benchmarking metric ground navigation," in *2020 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, 2020, pp. 116–121.

[7] A. Nair, F. Jiang, K. Hou, Z. Xu, S. Li, X. Xiao, and P. Stone, "Dynabarn: Benchmarking metric ground navigation in dynamic environments," in *2022 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, 2022, pp. 347–352.

[8] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997.

[9] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control," in *[1993] Proceedings IEEE International Conference on Robotics and Automation*. IEEE, 1993, pp. 802–807.

[10] Z. Xu, B. Liu, X. Xiao, A. Nair, and P. Stone, "Benchmarking reinforcement learning techniques for autonomous navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9224–9230.

[11] Z. Wang, X. Xiao, A. J. Nettekoven, K. Umasankar, A. Singh, S. Bommakanti, U. Topcu, and P. Stone, "From agile ground to aerial navigation: Learning from learned hallucination," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 148–153.

[12] Z. Xu, G. Dhamankar, A. Nair, X. Xiao, G. Warnell, B. Liu, Z. Wang, and P. Stone, "Applr: Adaptive planner parameter learning from reinforcement," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 6086–6092.

[13] "ROS movebase navigation stack," http://wiki.ros.org/move_base, accessed: 2021-09-9.

[14] "The barn challenge 2026," https://people.cs.gmu.edu/~xiao/Research/BARN_Challenge/BARN_Challenge26.html, accessed: 2026-06-23.

[15] C. Rösmann, W. Feiten, T. Wösch, F. Hoffmann, and T. Bertram, "Efficient trajectory optimization using a sparse model," in *2013 European Conference on Mobile Robots*. IEEE, 2013, pp. 138–143.

[16] C. Rösmann, F. Hoffmann, and T. Bertram, "Integrated online trajectory planning and optimization in distinctive topologies," *Robotics and Autonomous Systems*, vol. 88, pp. 142–153, 2017.

[17] H. Lee and Y. J. Kim, "Adaptive trajectory refinement for optimization-based local planning in narrow passages," *arXiv preprint arXiv:2510.26142*, 2025.

[18] R. Verschueren, G. Frison, D. Kouzoupis, J. Frey, N. van Duijkeren, A. Zanelli, B. Novoselnik, T. Albin, R. Quirynen, and M. Diehl, "acados – a modular open-source framework for fast embedded optimal control," *Mathematical Programming Computation*, 2021.

[19] X. Xiao, B. Liu, G. Warnell, J. Fink, and P. Stone, "Appld: Adaptive planner parameter learning from demonstration," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4541–4547, 2020.

[20] Z. Wang, X. Xiao, B. Liu, G. Warnell, and P. Stone, "Appli: Adaptive planner parameter learning from interventions," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 6079–6085.

[21] Z. Wang, X. Xiao, G. Warnell, and P. Stone, "Apple: Adaptive planner parameter learning from evaluative feedback," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7744–7749, 2021.

[22] H. Ma, J. S. Smith, and P. A. Vela, "Navtuner: Learning a scene-sensitive family of navigation policies," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 492–499.

[23] X. Xiao, Z. Wang, Z. Xu, B. Liu, G. Warnell, G. Dhamankar, A. Nair, and P. Stone, "Appl: Adaptive planner parameter learning," *Robotics and Autonomous Systems*, vol. 154, p. 104132, 2022.

[24] Y. Lu, B. Wang, Z. Wu, Y. Li, X. Lin, C. Mao, and X. Xiao, "Applv: Adaptive planner parameter learning from vision-language-action model," in *2026 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2026.

[25] G. Kahn, P. Abbeel, and S. Levine, "Badgr: An autonomous self-

supervised learning-based navigation system," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1312–1319, 2021.

[26] H. Karnan, G. Warnell, X. Xiao, and P. Stone, "Voila: Visual-observation-only imitation learning for autonomous navigation," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2497–2503.

[27] Z. Xu, B. Liu, X. Xiao, A. Nair, and P. Stone, "Benchmarking reinforcement learning techniques for autonomous navigation," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 9224–9230.

[28] Z. Xu, X. Xiao, G. Warnell, A. Nair, and P. Stone, "Machine learning methods for local motion planning: A study of end-to-end vs. parameter learning," in *2021 IEEE International Symposium on Safety, Security, and Rescue Robotics (SSRR)*. IEEE, 2021, pp. 217–222.

[29] Z. Xu, A. Nair, X. Xiao, and P. Stone, "Learning real-world autonomous navigation by self-supervised environment synthesis," *arXiv preprint arXiv:2210.04852*, 2022.

[30] Y. Pan, C.-A. Cheng, K. Saigol, K. Lee, X. Yan, E. A. Theodorou, and B. Boots, "Imitation learning for agile autonomous driving," *The International Journal of Robotics Research*, vol. 39, no. 2-3, pp. 286–302, 2020.

[31] X. Xiao, J. Biswas, and P. Stone, "Learning inverse kinodynamics for accurate high-speed off-road navigation on unstructured terrain," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 6054–6060, 2021.

[32] H. Karnan, K. S. Sikand, P. Atreya, S. Rabiee, X. Xiao, G. Warnell, P. Stone, and J. Biswas, "Vi-ikd: High-speed accurate off-road navigation using learned visual-inertial inverse kinodynamics," in *To Appear in 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022.

[33] P. Atreya, H. Karnan, K. S. Sikand, X. Xiao, G. Warnell, S. Rabiee, P. Stone, and J. Biswas, "High-speed accurate robot control using learned forward kinodynamics and non-linear least squares optimization," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022.

[34] A. Datar, C. Pan, M. Nazeri, and X. Xiao, "Toward wheeled mobility on vertically challenging terrain: Platforms, datasets, and algorithms," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024.

[35] A. Datar, C. Pan, and X. Xiao, "Learning to model and plan for wheeled mobility on vertically challenging terrain," *IEEE Robotics and Automation Letters*, 2024.

[36] D. Song, J. Liang, X. Xiao, and D. Manocha, "Vl-tgs: Trajectory generation and selection using vision language models in mapless outdoor environments," *IEEE Robotics and Automation Letters*, 2025.

[37] J. Liang, K. Weerakoon, D. Song, S. Kirubaharan, X. Xiao, and D. Manocha, "Mosu: Autonomous long-range robot navigation with multi-modal scene understanding," in *19th International Symposium on Experimental Robotics (ISER)*, 2025.

[38] A. Payandeh, D. Song, M. Nazeri, J. Liang, P. Mukherjee, A. H. Raj, Y. Kong, D. Manocha, and X. Xiao, "Social-llava: Enhancing robot navigation through human-language reasoning in social spaces," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.

[39] Y. Kong, D. Song, J. Liang, D. Manocha, Z. Yao, and X. Xiao, "Autospatial: Visual-language reasoning for social robot navigation through efficient spatial reasoning learning," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.

[40] M. Nazeri, J. Wang, A. Payandeh, and X. Xiao, "Vanp: Learning where to see for navigation with self-supervised vision-action pre-training," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024.

[41] K. S. Sikand, S. Rabiee, A. Uccello, X. Xiao, G. Warnell, and J. Biswas, "Visual representation learning for preference-aware path planning," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 11 303–11 309.

[42] T. Xu, C. Pan, and X. Xiao, "Vertiselector: Automatic curriculum learning for wheeled mobility on vertically challenging terrain," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.

[43] A. Datar, A. Pokhrel, M. Nazeri, M. B. Rao, C. Pan, Y. Zhang, A. Harrison, M. Wigness, P. R. Osteen, J. Ye, and X. Xiao, "M2p2: A multi-modal passive perception dataset for off-road mobility in extreme low-light conditions," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.

[44] M. Gupta and X. Xiao, "T-cbf: Traversability-based control barrier function to navigate vertically challenging terrain," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.

[45] A. Pokhrel, A. Datar, and X. Xiao, "Dom, cars don't fly!–or do they? in-air vehicle maneuver for high-speed off-road navigation," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.

[46] X. Cai, J. Queeney, T. Xu, A. Datar, C. Pan, M. Miller, A. Flather, P. R. Osteen, N. Roy, X. Xiao *et al.*, "Pietra: Physics-informed evidential learning for traversing out-of-distribution terrain," *IEEE Robotics and Automation Letters*, 2025.

[47] A. Datar, C. Pan, M. Nazeri, A. Pokhrel, and X. Xiao, "Terrain-attentive learning for efficient 6-dof kinodynamic modeling on vertically challenging terrain," *arXiv preprint arXiv:2403.16419*, 2024.

[48] M. Nazeri, A. Datar, A. Pokhrel, C. Pan, G. Warnell, and X. Xiao, "Verticoder: Self-supervised kinodynamic representation learning on vertically challenging terrain," in *2024 ieee international conference on robotics and automation (ICRA)*. IEEE, 2024.

[49] T. Xu, C. Pan, and X. Xiao, "Reinforcement learning for wheeled mobility on vertically challenging terrain," in *2024 IEEE International Symposium on Safety Security Rescue Robotics (SSRR)*. IEEE, 2024, pp. 125–130.

[50] J. Liang, A. Payandeh, D. Song, X. Xiao, and D. Manocha, "Dtg: Diffusion-based trajectory generation for mapless global navigation," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024.

[51] J. Liang, P. Gao, X. Xiao, A. J. Sathyamoorthy, M. Elnoor, M. Lin, and D. Manocha, "Mtg: Mapless trajectory generator with traversability coverage for outdoor navigation," in *2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE*, 2024.

[52] A. Pokhrel, M. Nazeri, A. Datar, and X. Xiao, "Cahsor: Competence-aware high-speed off-road ground navigation in se(3)," *IEEE Robotics and Automation Letters*, 2024.

[53] A. H. Raj, Z. Hu, H. Karnan, R. Chandra, A. Payandeh, L. Mao, P. Stone, J. Biswas, and X. Xiao, "Rethinking social robot navigation: Leveraging the best of two worlds," in *2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE*, 2024.

[54] D. Song, J. Liang, A. Payandeh, A. H. Raj, X. Xiao, and D. Manocha, "Vlm-social-nav: Socially aware robot navigation through scoring using vision-language models," *IEEE Robotics and Automation Letters*, 2024.

[55] B. Panigrahi, A. H. Raj, M. Nazeri, and X. Xiao, "A study on learning social robot navigation with multimodal perception," *arXiv preprint arXiv:2309.12568*, 2023.

[56] R. Mirsky, X. Xiao, J. Hart, and P. Stone, "Conflict avoidance in social navigation—a survey," *ACM Transactions on Human-Robot Interaction*, vol. 13, no. 1, pp. 1–36, 2024.

[57] A. Francis, C. Pérez-d'Arpino, C. Li, F. Xia, A. Alahi, R. Alami, A. Bera, A. Biswas, J. Biswas, R. Chandra *et al.*, "Principles and guidelines for evaluating social robot navigation algorithms," *ACM Transactions on Human-Robot Interaction*, 2024.

[58] C. Mavrogiannis, F. Baldini, A. Wang, D. Zhao, P. Trautman, A. Steinfeld, and J. Oh, "Core challenges of social robot navigation: A survey," *ACM Transactions on Human-Robot Interaction*, vol. 12, no. 3, pp. 1–39, 2023.

[59] H. Karnan, A. Nair, X. Xiao, G. Warnell, S. Pirk, A. Toshev, J. Hart, J. Biswas, and P. Stone, "Socially compliant navigation dataset (scand): A large-scale dataset of demonstrations for social navigation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 807–11 814, 2022.

[60] D. M. Nguyen, M. Nazeri, A. Payandeh, A. Datar, and X. Xiao, "Toward human-like social robot navigation: A large-scale, multi-modal, social human navigation dataset," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 7442–7447.

[61] X. Xiao, T. Zhang, K. M. Choromanski, T.-W. E. Lee, A. Francis, J. Varley, S. Tu, S. Singh, P. Xu, F. Xia, S. M. Persson, L. Takayama, R. Frostig, J. Tan, C. Parada, and V. Sindhwani, "Learning model predictive controllers with real-time attention for real-world navigation," in *Conference on robot learning*. PMLR, 2022.

[62] J. Hart, R. Mirsky, X. Xiao, S. Tejeda, B. Mahajan, J. Goo, K. Baldauf, S. Owen, and P. Stone, "Using human-inspired signals to disam-

biguate navigational intentions," in *International Conference on Social Robotics*. Springer, 2020, pp. 320–331.

[63] S. Pirk, E. Lee, X. Xiao, L. Takayama, A. Francis, and A. Toshev, "A protocol for validating social navigation policies," *arXiv preprint arXiv:2204.05443*, 2022.

[64] D. Das, Y. Lu, E. Plaku, and X. Xiao, "Motion memory: Leveraging past experiences to accelerate future motion planning," in *2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE*, 2024.

[65] Y. Lu and E. Plaku, "Leveraging single-goal predictions to improve the efficiency of multi-goal motion planning with dynamics," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 850–857.

[66] H.-D. Bui, Y. Lu, and E. Plaku, "Improving the efficiency of sampling-based motion planners via runtime predictions for motion-planning problems with dynamics," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 4486–4491.

[67] P. Long, T. Fan, X. Liao, W. Liu, H. Zhang, and J. Pan, "Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 6252–6259.

[68] Y. F. Chen, M. Liu, M. Everett, and J. P. How, "Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning," in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 285–292.

[69] Y. Lu, D. Das, E. Plaku, and X. Xiao, "Multi-goal motion memory," *arXiv preprint arXiv:2407.11399*, 2024.

[70] J.-S. Park, X. Xiao, G. Warnell, H. Yedidsion, and P. Stone, "Learning perceptual hallucination for multi-robot navigation in narrow hallways," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 10 033–10 039.

[71] A. Mazen, M. Faied, M. Krishnan, K. Yazdipaz, and I. Mateyaunga, "A two-stage tuning framework for fuzzy navigation: From intuitive fisvfh to robust S^3-FISVFH," *IEEE Transactions on Fuzzy Systems*, 2025, [under review].