

Autonomous Ground Navigation in Highly Constrained Spaces: Lessons Learned from The Third BARN Challenge at ICRA 2024

Competition Organizers: Xuesu Xiao¹, Zifan Xu², Aniket Datar¹, Garrett Warnell^{2,3}, and Peter Stone^{2,4},
Winning Teams: Joshua Julian Damanik⁵, Jaewon Jung⁵, Chala Adane Deresa⁵,
Than Duc Huy⁶, Chen Jinyu⁶, Chen Yichen⁶, Joshua Adrian Cahyono⁶,
Jingda Wu⁷, Longfei Mo⁷, Mingyang Lv⁷, Bowen Lan⁷, Qingyang Meng⁷, Weizhi Tao⁷, and Li Cheng⁷

Abstract—The 3rd BARN (Benchmark Autonomous Robot Navigation) Challenge took place at the 2024 IEEE International Conference on Robotics and Automation (ICRA 2024) in Yokohama, Japan and continued to evaluate the performance of state-of-the-art autonomous ground navigation systems in highly constrained environments. Similar to the trend in The 1st and 2nd BARN Challenge at ICRA 2022 and 2023 in Philadelphia (North America) and London (Europe), The 3rd BARN Challenge in Yokohama (Asia) became more regional, i.e., mostly Asian teams participated. The size of the competition has slightly shrunk (six simulation teams, four of which were invited to the physical competition). The competition results, compared to last two years, suggest that the field has adopted new machine learning approaches while at the same time slightly converged to a few common practices. However, the regional nature of the physical participants suggests a challenge to promote wider participation all over the world and provide more resources to travel to the venue. 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 3RD BARN CHALLENGE OVERVIEW

The 3rd BARN (Benchmark Autonomous Robot Navigation) Challenge [1] took place as a conference competition at the 2024 IEEE International Conference on Robotics and Automation (ICRA 2024) in Yokohama, Japan. As a continuation of The 1st and 2nd BARN Challenge at ICRA 2022 and 2023 in Philadelphia and London respectively, the 3rd 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 the first two competitions suggested, a problem far away from being solved [2], [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 and to a given goal without any collisions 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 3rd 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 2024 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 four teams from the simulation phase were then invited to compete in three different physical obstacle courses set up by the organizers at ICRA 2024 in the PACIFICO Yokohama conference center.

In this article, we report on the simulation qualifier and physical finals of The 3rd BARN Challenge at ICRA 2024, present the approaches used by the top three teams, discuss lessons learned from the challenge compared against The 1st and 2nd BARN Challenge at ICRA 2022 and 2023, 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 3rd BARN Challenge started on January 1st, 2024. 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

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

¹George Mason University ²The University of Texas at Austin ³Army Research Laboratory ⁴Sony AI ⁵Korea Advanced Institute of Science & Technology (KAIST) ⁶Nanyang Technological University (NTU) ⁷The Hong Kong Polytechnic University (PolyU)

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

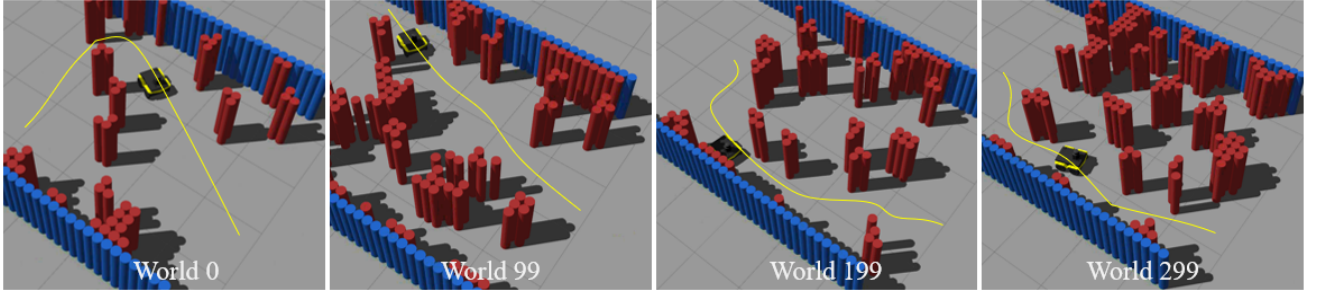


Fig. 1: Four example BARN environments in the Gazebo simulator (ordered by ascending relative difficulty level).

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 two competitions were also open sourced [1]. 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

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

where the indicator function 1_{success} evaluates to 1 if the robot reaches the navigation goal without any collisions, 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):

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

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 last two challenges, considering the performance upper bound, 0.25, has been closely

approached by multiple teams. In The 3rd 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 [1].

B. Results

The simulation qualifier started on January 1st, 2024 and lasted through a soft submission deadline (April 1st, 2024) and a hard submission deadline (May 1st, 2024). 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 Yokohama. 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, six teams, five 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 Tab. I with the baselines shown in the fourth column as a reference.

TABLE I: Simulation Results.

Rank.	Team	Score	Baseline
1	LiCS-KI	0.4762	
2	AIMS	0.4723	LfLH [10], e2e [9]
3	EIT-NUS	0.3795	APPLR-DWA [11], E-Band [8]
4	MLDA_EEE	0.2476	(Fast & Default) DWA [7]
5	Tartu Team	NA	
6	CCWSS	NA	

The top two simulation teams, LiCS-KI from Korea Advanced Institute of Science and Technology (KAIST) and AIMS from The Hong Kong Polytechnic University (HKPU) outperformed all last year’s winning teams (KUL+FM, IN-VENTEC, and University of Almeria). However, there is still a gap between the new performance upper bound (0.5) and the top performance (0.4762). The top four teams, LiCS-KI, AIMS, EIT-NUS from Eastern Institute of Technology, Ningbo, China, and MLDA_EEE from Nanyang Technological University (NTU) were invited to the physical finals

²https://github.com/Daffan/ros_jackal

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

at ICRA 2024. The top simulation (and also final winning) team, LiCS-KI, was the only team that submitted after the soft deadline but before the hard deadline.

III. PHYSICAL FINALS

The physical finals took place at ICRA 2024 in the PACIFICO Yokohama conference center on May 15th and May 16th, 2024 (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 3rd BARN Challenge in Yokohama, Japan.

A. Rules

Physical obstacle courses were set up using 120 cardboard boxes in the conference center. The organizers used the same guidelines to set up three obstacle courses as in the first two 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. Considering that KUL+FM finished all three physical obstacle courses in The 2nd BARN Challenge, the organizers intentionally increased the difficulty this year, i.e., introducing sharper turns and smaller clearances.

The organizers also used the same competition rules agreed upon by all the physical competition participants: Each team has 20 minutes to set up their navigation system after each obstacle course was constructed. After the 20-minute set-up time, each team had the opportunity to run five timed trials (after notifying the organizers to be timed) within another 20-minute period. The fastest three out of the five timed trials were counted, and the team that had the most

successful trials (reaching the goal without any collision) would be the winner. In the case of a tie, the team with the fastest average traversal time would be declared the winner.

B. Results

The four teams’ navigation performance is shown in Tab. II. Due to the intentionally increased navigation difficulty, the teams struggled more on obstacle avoidance, similar to The 1st BARN Challenge, and focused less on increasing speed, as the teams did during The 2nd BARN Challenge. The detailed results of all five timed trials (in seconds, only the top three were counted in the final score) are listed in the last three columns of Tab. II, where “X” indicates failure.

The winner, LiCS-KI, successfully and quickly finished all ten trials in the first two courses, but failed all five trials in the 3rd course, the extremely difficult one. MLDA_EEE also completely failed in the last, most difficult course, but succeeded in three and two trials in the first two courses. AIMS was able to slowly but successfully finish three trials in the last course, but did not perform well in the first two, especially the 2nd course, possibly due to a bug caused by sensor dimension mismatch. As a result, LiCS-KI won the competition by the most successful trials (6/9), while the tie between MLDA_EEE and AIMS was broken by the average traversal time (79s vs. 109s).

IV. TOP THREE TEAMS AND APPROACHES

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

A. LiCS-KI (KAIST)

The LiCS-KI team from KAIST introduced an end-to-end local navigation method for indoor navigation and deployed their Learned-imitation on Cluttered Space (LiCS) framework [13]. The main innovation is the use of a transformer-based network trained using Behavior Cloning (BC) with robust expert demonstrations under controlled noise. This method enables the robot to navigate robustly and rapidly through highly cluttered spaces. Additionally, a safety check layer is added to ensure safe navigation in untrained environments, particularly during real-world challenges.

1) **Neural network:** The neural network used by LiCS consists of a Transformer encoder and decoder, as depicted in Fig. 3. The encoder employs a Vision Transformer (ViT) [14] model with class token omitted, while the decoder is a standard transformer decoder without positional embedding and masked multi-head attention. During the encoding process, the LiDAR input is segmented into N patches, projected through a linear network, and concatenated with learnable position embedding. The decoder processes the encoded LiDAR data alongside the normalized local goal, provided by the global planner, to predict the optimal linear and angular velocities (v and ω).

TABLE II: Physical Results.

Rank.	Team	Success/Total	Average Time	Course 1	Course 2	Course 3
1	LiCS-KI	6/9	30/35/NA	32/31/32/27/30	37/37/40/29/32	X/X/X/X/X
2	MLDA_EEE	5/9 (79)	72/89/NA	68/X/77/X/70	X/X/X/85/93	X/X/X/X/X
3	AIMS	5/9 (109)	90/NA/121	92/88/X/X/X	X/X/X/X/X	119/118/126/X/X
4	EIT-NUS	0/9	NA/NA/NA	X/X/X/X/X	X/X/X/X/X	X/X/X/X/X

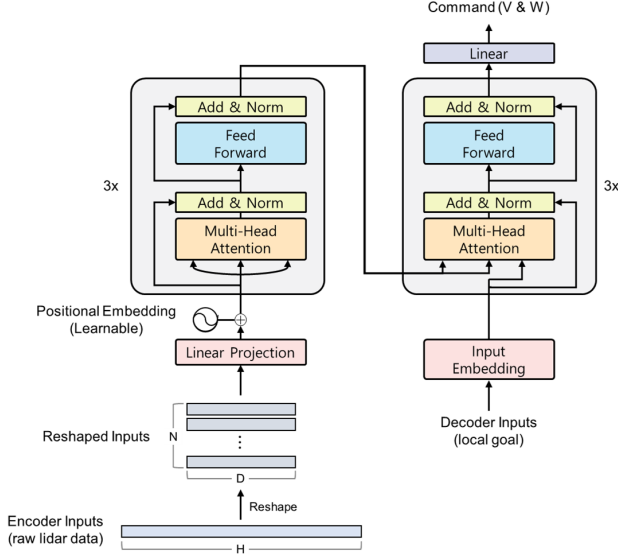


Fig. 3: Transformer-based neural network used in LiCS [13].

2) **Behavior Cloning (BC):** The proposed network is trained using BC to replicate the expert algorithm from the previous year’s winning team, KUL+FM. To address the inherent performance issues possessed by BC [15], a Gaussian noise $\mathcal{N}(0, \sigma^2)$ is injected to the control inputs during expert demonstrations. This noise augmentation allows the demonstrations to cover a variety of states for training, enhancing the policy network’s robustness [16]–[18].

3) **Safety check layer:** The safety check layer uses geometric calculations based on combined LiDAR and costmap data to enhance model safety. For linear motion ($|v| > 0$, $\omega = 0$), the robot travels along its x-axis. Safety is assessed by ensuring no obstacles are within a predefined rectangular safety zone extending from the robot’s front (Fig. 4a). In radial motion scenarios ($|v| > 0$, $|\omega| > 0$), where the robot follows a circular trajectory, the safety check involves ensuring no obstacles are present within two polygons that represent the robot’s footprint at the start and end of a movement interval, connected by arcs defining the robot’s outer and inner turning radii (Fig. 4b). Imminent collisions detected from this safety check layer trigger recovery actions, including speed reduction, in-place rotation, and backward movement.

4) **Implementation:** The training dataset was collected by recording simulations of the KUL+FM approach with injected Gaussian noise ($\sigma = 0.25$) across various scenarios. The network, consisting of three layers each in the ViT

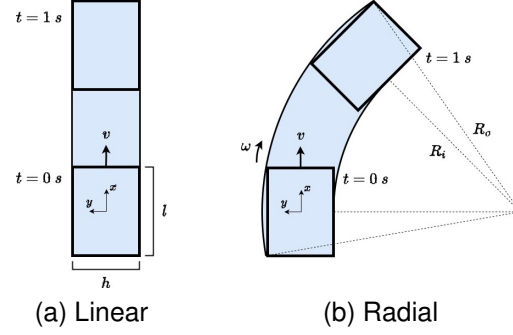


Fig. 4: Safety zone illustration for the safety check layer of LiCS during linear and radial movement.

encoder and transformer decoder, was trained in a supervised manner using the combined dataset over 100 epochs. In both simulated and real-world challenges, an A* algorithm was used as the global planner with dynamic obstacle inflation parameters that are adjusted based on velocity, $r_{\text{infl}} = r_{\text{min}} + (v/v_{\text{max}}) \times (r_{\text{max}} - r_{\text{min}})$. The safety layer was implemented solely in the real-world challenge, as the obstacle courses differed from the simulation environments. `hector_mapping` SLAM was also used during the real-world challenge to improve localization accuracy.

B. MLDA_EEE (NTU)

Team MLDA_EEE tackled the challenge using a classical approach with Model Predictive Control (MPC) with mode switching logic for different scenarios. The various modes all use MPC formulation with different initialization and constraints to bias the solver towards a feasible solution.

1) **Formulation:** The optimization variables of the MPC problem include the robot’s coordinates and headings as the state variables, $\mathbf{x} = [x, y, \theta]$, and the velocity and acceleration of both left and right wheels as the control, $\mathbf{u} = [v_r, v_l, a_r, a_l]$. The MPC minimizes the objective function J over the horizon of N steps, subject to the constraints of wheeled differential-drive model, the current odometry readings, and other additional constraints, to make the robot follow a reference trajectory:

$$\arg \min_{\mathbf{x}, \mathbf{u}} \sum_{k=0}^{N-1} J(\mathbf{x}, \mathbf{u}).$$

The objective function includes (1) error to the reference trajectory taken from the global planner, (2) error reference velocity, and (3) acceleration:

$$J = w_v (|v_k| - v_{ref})^2$$

$$+w_x[(x_k - x_{ref_k})^2 + (y_k - y_{ref_k})^2] \\ +w_a(a_k - a_{k+1})^2.$$

2) **Implementation:** The global trajectory is given by the global planner from ROS `move_base` package. The `map_server` is updated using `hector_mapping` to increase the reliability of the costmap. To reduce the computation time in the MPC, we minimized the number of obstacles considered in the optimization process. A ROS node is used to sample the raw LiDAR scan every 15 points. The `local_costmap` occupancy grid is used to obtain obstacles in the blind spot of the LiDAR, similar to INVENTEC [3]. These are published as point clouds with (x, y) coordinates used in the MPC as shown in Fig. 5. The local plan is obtained from optimizing the MPC problem using the non-linear solver CasADi [19].



Fig. 5: Rviz visualization of obstacles in the MPC. White squares: obstacle coordinates sampled from raw LiDAR scan. Yellow squares: blind spot obstacle coordinates obtained from `local_costmap`.

3) **Behavior of Different Modes:** Different modes have different MPC parameters such as the weights of the objective terms, control limits, and additional constraints on the reference global trajectory to allow safe maneuver near obstacles. These parameters are also fine-tuned in the physical runs.

The various modes include (1) “Safe”: the robot has high velocity, (2) “Obstacle Present”: obstacle is detected 1m away from the robot, and (3) “Close Obstacle”: obstacle is detected 0.5m away. Within the “Close Obstacle” mode, there is a “Reversing” mode which is triggered when the heading along the reference trajectory is more than 90° from the current heading (Fig. 6 top). This happens when the robot encounters a dead-end and the global plan suggests a new trajectory. In this mode, the optimization variables are initialized such that the heading points away from the goal, priming the optimal solution to result in the robot backtracking, instead of performing a sudden rotation, to prevent collision with nearby obstacles. When the robot backtracks to a safe space, the “Obstacle Present” parameters

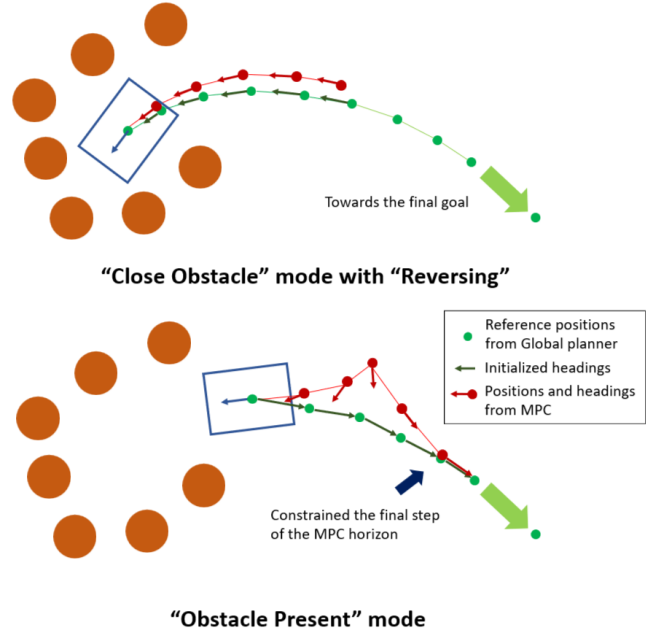


Fig. 6: Different modes with different initialization and limits. “Close obstacle” with “reversing” mode has the headings pointing away from the goal to allow safe backtracking (top); “Obstacle present” mode has normal heading initialization and constraints allowing the robot to rotate to the correct heading (bottom).

and the constraints on the final MPC horizon step allow the robot to regain the correct heading towards the final goal (Fig. 6 bottom).

C. AIMS (PolyU)

To effectively address the highly constrained environments in The BARN Challenge, the AIMS team developed a local planner utilizing advanced dynamic-window-based methods. This approach ensures collision-free navigation in narrow pathways by discretizing the sampling space into geometric elements for rapid determination in sensor space. It also incorporates backward sampling to assist the vehicle in adjusting its pose and extricating itself from tight situations. Given the competition rules prohibiting pre-mapping, the strategy focuses on real-time path planning rather than relying on real-time mapping and localization. This means the vehicle must continuously explore unknown areas while in motion, with the global plan constantly adjusted as exploration progresses. The algorithm is designed to continuously adapt to environmental changes and respond quickly to maintain collision-free navigation. The local planner operates without the need for global environmental data, allowing the vehicle to navigate challenging courses safely and efficiently, even without comprehensive maps and detailed localization information.

1) **Efficient Geometry-Based Obstacle Detection:** The method involves sampling potential trajectories with varying curvatures and applying geometric constraints to rapidly

identify potential collision points among these predicted paths. This approach facilitates the selection of the optimal collision-free trajectory. Drawing inspiration from last year's winning team (KUL+FM), the local planner is integrated directly with the sensor data, thereby bypassing potential inaccuracies in the costmap and accelerating obstacle detection.

Specifically, the anticipated driving area is discretized into rectangles and triangles, arranged by proximity. By scanning these shapes for LiDAR-detected points to identify obstacles, it quickly determines whether the sampled areas are collision-free. Before this scanning process, a crucial step involves filtering the LiDAR points within each geometric shape based on their distance and angle relative to the sensor, significantly reducing the search space required for each geometric assessment. By leveraging direct sensor integration and geometric analysis, this enhanced method ensures real-time adjustments and precise obstacle detection, making the navigation solution robust against highly constrained environments.

2) **Additional Backwards Sampling:** A further enhancement to the traditional sampling-based method is the implementation of sampling during both forward and backward driving. This backward sampling design assists the vehicle in effectively adjusting its pose to extricate itself when direct forward movement is not possible.

To be specific, both forward and backward driving involve sampling the potential trajectory curvatures of the vehicle. The difference lies in the judgment logic for backward sampling, which shifts from selecting samples that are closer to the local goal to selecting states that have more viable forward sampling points. This means that for each backward pose sampled, a second round of forward sampling is performed to find out those poses that have more feasible forward paths. A backward sampling result with more feasible forward paths typically indicates a superior vehicle pose in highly constrained environments, enabling better handling of complex situations when the vehicle reaches such a pose.

V. DISCUSSION

We discuss new findings and lessons from The 3rd BARN Challenge, not only from the technical perspective, but also from the competition organization side.

A. First physical competition win of end-to-end learning

In The 1st and 2nd BARN Challenges, the winning teams of the physical competition used classical approaches (UT AMRL and KUL+FM). However, the winning team this year, LiCS-KI, adopted an end-to-end imitation learning approach, which is the first physical win by end-to-end learning [20]. One interesting fact is that the expert used to provide demonstration data is a classical approach used by last year's winning team, KUL+FM, and LiCS-KI also added Gaussian noise to perturb the model input in order to achieve robustness, a classic data augmentation technique. Assisted by a Transformer architecture and safety check layer, LiCS-KI's

approach outperformed its expert demonstrator, KUL+FM, in the simulation qualifier. It is worth noting that KUL+FM did not participate in the physical competition this year, so it is unclear whether the imitator can outperform the demonstrator in the physical runs.

B. First usage of Transformers in the challenge

LiCS-KI is the first team in The BARN Challenge which used a Transformer architecture as the main local planner, compared to classical neural architectures used in the past years. The power of Transformers is one potential reason of LiCS-KI's win in both the simulation qualifier and physical finals, along with the data augmentation technique and safety check layer. The success of the Transformer architecture suggests the potential of better neural architecture for robot navigation tasks, not only to address visual inputs [21]–[24], off-road conditions [25]–[32], social contexts [33]–[43], kinodynamic constraints [44], or multi-robot navigation [45]–[47], but also for purely geometric obstacle avoidance [10], [48]–[51]. The revolutionary success of Transformers on computer vision and natural language processing tasks may also inspire future navigation research.

C. Successful sim-to-real transfer of learning algorithms

Similar to The 2nd BARN Challenge [3], the 3rd year of the competition did not exhibit a significant sim-to-real gap. The first place winner of both simulation and physical course challenges, the LiCS-KI team, utilized a learning-based algorithm. Not only winning in terms of success rate during the physical finals, the team performed with the fastest average traversal time. This result highlights a small sim-to-real performance gap. It also suggests that learning-based models, particularly those trained in simulated environments, are becoming increasingly effective at handling the unpredictable nature of real-world settings when combined with the strategic use of imitation learning, specifically through behavior cloning, coupled with advanced data augmentation and neural architecture. This approach also contrasts with the more commonly used reinforcement learning [22], [23], [52]–[54] in past competitions. By employing imitation learning [26], [29]–[31], [55], [56], the team was able to quickly deploy behaviors mimicking or even surpassing expert demonstrations, reducing the need for the trial-and-error learning phases typical of reinforcement learning. Additionally, the use of controlled noise during training helped the algorithm account for unforeseen variables and disturbances encountered.

D. Strong connections to previous years

While last year's 2nd place winner, INVENTEC, based their approach on the strongest baseline, LfLH [10] along the Learning from Hallucination (LfH) [10], [48], [49], [51] line of work, the approaches developed by this year's teams started to show strong connections to previous years' methods. LiCS-KI used the approach by last year's KUL+FM as expert to generate demonstration data, while AIMS also leveraged KUL+FM's idea of local planning directly in

the sensor space, instead of using costmaps, which are susceptible to inaccuracies. Along with the point in the following subsection, the community has started to form a few common practices to address the problem of navigation in highly constrained spaces, which also have real-world implications when deploying autonomous mobile robots in natural obstacle-occupied spaces.

E. Importance of a hybrid paradigm

All teams adopted a hybrid paradigm in terms of a finite-state-machine setup, which requires different components to address different situations in the obstacle courses, especially safety checking of the actions produced by a main planner, differently parameterized MPC planners, and specifically designed reversing motions to back up the robot from undesirable scenarios. Such a pragmatic practice suggests that a single stand-alone approach that is able to address all variety of obstacle configurations all together is still out of our reach. Even for the end-to-end learning by LiCS-KI, a separate safety check layer is still required during hardware implementation. Most teams also specifically designed reversing or backtracking behaviors to address situations where the robot got stuck. However, more complex systems may introduce extra complications at the same time, e.g., proper parameter tuning for each component and appropriate transition conditions between different components.

F. Tie-Breaking by Average Time for 2nd and 3rd Place

Qualitatively speaking, this year's physical obstacle courses were slightly more difficult than last year's. Unfortunately, no team could finish all nine physical trials. LiCS-KI outperformed MLDA_EEE and AIMS by one more successful trial, while MLDA_EEE and AIMS were tied in terms of success rate. The tie was broken by average time. However, it is worth noting that due to the different length and difficulty of the three physical obstacle courses, it is difficult to make an absolutely fair comparison using average time of successful trials to break the tie: AIMS succeeded in three trials in the longest 3rd obstacle course, while didn't finish one single trial in the shorter 2nd obstacle course. MLDA_EEE's performance was the opposite, which presents an advantage. Such a situation increases AIMS's average traversal time compared to MLDA_EEE, causing the rank of 2nd and 3rd place.

G. More financial support for participation is needed

One unfortunate fact about The 3rd BARN Challenge is that all four teams that participated in the physical finals are Asian teams. Considering that ICRA 2024 took place in Yokohama, Japan, not many teams from places far away from Japan submitted their navigation stack to participate in the competition. The regional participation of the competition is not ideal to evaluate the entire field's progress and compare the performance of top teams all over the world. The organizers will try to reach out to more potential sponsors to provide more financial support to invite participants to travel from other continents. Another potential

solution is to provide remote participation options, which was attempted last year. However, the need of fine-tuning the navigation systems for real-world deployment and to fit to every different obstacle course makes it impractical for the organizers to run the remote participants' systems and achieve reasonable performance out-of-the-box. How to remove the reliance on extensive system tuning is still an open question for robust obstacle avoidance in a variety of real-world scenarios [11], [57]–[61].

VI. FUTURE PLANS

Based on the first three year's BARN challenges, the organizers plan to make the following changes in the next BARN challenge in ICRA 2025. First, dynamic obstacles will be introduced to the currently static obstacle courses [6], [51]. For the first competition with dynamic obstacles, the organizers will allow collisions with dynamic obstacles and only add a penalty, whereas collisions with static obstacles will still be counted as a total failure. The addition of dynamic obstacles will stress-test the robustness of obstacle avoidance and also make the competition more interesting to watch. Second, to further encourage the teams to reduce the need of on-site fine-tuning, the organizers also plan to add a few "cold trials" at the beginning of each obstacle course: All teams will be required to directly navigate through each obstacle course without any fine-tuning of the system first. Successful cold trials will be rewarded by bonus points, before the teams are allowed to fine-tune their systems and start their regular trials. The organizers also plan on reducing the allowed time to fine-tune the system to discourage extensive dependence on manual trial and error before autonomous navigation.

REFERENCES

- [1] "The barn challenge 2024," <https://cs.gmu.edu/~xiao/Research/BARN.Challenge/BARN.Challenge24.html>, accessed: 2024-06-02.
- [2] 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.
- [3] 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.
- [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] J. J. Damanik, J.-W. Jung, C. A. Deresa, and H.-L. Choi, "Lics: Navigation using learned-imitation on cluttered space," 2024.
- [14] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," Jun. 2021, arXiv:2010.11929 [cs]. [Online]. Available: <http://arxiv.org/abs/2010.11929>
- [15] S. Ross and J. A. Bagnell, "Efficient Reductions for Imitation Learning," 2010.
- [16] M. Green and J. B. Moore, "Persistence of excitation in linear systems," *Systems & Control Letters*, vol. 7, no. 5, pp. 351–360, Sep. 1986. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/0167691186900526>
- [17] M. Laskey, J. Lee, R. Fox, A. Dragan, and K. Goldberg, "DART: Noise Injection for Robust Imitation Learning," 2017.
- [18] L. Ke, J. Wang, T. Bhattacharjee, B. Boots, and S. Srinivasa, "Grasping with Chopsticks: Combating Covariate Shift in Model-free Imitation Learning for Fine Manipulation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. Xi'an, China: IEEE, May 2021, pp. 6185–6191. [Online]. Available: <https://ieeexplore.ieee.org/document/9561662/>
- [19] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.
- [20] 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.
- [21] 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.
- [22] 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.
- [23] 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.
- [24] 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.
- [25] 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.
- [26] A. Datar, C. Pan, and X. Xiao, "Learning to model and plan for wheeled mobility on vertically challenging terrain," *arXiv preprint arXiv:2306.11611*, 2023.
- [27] 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.
- [28] 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.
- [29] 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.
- [30] 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.
- [31] 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.
- [32] A. Pokhrel, A. Datar, M. Nazeri, and X. Xiao, "Cahsor: Competence-aware high-speed off-road ground navigation in se (3)," *arXiv preprint arXiv:2402.07065*, 2024.
- [33] 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.
- [34] D. Song, J. Liang, A. Payandeh, X. Xiao, and D. Manocha, "Socially aware robot navigation through scoring using vision-language models," *arXiv preprint arXiv:2404.00210*, 2024.
- [35] 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.
- [36] 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.
- [37] 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.
- [38] 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.
- [39] 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.
- [40] 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.
- [41] 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. Sindhvani, "Learning model predictive controllers with real-time attention for real-world navigation," in *Conference on robot learning*. PMLR, 2022.
- [42] 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.
- [43] 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.
- [44] 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.
- [45] 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.
- [46] 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.
- [47] 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.
- [48] 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.

- [49] 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.
- [50] 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.
- [51] S. A. Ghani, Z. Wang, P. Stone, and X. Xiao, "Dyna-lflh: Learning agile navigation in dynamic environments from learned hallucination," *arXiv preprint arXiv:2403.17231*, 2024.
- [52] 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.
- [53] 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.
- [54] 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.
- [55] 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.
- [56] 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.
- [57] 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.
- [58] 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.
- [59] 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.
- [60] 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.
- [61] 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.