

Autonomous Ground Navigation in Highly Constrained Spaces: Lessons Learned from The Forth BARN Challenge at ICRA 2025

Competition Organizers: Xuesu Xiao¹, Zifan Xu², Saad Abdul Ghani¹,
Aniket Datar¹, Daeun Song¹, and Peter Stone^{2,3},

Winning Teams: Amna Mazen⁴, Kamyab Yazdipaz⁴, Innocent Mateyaunga⁴,
Mariam Faied⁵, Mohan Krishnan⁵,

Yuanjie Lu¹, Tong Xu¹,

Nick Mohammad⁶, Woosung Kim⁶, Jon Reasoner⁶, and Nicola Bezzo⁶

Abstract—The forth Benchmark Autonomous Robot Navigation (BARN) Challenge took place at the 2025 IEEE International Conference on Robotics and Automation (ICRA 2025) in Atlanta, GA, USA and continued to evaluate the performance of state-of-the-art autonomous ground navigation systems in highly constrained environments. It is the first time The BARN Challenge came back to North American since the first challenge in Philadelphia, after going around the world to London (Europe) and Yokohama (Asia). Nine teams participated in the simulation competition, eight of which were invited to the physical competition, while four of them finally attended in Atlanta. A few changes were adopted for the first time, such as introducing dynamic obstacles and adjusting the system tuning rules, to encourage more robust navigation performance. 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 FORTH BARN CHALLENGE OVERVIEW

The forth BARN Challenge took place as a conference competition at ICRA 2025 in Atlanta, GA, USA. As a continuation of the first, second, and third BARN Challenge at ICRA 2022, 2023, and 2024 in Philadelphia, London, and Yokohama respectively, the forth 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]–[3].

Each team needed to develop an entire navigation software stack for a standardized and provided mobile robot, i.e., a Clearpath Jackal [4] 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 forth 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 2025 conference using the BARN dataset [5] (with the recent addition of DynaBARN [6]), which is composed of 300 obstacle courses in Gazebo simulation randomly generated by cellular automata. The top eight teams from the simulation phase were then invited to compete in three different physical obstacle courses set up by the organizers at ICRA 2025 in the GWCC Atlanta conference center. A physical DynaBARN arena has been introduced for the first time, with other rule changes to encourage more robust navigation in highly constrained spaces.

In this article, we report on the simulation qualifier and physical finals of The forth BARN Challenge at ICRA 2025, present the approaches used by the top three teams, discuss lessons learned from the challenge compared against the first, second, and third BARN Challenge at ICRA 2022, 2023, and 2024, 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 forth BARN Challenge started on January 1st, 2025. The qualifier used the BARN dataset [5], which consists of 300 5m × 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

¹George Mason University ²The University of Texas at Austin ³Sony AI ⁴Michigan Technological University ⁵University of Detroit Mercy ⁶University of Virginia

¹<https://github.com/dperille/jackal-map-creation>

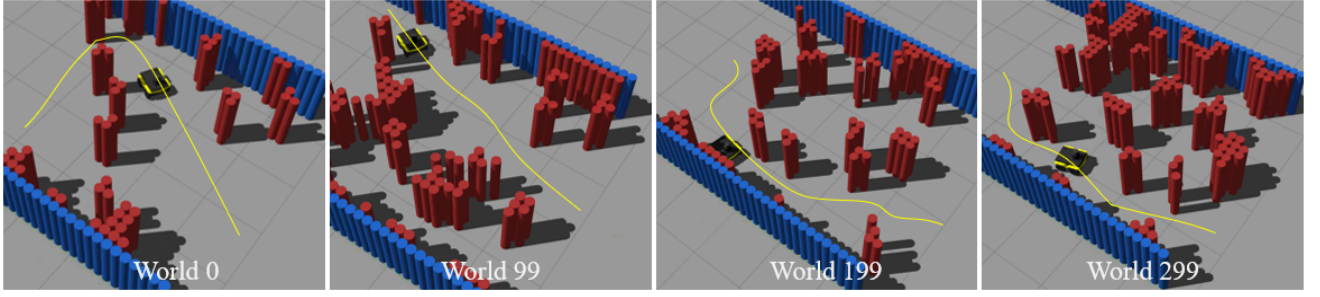


Fig. 1: Four example BARN environments in the Gazebo simulator (ordered by ascending relative difficulty level).

In addition to the 300 BARN environments, six baseline approaches were also provided for the participants’ reference, ranging from classical sampling-based [7] and optimization-based navigation systems [8], to end-to-end machine learning methods [9], [10], and hybrid approaches [11]. 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 [12]. Additionally, the winning teams’ navigation stacks from the last three competitions were also open sourced [13]. 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

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 pipeline³ to run each team’s navigation system and compute a standardized performance metric that considers navigation success rate (collision or not reaching the goal counts 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_{i}^{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 disproportionately 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, 2025 and lasted through a soft submission deadline (April 1st, 2025) and a hard submission deadline (May 1st, 2025). 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 Atlanta. 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, nine teams, four from North America, four from Asia, and one from Europe, 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. For The forth BARN Challenge, 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.

The top two simulation teams, FSMT, an individual participant, and RobotiXX from George Mason University (GMU) outperformed all last year’s winning teams (LiCS-KI, MLDA.EEE, and AIMS). However, there is still a gap between the performance upper bound (0.5) and the top performance (0.4878).

The top eight teams were invited to the physical finals at ICRA 2025, of which only four—RobotiXX from GMU, USA, EW-Glab from Ewha Womans University, Korea, Robotics and Remote Sensing Lab (RRSL) from Michigan Technological University (MTU), USA, and Autonomous

²https://github.com/Daffan/ros_jackal

³<https://github.com/Daffan/nav-competition-icra2022>

TABLE I: Simulation Results.

Rank.	Team	Score	Baseline
1	FSMT	0.4878 (0.0000)	
2	RobotiXX	0.4873 (0.1042)	
3	EW-Glab	0.4736 (0.0000)	
4	LiCS-KI	0.4611 (0.1125)	
5	MDP-DWA	0.4525 (0.0000)	
6	MLDA-NTU	0.4510 (0.0201)	LfLH [10], e2e [9]
7	RRSL	0.3558 (0.0001)	
8	UVA AMR	0.3475 (0.0000)	APPLR-DWA [11], E-Band [8], (Fast & Default) DWA [7]
9	Armans Team	NA (NA)	

Mobile Robotics Lab (AMR) from The University of Virginia, USA—could attend.

III. PHYSICAL FINALS

The physical finals took place at ICRA 2025 in the GWCC Atlanta conference center on May 21st and May 22nd, 2025 (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, sponsor (Clearpath Robotics), and organizers at The forth BARN Challenge in Atlanta, GA, USA.

A. Setup

Physical obstacle courses were set up using 225 cardboard boxes in the conference center. The organizers used the same guidelines to set up three obstacle courses as in the first three 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.

This year, however, dynamic obstacles were added at the end of the static obstacle course to further test the robustness of planning systems. A set of (6-10) randomly moving iRobot Creates were placed in an enclosed area through which the Jackal had to maneuver through. The Creates had a 2 foot cylinder placed on top to allow the Jackal’s Hokuyo LiDAR to sense them (Fig. 2). Each Roomba was 0.33 m in diameter and moved at a maximum linear speed of 0.5 m/s with a combination of behaviors such as spiraling outwards, following walls, and bouncing off each other.

B. Rules

In order to discourage onsite system fine-tuning to fit the navigation stack specifically to one obstacle course and to accommodate the newly introduced DynaBARN arena, the organizers modified the physical competition rules compared to previous years’, which were also adjusted throughout the competition after discussions with and consensus of all participating teams.

Each team has five minutes to set up their navigation systems after they received the robot without accessing each obstacle course. Different than previous years, in the first obstacle course, after the 5-minute setup time, each team directly ran five timed trials one after another without fine-tuning. While only one team, RRSL, was able to finish trials without any fine-tuning for the first obstacle course, all other teams failed to do so. Therefore, starting from the second 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, 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.

Different than previous years, a trial would be successful if the Jackal could maneuver through the static obstacle course without any “hard collisions”. A collision would be considered “hard” if it caused a cardboard box to move from its original position, but not if the Jackal simply grazed a cardboard box without causing it to move. The traversal time for the static obstacle course would be recorded and used for tie-breaking.

After each static obstacle course, the Jackal would enter the physical DynaBARN arena. Collisions with dynamic obstacles would not be penalized, but clear dynamic obstacle avoidance behavior (such as stopping, reversing, or circumventing) was rewarded by a 5% decrease in traversal time of the static obstacle course to encourage robust obstacle avoidance behavior when facing moving obstacles.

TABLE II: Physical Results.

Rank.	Team	Success/Total	Average Time	Course 1	Course 2	Course 3
1	RRSL	7/9(+2)	43/56/105	49/49/46(49)/43/41(43)	59/51/X/X/59(62)/X	X/X/X/X/105/X
2	RobotiXX	3/9(+2)	NA/83/151	X/X/X/X/X	X/70/X/X/X/95	X/X/X/X/151(159)
3	UVA AMR	1/9(+2)	NA/57/NA	X/X/X/X/X	X/X/57/X/X/X	X/X/X/X/X/X
4	EW-Glab	1/9(+2)	NA/92/NA	X/X/X/X/X	X/92(97)/X/X/X/X	X/X/X/X/X/X

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. If a trial was given the dynamic obstacle avoidance reward the raw time is shown in brackets and the adjusted time is shown outside. The (+2) in the total runs indicate the two extra cold trials present in courses two and three.

The winner, RRSL, successfully and quickly finished all five back-to-back trials in the first course and outperformed the other teams, while grazing obstacles a few times without incurring any hard collision. However, they faced difficulty in the second course and the third most difficult one. The runner-up, RobotiXX, went through the second and third obstacle courses twice and once respectively without any hard or soft collision. UVA AMR was able to successfully finish a timed trial in the second course but struggled with all other trials, possibly due to a bug caused by the sensor's input. EW-Glab was also only able to successfully finish a timed trial in the second course, possibly due to sensor-level noise and sparse sampling in their planning algorithm, which made it difficult for the robot to perceive narrow gaps and plan safe trajectories.

As a result, RRSL won the competition by the most successful trials (7/9); RobotiXX came in second place with the second most successful trials (3/9); and the tie between UVA AMR and EW-Glab was broken by the average traversal time (57s vs. 92s).

IV. TOP THREE TEAMS AND APPROACHES

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

A. RRSL (MTU)

Robotics and Remote Sensing Lab (RRSL) proposed a fuzzy-based navigation algorithm termed as the Search-Smooth-Safeguard FISVFH (S^3 -FISVFH) [15]. The backbone of the proposed navigation framework is the Fuzzy Inference System Vector Field Histogram (FISVFH) algorithm [16]. FISVFH [16] integrates a Fuzzy Inference System (FIS) with the VFH algorithm [17], which is a real-time navigation algorithm for mobile wheeled robots. A FIS employs human-like reasoning to manage uncertainty and make decisions from imprecise data. This combination leverages the VFH's ability to make real-time path corrections using sensor data, while the FIS effectively handles uncertainties such as

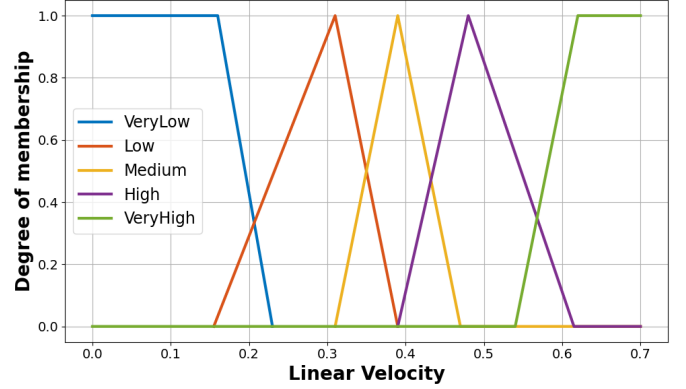


Fig. 3: Optimized MFs for the robot's linear velocity controller after tuning the FISVFH [16] algorithm

sensor noise and environmental disturbances. The FISVFH algorithm comprises two FIS controllers: one generates the robot's linear velocity using front-facing LiDAR data, while the other determines its angular velocity based on the robot's current heading and the direction toward the goal.

S^3 -FISVFH [15] was developed in two main stages. First, the FISVFH controller was tuned to optimize its fuzzy inference system parameters. In the second stage, enhancements were introduced to address key limitations identified in the original FISVFH approach.

1) **FISVFH Tuning Procedure:** A FIS controller is defined by five key parameters: the number of membership functions (MFs), the type of MFs, the range of each MF, and the precise shape and position of each MF. The original FISVFH [16] relied on heuristic, intuition-driven selection of these parameters. To improve consistency and performance, a structured two-stage framework was developed to optimize all five parameters systematically. In the first stage, a Design of Experiments (DoE) approach was employed to determine the optimal high-level structure statistically [18]. This process focused on the first three parameters, analyzing various configurations for the number, type, and operational range of the MFs. The second stage focused on data-driven optimization of the shape and position of the MFs using an optimization-based planner, the Bézier-based Trajectory Planner (BTP) [19] as a benchmark. Fig. 3 shows the tuned MFs of the robot's linear velocity. Although the Tuned FISVFH performed well overall, it remained susceptible to certain failure scenarios, including getting trapped, exhibiting oscillatory behavior, and failing to stop when faced with imminent collision scenarios.

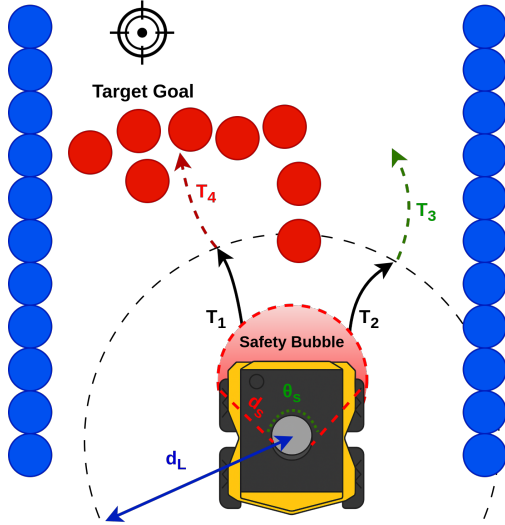


Fig. 4: The S^3 -FISVFH algorithm uses the Search component to evaluate the possible path to avoid traps.

2) **The S^3 -FISVFH Enhancement:** The S^3 -FISVFH algorithm integrates three critical components, Search, Smooth, and Safeguard, on top of the tuned controller to address the previously mentioned limitations of Tuned FISVFH:

- **Search:** A lookahead search mechanism uses a custom version of A* algorithm [20] to prevent the robot from getting stuck in traps. As illustrated in Fig. 4, a purely reactive planner might select trajectory T_1 , as it is most aligned with the target goal. However, this path leads directly into a trap (T_4). The Search component simulates trajectories beyond the limited sensor horizon, allowing it to foresee this trap. It evaluates alternative paths and identifies that trajectory T_2 , while initially deviating from the goal, leads to a viable path (T_3).
- **Smooth:** A temporal smoothing filter is applied to mitigate the oscillatory behavior in Tuned FISVFH due to rapid changes in the robot's environment. This component produces more stable turning and continuous motion by using a weighted average of the robot's current and previous headings.
- **Safeguard:** A fundamental limitation of the original FISVFH is its inability to issue a stop or reverse command, even for imminent collisions. To address this, the Safeguard protocol introduces a "Safety Bubble" as illustrated by the shaded red sector in front of the robot in Fig. 4. If an obstacle is detected in this zone, this component overrides the FIS, stops, and reverses the robot.

B. RobotiXX (GMU)

The RobotiXX team from GMU introduced a novel Decremental Dynamics Planning (DDP) paradigm [21], which dynamically incorporates dynamic constraints throughout the entire planning framework. This approach employs a

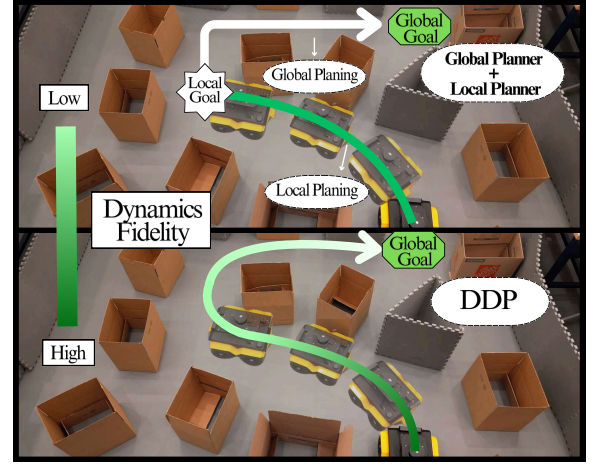


Fig. 5: Contrasting the traditional global and local planning paradigm (top), where either full (green) or zero (white) robot dynamics is considered, DDP starts with high fidelity dynamics in the early part of trajectory rollout and gradually decreases dynamics fidelity for computation efficiency (bottom).

progressive fidelity reduction strategy that balances computational efficiency with modeling accuracy.

Specifically, DDP starts with high-fidelity dynamics modeling in the early trajectory rollout stages, capturing essential dynamic properties of the robot, e.g., velocity, acceleration, and turning radius constraints, and ensuring that the robot can precisely navigate complex environments and avoid highly constrained obstacles. As the trajectory rollout progresses, the fidelity of dynamic modeling gradually decreases by simplifying the model to improve computational efficiency while ensuring that dynamic feasibility is not significantly compromised (Fig. 5). In the implementation, DDP balances dynamics fidelity and onboard computation by adjusting both (1) the dynamics integration interval and (2) the number of robot state points for state-space collision checking at each time step. This approach augments the $\mathbb{SE}(2)$ robot state (x_t, y_t, ψ_t) to $(x_t, y_t, \psi_t, x_t^1, y_t^1, \dots, x_t^n, y_t^n)$, where $(x_t^i, y_t^i)_{i=1}^n$ are n state points on the robot boundary for collision checking at step t . The robot dynamics parameters are $\theta_t = (\Delta_t, c_t^1, c_t^2, \dots, c_t^n)$, where Δ_t denotes the dynamics integration interval at step t and $\{c_t^i\}_{i=1}^n$ are binary indicators of whether the position of the i -th state point (x_t^i, y_t^i) should be calculated and collision-checked at step t ($c_t^i = 1$) or not ($c_t^i = 0$). To gradually increase the dynamics integration interval over the rollout time \mathcal{T} (in seconds), the following function is employed:

$$\Delta_t = \mathcal{T} \cdot \left[\left(\frac{t+1}{T} \right)^p - \left(\frac{t}{T} \right)^p \right], \quad (1)$$

where T denotes the number of time steps, and p is a hyperparameter controlling the rate of change in the integration interval. At each integration interval, only the subset of collision points defined by $\{c_t^i\}_{i=1}^n$ are evaluated. The number of points checked along the trajectory (N_t) is

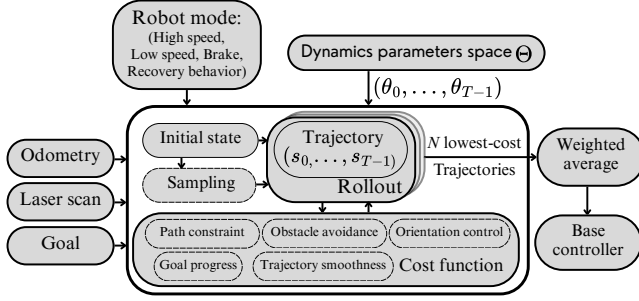


Fig. 6: RobotiXX DDP-based navigation system.

gradually reduced using:

$$N_t = n \cdot \left(1 - \left(\frac{t}{T}\right)^p\right). \quad (2)$$

Only state points i , for which $c_t^i = 1$, are calculated and collision-checked at time t to achieve decremental dynamics and computational savings.

The DDP-based navigation system allows the robot to operate in four distinct modes: high-speed, low-speed, braking, and recovery (including rotation and reverse). The robot begins in high-speed mode and shifts to low-speed mode when its linear velocity remains below a threshold for a set duration, reducing the maximum linear and angular velocity limits for safe obstacle avoidance. If the linear velocity stays too low, the robot brakes before entering recovery behavior. Similarly, when the robot's speed exceeds a specified threshold, it transitions from recovery to low-speed mode and then back to high-speed mode if necessary. The navigation system framework is illustrated in Fig. 6. For each planning iteration, the robot randomly samples linear and angular velocities with added noise. During trajectory rollout, the system employs these velocity samples and robot dynamics parameters to predict potential future trajectories and evaluates them against a cost function. The cost function considers multiple factors: proximity to the goal, distance from obstacles, total path length, trajectory smoothness, and orientation relative to the goal. After evaluating all trajectories, only the $N = 10$ collision-free trajectories with the lowest cost are retained, and robot actions are generated using a weighted average of their linear and angular velocities based on trajectory cost.

C. UVA AMR (UVA)

The UVA AMR's implementation builds on the lab's prior work [22], [23] and consists of four main elements: 1) a static obstacle avoidance planner, 2) a control strategy, 3) a dynamic obstacle detection approach, and 4) a dynamic obstacle avoidance method.

1) Static Obstacle Avoidance—Planning: For planning purposes, the pipeline leverages a safe-corridor-based motion planner [24] to generate dynamically feasible trajectories that a low-level controller can track. Consider a robot tasked with reaching a final goal x_g in a cluttered environment, as illustrated in Fig. 7. The receding horizon planning pipeline begins by computing a 0-order global path (purple line) from

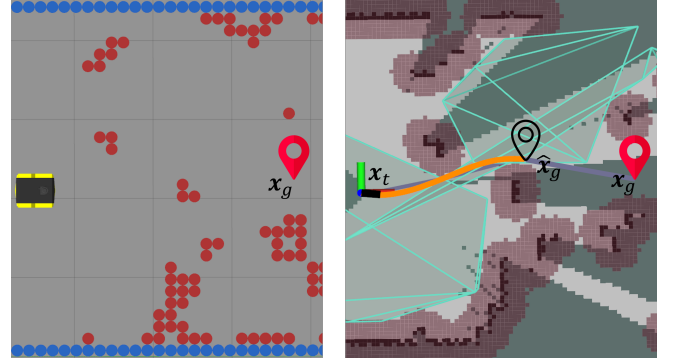


Fig. 7: Illustration of UVA AMR's static obstacle avoidance strategy.

the current robot position x_t to the goal x_g using a graph-based search method, such as Jump Point Search (JPS) [25].

To avoid frequent failures in the back-end trajectory optimizer when planning over excessively long horizons, the global path is truncated by walking along the JPS path by a fixed user-defined length to determine an intermediate goal \hat{x}_g . The Fast Iterative Region Inflation (FIRI) algorithm [26] is then used to generate corridor \mathcal{C} of intersecting convex polytopes along the path from the robot position x_t to goal x_g . This safe corridor, along with the robot's current state x_t and intermediate goal \hat{x}_g , is provided to the back-end optimizer, which solves for a smooth, dynamically feasible trajectory reference $r(\xi)$ parameterized by arc length ξ (orange curve in Fig. 7).

For mapping, UVA AMR used the `slam_toolbox`⁴ ROS package. Local obstacle information is captured using a sliding window costmap centered on the robot. To account for the robot's footprint, AMR dilated the obstacles in the costmap to ensure safe clearance during planning. In practice, the dilation radius was found to be one of the most critical parameters in the physical competition—too small and the robot risks collision; too large and planning becomes overly conservative, leading to deadlocks.

2) Static Obstacle Avoidance—Control: To reliably track $r(\cdot)$, AMR used a Model Predictive Contour Controller (MPCC) to generate a receding horizon of control inputs (black line extending from the robot in Fig. 7). Unlike traditional time-based trajectory tracking, which often relies on smooth polynomial curves that limit the robot's ability to sustain maximum speed over an entire segment, AMR's arc-length parameterized formulation allows the MPCC to freely modulate acceleration along the reference path $r(\xi)$. This provides more expressive control and enables faster, more efficient navigation by removing the need for time-parameterized polynomial curves.

The MPCC optimizes over acceleration commands to ensure smooth actuation, which are then integrated into velocity commands before being sent to the Jackal. To further improve tracking performance—particularly during aggressive maneuvers like high speed turns—AMR augmented

⁴https://github.com/SteveMacenski/slam_toolbox

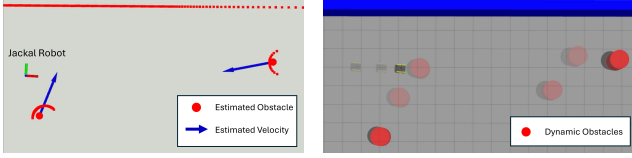


Fig. 8: Rviz and Gazebo screen captures of dynamic obstacle detection and backup maneuver.

the MPCC with a Control Lyapunov Function constraint. This introduces a stabilizing term that actively corrects tracking deviations, ensuring reliable and safe trajectory tracking, which is critical for navigating cluttered environments. Although not implemented during this competition, the robustness of this approach could be further improved by incorporating a Control Barrier Function [27] safety filter, as demonstrated in prior work [23]. In particular, collisions (e.g., clipping corners) were due to insufficient tuning of the obstacle padding in the local costmap, underscoring the need for additional safety enforcement.

3) Dynamic Obstacle Detection: Our methodology for dynamic obstacle detection relies on two key assumptions: 1) dynamic obstacles used in the competition are circular and 2) their motion is characterized by bounded and consistent speeds in the short term. The process begins by segmenting the LiDAR scan into point clusters and identifying dynamic obstacle candidates that satisfy the circular object assumption. These candidates are then passed to a tracking-by-detection algorithm, which associates them with previously identified tracks across consecutive frames [28]. A dynamic obstacle candidate is classified as a dynamic obstacle only if it meets the following criteria: (i) consistent and sustained movement is detected, (ii) size and shape requirements are within bounds, and (iii) the expected velocity of the candidate is within bounds. To refine the state of dynamic obstacles, the system employs a Kalman filter that leverages a constant-velocity model. This filter predicts each obstacle's future position while incorporating new measurements to correct the estimate. The result is a smoothed trajectory and a stable velocity estimate for each dynamic obstacle, providing robust estimation for downstream dynamic obstacle avoidance, even in the presence of noisy sensor data.

4) Dynamic Obstacle Avoidance: The robot's primary response to a predicted collision is to override the MPCC and publish a zero-velocity command, bringing the robot to a stop. However, this can result in a critical failure case where a dynamic obstacle continues along a collision path with the stationary robot. To address this issue, the robot employs a more active avoidance strategy by triggering an evasive backup maneuver until no collision is predicted. The outcome of this maneuver is shown in Fig. 8. Once the dynamic obstacle is no longer a collision threat, our approach validates that the MPCC trajectory is safe and hands control back to the primary planner, enabling the robot to resume its mission towards x_g .

V. DISCUSSION

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

A. Required fine-Tuning for each obstacle course

The organizers and teams decided to change the rules at the beginning of the physical competition to disallow teams to fine-tune for each obstacle course and incentivize reliability of the developed navigation systems in a variety of deployment scenarios. Unfortunately, such a rule change turned out to be too soon for the BARN Challenge. Even RSSL was able to achieve five successful trials without any fine-tuning in the first obstacle course, the Jackal gently touched the cardboard boxes without displacing them in two trials. The RobotiXX, UVA AMR, and EW-Glab teams were not able to finish one single trial without any hard collision in the first obstacle course. Such unsatisfactory behavior motivated the organizers and teams to agree to revert to the original rules, i.e., allowing fine-tuning to achieve the three best out of five timed trials within 20 minutes, with the cold trial as a bonus to reward navigation without fine-tuning for every environment. Interestingly, not a single team was able to successfully traverse the more difficult obstacle courses two and three without any fine-tuning (except RSSL for course two), all failing to get the bonus successful trial.

This new experience in the forth BARN Challenge reveals the gap between the existing navigation systems and an ideal one that can work without requiring human expertise across different deployment scenarios. Therefore, the organizers plan to continue using similar incentives to discourage fine-tuning for future BARN challenges, such as disallowing any fine-tuning in the first easiest obstacle course and using the cold trial to reward navigation without fine-tuning. Researchers are encouraged to develop parameter-free navigation systems, systems whose performances in different environments are insensitive to different parameterizations, or autonomous parameter tuning techniques without requiring human expertise [11], [29]–[33].

B. First year with classical systems across all winning teams

The forth BARN Challenge is the first year, in which all winning teams adopted classical approaches without any learning [34], as well as EW-Glab. LiCS-KI from Korea Advanced Institute of Science and Technology utilized a learning-based method. One of the team members was present at the competition in person but unfortunately was unable to participate due to illness. LiCS-KI's performance this year in simulation using reinforcement learning [35]–[39] was not as good as last year's using imitation learning [40]–[45], but with the hope to achieve better sim-to-real transfer and therefore better real-world performance. Unfortunately, the system's performance was not tested in the physical finals.

Notice LiCS-KI's Transformer-based method using imitation learning has won last year's BARN challenge. Learning method also has the potential to address challenges beyond

geometric obstacle avoidance [10], [46]–[49], such as visual inputs [35], [36], [50]–[55], off-road conditions [41], [42], [44], [45], [56]–[66], social contexts [67]–[77], kinodynamic constraints [78]–[80], or multi-robot navigation [81]–[84].

C. Unsatisfactory dynamic obstacle avoidance

Since most teams did not explicitly optimize their navigation systems for dynamic obstacle avoidance, they did not achieve good navigation performance in the simulated DynaBARN environments (brackets in Table I). Only RobotiXX and LiCS-KI were able to achieve a reasonable amount of dynamic obstacle avoidance behaviors (> 0.1). A few teams were not able to avoid any dynamic obstacle in DynaBARN.

During the physical competition, due to the poor performance in the static obstacle courses, the Jackal did not reach the dynamic obstacle field at the end in most trials. In a few trials when Jackal did reach, it did not encounter any dynamic obstacle on the way to the final goal, therefore not receiving the 5% time reduction reward on the static traversal time. In fact, the reduction rewards were only applied five times for RRSL (three), RobotiXX, and EW-Glab during the entire physical competition. The organizers will consider separating the physical DynaBARN environment from the static obstacle courses in future challenges, i.e., not being able to finish the static courses won't affect being able to navigate the DynaBARN area.

D. One single path through the obstacle course for fairness

In the second physical obstacle course, the organizers arranged a dense obstacle field in the middle of the traversal in a way that there were multiple paths to go through it, e.g., from left, middle, and right. The original intention of this arrangement was to encourage different navigation behaviors and test out the navigation systems' ability to make a decision when multiple topological path options exist. However, it turned out that different path options chosen by different teams would lead to different mapping results, affecting global planning later in the trials. To be specific, one path option allowed the robot to perceive certain obstacles and reveal a dead-end, while taking another option the robot did not see those critical obstacles and left the dead-end open. Towards later of the obstacle course, a narrow passage prompted the robot who took the second option to reconsider during global planning the earlier open part, which was actually a dead-end, getting the robot stuck when turning around to go back to the dead-end. Considering all path options were feasible, the team who chose the second option was penalized for reasons that were not related with poor navigation performance, which was not fair. Therefore, in future competitions, the organizers will try to maintain only one single possible path through the entire obstacle course to avoid complications caused by reasons not related to navigation and to assure fairness.

VI. FUTURE PLANS

Based on the first four year's BARN challenges, in the next BARN challenge in ICRA 2026, the organizers have

the following plans. First, the adoption of the no-tuning obstacle course one as well as the cold start bonus successful trials for obstacle courses two and three will be continued to incentivize navigation systems that do not require fine-tuning based on human expertise for every different deployment scenario. The allowed fine-tuning time (currently 20 minutes) will be reduced to discourage extensive dependence on manual trial and error. Second, the organizers will separate the dynamic and static portion of the obstacle course so failing in one won't affect being evaluated in another. More interactions with dynamic obstacles will thus become possible to extensively test existing navigation systems' dynamic obstacle avoidance capability. Third, it will be strictly ensured that there is only one possible path going through all obstacle courses, therefore preventing the mapping complications caused by taking different paths and perceiving different obstacles during the traversal. Lastly, the organizers will reach out to more potential competition sponsors to provide financial support for more teams to travel to the conference to participate the physical finals.

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] "Jackal ugv - small weatherproof robot - clearpath," <https://clearpathrobotics.com/jackal-small-unmanned-ground-vehicle/>, accessed: 2022-07-21.
- [5] 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.
- [6] 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.
- [7] 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.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] 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.

- [12] "ROS movebase navigation stack," http://wiki.ros.org/move_base, accessed: 2021-09-9.
- [13] "The barn challenge 2024," <https://cs.gmu.edu/~xiao/Research/BARN.Challenge/BARN.Challenge24.html>, accessed: 2024-06-02.
- [14] "The barn challenge 2025," <https://cs.gmu.edu/~xiao/Research/BARN.Challenge/BARN.Challenge25.html>, accessed: 2025-06-09.
- [15] A. Mazen, M. Faied, M. Krishnan, K. Yazdipaz, and I. Mateyaunga, "A two-stage tuning framework for fuzzy navigation: From intuitive fisvfh to robust S³-FISVFH," *IEEE Transactions on Fuzzy Systems*, 2025, [under review].
- [16] K. Balan, M. P. Manuel, M. Faied, M. Krishnan, and M. Santora, "A fuzzy based accessibility model for disaster environment," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 2304–2310.
- [17] J. Borenstein, Y. Koren *et al.*, "The vector field histogram-fast obstacle avoidance for mobile robots," *IEEE transactions on robotics and automation*, vol. 7, no. 3, pp. 278–288, 1991.
- [18] A. Mazen, M. Faied, and M. Krishnan, "Tuning of robot navigation performance using factorial design," *Journal of Intelligent & Robotic Systems*, vol. 105, no. 3, p. 50, 2022.
- [19] A. Mazen, M. Faied, and K. Mohan, "Optimal kinodynamic trajectory planner for mobile robots in an unknown environment using bézier contours," *IEEE Access*, 2024.
- [20] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.
- [21] Y. Lu, T. Xu, L. Wang, N. Hawes, and X. Xiao, "Decremental dynamics planning for robot navigation," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.
- [22] N. Mohammad, J. Higgins, and N. Bezzo, "A gp-based robust motion planning framework for agile autonomous robot navigation and recovery in unknown environments," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 2418–2424.
- [23] N. Mohammad and N. Bezzo, "Soft actor-critic-based control barrier adaptation for robust autonomous navigation in unknown environments," 2025. [Online]. Available: <https://arxiv.org/abs/2503.08479>
- [24] J. Tordesillas and J. P. How, "FASTER: Fast and safe trajectory planner for navigation in unknown environments," *IEEE Transactions on Robotics*, 2021.
- [25] D. Harabor and A. Grastien, "Online graph pruning for pathfinding on grid maps," in *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, ser. AAAI'11. AAAI Press, 2011, p. 1114–1119.
- [26] Q. Wang, Z. Wang, M. Wang, J. Ji, Z. Han, T. Wu, R. Jin, Y. Gao, C. Xu, and F. Gao, "Fast iterative region inflation for computing large 2-d/3-d convex regions of obstacle-free space," *IEEE Transactions on Robotics*, vol. 41, pp. 3223–3243, 2025.
- [27] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European Control Conference (ECC)*, 2019, pp. 3420–3431.
- [28] M. S. Darms, P. E. Rybski, C. Baker, and C. Urmson, "Obstacle detection and tracking for the urban challenge," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 3, pp. 475–485, 2009.
- [29] 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.
- [30] 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.
- [31] 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.
- [32] 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.
- [33] 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.
- [34] X. Xiao, B. Liu, G. Warnell, and P. Stone, "Motion planning and control for mobile robot navigation using machine learning: a survey," *Autonomous Robots*, pp. 1–29, 2022.
- [35] 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.
- [36] 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.
- [37] 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.
- [38] 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.
- [39] 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.
- [40] 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.
- [41] 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.
- [42] 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.
- [43] 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.
- [44] 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.
- [45] 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.
- [46] X. Xiao, B. Liu, G. Warnell, and P. Stone, "Toward agile maneuvers in highly constrained spaces: Learning from hallucination," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1503–1510, 2021.
- [47] X. Xiao, B. Liu, and P. Stone, "Agile robot navigation through hallucinated learning and sober deployment," in *2021 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2021, pp. 7316–7322.
- [48] B. Liu, X. Xiao, and P. Stone, "A lifelong learning approach to mobile robot navigation," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1090–1096, 2021.
- [49] S. A. Ghani, Z. Wang, P. Stone, and X. Xiao, "Dyna-lflh: Learning agile navigation in dynamic environments from learned hallucination," in *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.
- [50] D. Song, J. Liang, X. Xiao, and D. Manocha, "VI-tgs: Trajectory generation and selection using vision language models in mapless outdoor environments," *IEEE Robotics and Automation Letters*, 2025.
- [51] J. Liang, K. Weerakoon, D. Song, S. Kirubakaran, 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.
- [52] 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.
- [53] 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 Inter-*

- national Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2025.
- [54] 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.
 - [55] 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.
 - [56] 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.
 - [57] 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.
 - [58] 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.
 - [59] 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.
 - [60] 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.
 - [61] 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.
 - [62] 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.
 - [63] 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.
 - [64] 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.
 - [65] J. Liang, P. Gao, X. Xiao, A. J. Sathiamoorthy, 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.
 - [66] 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.
 - [67] 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.
 - [68] 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.
 - [69] 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.
 - [70] 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.
 - [71] 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.
 - [72] 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.
 - [73] 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.
 - [74] 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.
 - [75] 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.
 - [76] J. Hart, R. Mirsky, X. Xiao, S. Tejeda, B. Mahajan, J. Goo, K. Baldauf, S. Owen, and P. Stone, "Using human-inspired signals to disambiguate navigational intentions," in *International Conference on Social Robotics*. Springer, 2020, pp. 320–331.
 - [77] 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.
 - [78] 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.
 - [79] 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.
 - [80] 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.
 - [81] 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.
 - [82] 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.
 - [83] Y. Lu, D. Das, E. Plaku, and X. Xiao, "Multi-goal motion memory," *arXiv preprint arXiv:2407.11399*, 2024.
 - [84] 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.