2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2025)
October 19-25, 2025. Hangzhou, China

2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 979-8-3315-4393-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/IROS60139.2025.11247669

T-CBF: Traversability-based Control Barrier Function to Navigate Vertically Challenging Terrain

Manas Gupta and Xuesu Xiao

***Abstract*— Safety has been of paramount importance in motion planning and control techniques and is an active area of research in the past few years. Most safety research for mobile robots target at maintaining safety with the notion of collision avoidance. However, safety goes beyond just avoiding collisions, especially when robots have to navigate unstructured, vertically challenging, off-road terrain, where vehicle rollover and immobilization is as critical as collisions. In this work, we introduce a novel Traversability-based Control Barrier Function (T-CBF), in which we use neural Control Barrier Functions (CBFs) to achieve safety beyond collision avoidance on unstructured vertically challenging terrain by reasoning about new safety aspects in terms of traversability. The neural T-CBF trained on safe and unsafe observations specific to traversability safety is then used to generate safe trajectories. Furthermore, we present experimental results in simulation and on a physical Verti-4 Wheeler (V4W) platform, demonstrating that T-CBF can provide traversability safety while reaching the goal position. T-CBF planner outperforms previously developed planners by 30% in terms of keeping the robot safe and mobile when navigating on real world vertically challenging terrain.**

I. Introduction

Field robots deployed in real-world applications often need to traverse unstructured, complex, and unpredictable off-road terrain, where safety becomes a top priority. The robotics community has developed numerous techniques to address motion planning safety: Traditional model-based control methods, including Hamilton-Jacobi Reachability Analysis, Control Barrier Functions, Control Lyapunov Functions, and Model Predictive Control, have proven effective but struggles with scalability and generalizability across diverse environments. To overcome these limitations, data-driven approaches [1], [2] are emerging, offering improved adaptability to new environments but at a price of lacking formal safety guarantees.

Despite such a plethora of techniques, most research on mobile robot safety is centered around collision avoidance, defining safety in motion planning as the ability to keep robots within obstacle-free regions. However, in field robotics, safety takes on a more complex meaning, as robots must operate in challenging terrain where the traditional notion of collision avoidance is no longer adequate. Fig. 1 illustrates a wheeled robot navigating unstructured terrain, highlighting scenarios where the robot becomes unsafe when it gets stuck or rolls over in different configurations on vertically challenging terrain [3]. These configurations can be avoided by motion planning algorithms that emphasize on safety beyond collision avoidance, e.g., traversability factors such as vehicle rollover and immobilization.

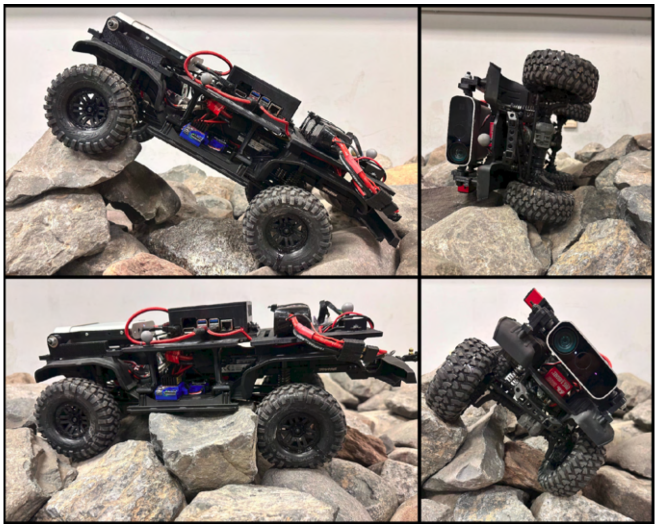

Fig. 1: Unsafe configurations of a wheeled robot navigating vertically challenging terrain. Such risks can be mitigated by a planning algorithm that emphasizes traversability safety beyond traditional collision avoidance.

Recent advancements in designing scalable and generalizable Control Barrier Functions (CBFs) for mobile robots show promises to facilitate traversability safety beyond collision avoidance. Leveraging machine learning to learn CBFs from data enhances adaptability to diverse environments. To improve generalizability, observation-based approaches have been introduced, offering greater scalability compared to traditional state-based CBFs. These approaches utilize perception inputs, such as LiDAR point clouds or camera images, to extract environmental obstacle information. Existing CBFs in these approaches learn a distance function from the observations to maintain safe distance from obstacles. However, extending the safety notion beyond obstacle avoidance to off-road traversability safety remains a challenge.

To address this challenge, this work proposes a Traversability-based Control Barrier Function (T-CBF) to enable traversability safety for navigation of wheeled robots on unstructured and vertically challenging terrain using onboard perception inputs. The main contributions are threefolds: (1) The introduction of a novel approach to construct a CBF that

This work has been taken place at RobotiXX Laboratory in Department of Computer Science, George Mason University {mgupta27, xiao}@gmu.edu. This research is supported by National Science Foundation (NSF, 2350352), Army Research Office (ARO, W911NF2320004, W911NF2420027, W911NF2520011), Air Force Research Laboratory (AFRL), US Air Forces Central (AFCENT), Google DeepMind (GDM), Clearpath Robotics, Raytheon Technologies (RTX), Tangenta, Mason Innovation Exchange (MIX), and Walmart.

979-8-3315-4393-8/25/$31.00 ©2025 IEEE

extends the concept of safety to traversability by incorporating perception-based information. (2) Demonstrating the generalizability of T-CBF to previously unseen unstructured environments. (3) A statistical evaluation and experimental verification of T-CBF on an autonomous Verti-4 Wheeler (V4W) [4] platform in diverse environments. The results demonstrate that T-CBF providesmay prevent the theoretical guarantees outlined in Section V from fully applying in real-world scenarios. To support the practical effectiveness of our learned certificates, the following section presents experimental results demonstrating that the controller consistently operates safely and achieves its objectives across diverse simulated and physical environments. traversability safety beyond collision avoidance while navigating complex, vertically challenging, unstructured environments.

II. Related Work

This section discusses the related work on navigation in unstructured environments, the application of CBFs to ensure safety in navigation, and their extension to learning-based approaches.

A. Navigation in Unstructured Environments

Navigation for wheeled robots in unstructured off-road environments presents significant challenges. Existing work have investigated various challenges and planning algorithms for navigation in unstructured off-road environments [5]–[7]. Due to complexity of modeling system dynamics and terrain interactions, most approaches utilize machine learning to learn kinodynamic model [8]–[10] or reduce uncertainty in terrain interactions [11]. However, these methods do not explicitly ensure traversability safety which is crucial for navigation in unstructured terrain.

B. Control Barrier Functions

The classical formulation of CBFs provides safety guarantees in various robotics applications. Rabiee and Hoagg [12] integrated CBFs into the Model Predictive Path Integral (MPPI) algorithm to ensure safety of sampled trajectories. Similarly, Grandia et al. [13] applied CBFs within a Model Predictive Control framework to determine safe foot placements for legged robots navigating structured terrain such as stepping stones. For robotic manipulation, Dai et al. [14] employed CBFs for object avoidance using differentiable optimization solvers while maintaining safety guarantees. Furthermore, Patterson et al. [15] utilized CBFs to regulate self-contact in soft-rigid robots, demonstrating their versatility in maintaining safe interactions across different robotic domains. The classical approaches struggles to generalize to new environments and are often overly conservative in order to provide formal safety guarantees.

C. Learned Control Barrier Functions

With the growing adoption of data-driven methods, CBFs are becoming increasingly significant in robotics applications. In general, synthesizing a CBF requires a mathematical formulation that can either be learned by a neural network or directly incorporated as a constraint in an optimization framework. For approaches utilizing perception data, Long et al. [16] developed a CBF that is created online using range sensing for obstacle avoidance, and Dawson et al. [17] designed a hybrid controller incorporating a learned CBF to navigate using LiDAR point cloud data. Similarly, Abidi et al. [18] leveraged camera sensor images to construct a vision-based learned CBF for obstacle avoidance while maintaining a safe following distance. Additionally, Harms et al. [2] introduced a neural CBF-based safety filter for quadrotor navigation in unknown environments. These approaches demonstrate the effectiveness of machine learning in designing scalable CBFs that generalize to new environments. Reinforcement learning has also been used in conjunction with CBFs to develop safe navigation policies for obstacle avoidance [19].

However, all these approaches focus on collision avoidance as a safety concept, which is not sufficient for navigation on unstructured terrain. This research extends prior work on CBFs from collision avoidance to traversability safety.

III. Preliminaries

Consider a closed-loop non-linear control affine system,

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{u}), \\ o &= k(\mathbf{x}, m) \end{aligned} \tag{1}$$

where $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^n$ and $\mathbf{u} \in \mathcal{U} \subset \mathbb{R}^m$ are the robot state and control input respectively. $o \subset \mathbb{R}^{L*W}$ is the onboard observation, which is a 2.5D elevation map for vertically challenging terrain with L and W being length and width of the elevation patch. $k(\cdot, \cdot)$ maps the state $\mathbf{x}$ and environment variable m, which is the terrain underneath the robot, to the 2.5D elevation map observation o. $f(x, u)$ is locally Lipschitz continuous function.

Definition 1: A set $S \subset \mathbb{R}^n$ is called a forward invariant set for the system (1), if for any $\mathbf{x}(t_0) \in S$ there exists an input trajectory $\mathbf{u}_t \in \mathcal{U}$ such that $\mathbf{x}(t) \in S \ \forall t \geq t_0$ under system (1) [20].

Definition 2: (Control Barrier Function) The system (1) is considered safe if $\mathbf{x}(t) \in \mathcal{X}_s \subset \mathcal{X}$, $\mathbf{u}_t \in \mathcal{U}_s \subset \mathcal{U}$, $\forall t \geq 0$, where $\mathcal{X}_s$ and $\mathcal{U}_s$ are safe states and safe control inputs respectively.

Let $S \subset \mathcal{X}_s$ be a zero-superlevel set for a smooth continuous differentiable function $h : \mathbb{R}^n \to \mathbb{R}$ that satisfies the following conditions:

$$S = \{\mathbf{x} \in \mathcal{X} \,|\, h(\mathbf{x}) \geq 0\} \tag{2a}$$

$$\partial S = \{\mathbf{x} \in \mathcal{X} \,|\, h(\mathbf{x}) = 0\} \tag{2b}$$

$$Int(S) = \{\mathbf{x} \in \mathcal{X} \,|\, h(\mathbf{x}) > 0\} \tag{2c}$$

where ∂S is the boundary and $Int(S)$ is the interior of set S respectively. For forward invariant set S, the function h with property that $S = \{\mathbf{x} \in \mathcal{X} \,|\, h(\mathbf{x}) \geq 0 \}$ and $\{\mathbf{x} \in \mathcal{X} \,|\, \frac{dh}{d\mathbf{x}}(\mathbf{x}) = 0\} \cap \{\mathbf{x} \in \mathcal{X} \,|\, h(\mathbf{x}) = 0\} = \emptyset$ is a control barrier function if there exists an extended class $\mathcal{K}_\infty$ function $\alpha(\cdot)$ such that for the system (1) :

$$h(\mathbf{x}) > 0 \ \forall \mathbf{x} \in S, \tag{3a}$$

$$\sup_{\mathbf{u}\in\mathcal{U}}[\mathcal{L}_f h(\mathbf{x}) + \mathcal{L}_g h(\mathbf{x})\mathbf{u}] \geq -\alpha(h(\mathbf{x})) \tag{3b}$$

where $[\mathcal{L}_f h(\mathbf{x})+\mathcal{L}_g h(\mathbf{x})\mathbf{u}] = \dot{h}(x,u)$ and $\mathcal{L}_f$ and $\mathcal{L}_g$ are Lie derivatives. Hence, satisfying (3b) makes safe set S forward invariant and $\mathbf{u}$ a safe control.

IV. Methodology

Based on the preliminaries, we present T-CBF for safe navigation in vertically challenging terrain.

A. Traversability-based Control Barrier Function

Since obstacle avoidance alone is insufficient for safely traversing complex unstructured terrain, we introduce T-CBF, a neural CBF, that is trained on real world observation data and robot dynamic model. The neural network architecture of T-CBF is illustrated in Fig 2. The network takes a 2.5D elevation map patch of size 100 by 40 pixels as input, capturing the elevation information of the terrain. The elevation patch is passed through a series of 2D convolution layers followed by a series of feed-forward layers. The control actions are encoded using an encoder, which is then used in the loss function (described in IV-B), to train the model. Fig 2 illustrates the output of the T-CBF, which is the safe region (dotted red boundary) based on traversability safety.

Formulating safety constraints for high dimensional systems on unstructured terrain is complicated due factors such as suspension and tire deformation, varying tire-terrain friction, vehicle weight distribution and momentum, etc. To address this, we utilize manually driven runs to collect data and identify unsafe states (Fig. 1) of the robot based on the following criteria: Denote the robot state as $\mathbf{x}_t = (x_t, y_t, z_t, r_t, p_t, \phi_t)$, where the first three are translational (x, y, z) and last three are rotational (roll, pitch, yaw) components respectively along x, y, and z axis, and control input as $\mathbf{u}_t = (v_t, \omega_t)$, where v is translational velocity and ω is angular velocity which also correlates to steering curvature.

1) $p_t \geq p_{thresh}$: Pitch angles greater than a threshold value as large pitch angles lead to robot getting stuck;
2) $\phi_t \geq \phi_{thresh}$: Roll angles greater than a threshold as large roll angles increases the risk of vehicle rollover;
3) $(\Delta x, \Delta y, \Delta z) < \Delta_{thresh}$ and $(u > u_{thresh})$: Smaller robot position changes with greater velocity and steering commands, compared to their respective thresholds, indicates that the robot is immobilized, e.g., due to insufficient contact with the terrain or tire slipping.

The first two inequalities can be added as constraints in an optimization framework to maintain safety as those can be predicted by a learned kinodynamic model. However the third inequality is difficult to enforce as a constraint because of the many different factors, as described above, which are involved in robot-terrain interactions after the control is applied. In T-CBF we learn a control barrier function from the samples of safe and unsafe observations which can be applied as a constraint to a control optimization problem.

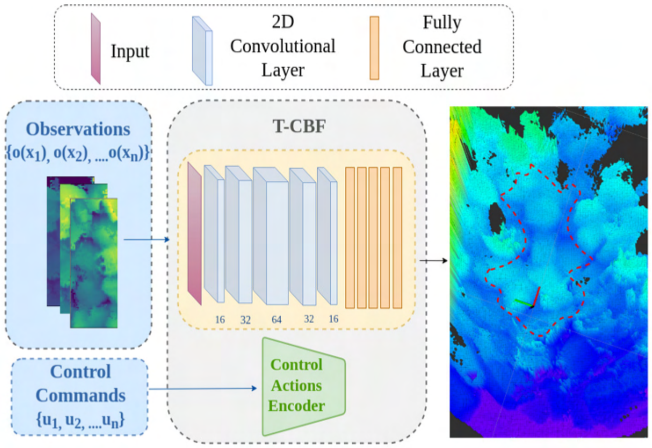

Fig. 2: T-CBF Neural Network Architecture: The observations (elevation maps) are processed through a neural network as CBF and control actions are processed through an encoder. The loss function helps learn a CBF enforcing satisfaction of Eqns. (4a), (4b), and (4c). The output, safe region (dotted red boundary), is obtained by sampling observations on elevation map from a fixed position with different control commands.

B. Learning T-CBF from Observations

Unlike traditional state-based formulations of CBFs, we define CBF as function of state $\mathbf{x}$ and observations $\mathbf{o} \in \mathcal{O} \subset \mathbb{R}^o$ [2] [17]. This enables the CBF to generalize into new environments where new observations can be used to create CBF defining safe regions around the robot. The T-CBF based on observations $h : \mathcal{O} \to \mathbb{R}$ is defined as:

$$h(o, x) \geq 0 \quad \forall o \in \mathcal{O}, \forall x \in \mathcal{X}_{\text{safe}}, \tag{4a}$$

$$h(o, x) < 0 \quad \forall o \in \mathcal{O}, \forall x \in \mathcal{X}_{\text{unsafe}}, \tag{4b}$$

$$h[o(f(x_t, u))] \geq -\alpha_h(h(o, x_t)), \ \forall u \in \mathcal{U}_{\text{safe}}, \forall x \in \mathcal{X}_{\text{safe}} \tag{4c}$$

where $0 \leq \alpha_h < 1$ is an extended class $\mathcal{K}_\infty$ function. We formulate our CBF similar to Harms et al. [2] and use the switching set approach to avoid the observation dynamics approximation that comes from differentiating $h(o, x)$ and results in $\frac{d}{dt}h(o, x) = h[o(f(x_t, u))]$ which represents the state dynamics of the system. The formulated CBF is a Discrete Time Control Barrier Funtion.

To train the network illustrated in Fig. 2, we use the following loss function using the elevation map, o_t and o_{t+1} at time t and $t+1$, as an observation input:

$$Loss = c_1 \sum_{(o_t)^i \in \mathcal{O}_{\text{unsafe}}} ReLU(h(o_t)^i + \epsilon_1) + \tag{5a}$$

$$c_2 \sum_{(o_t)^i \in \mathcal{O}_{\text{safe}}} ReLU(\epsilon_2 - h(o_t)^i) + \tag{5b}$$

$$c_3 \sum_{(o_t, o_{t+1})^i \in \mathcal{O}_{\text{safe}}} ReLU(\epsilon_3 - (h(o_{t+1})^i u_e + \alpha(h(o_t)^i))). \tag{5c}$$

In the loss function, the terms (5a) and (5b) ensure satisfaction of safe and unsafe observation inequalities in (4a) and (4b) respectively, which is a standard practice in training a neural CBF. However, we add (5c) explicitly to enforce satisfaction of equation (4c). In unstructured environment, a terrain patch can be safe to traverse with a control command u, but unlike collision avoidance, the same patch can also become unsafe with another control command u', hence we explicitly incorporate encoded control command u_e in the equation (5c) of the loss function to formulate traversability safety. Thresholds $\epsilon_1, \epsilon_2, \epsilon_3 \geq 0$ are added to enforce strict satisfaction of all the inequalities and weights $c_1, c_2, c_3 \geq 0$ are hyper-parameters.

The learning process involves training the T-CBF network using a small data set of over 4,000 real-world observation samples, evenly distributed between safe and unsafe sets with 20% reserved for validation. We train the network for 150 epochs using the Adam Optimizer.

C. Planning Algorithm

We formalize the navigation planning problem as a control optimization framework:

$$u = \arg\min_{u} \quad \lambda_1 \|u\| + \lambda_2 \, \mathcal{C}_{\text{goal}} + \lambda_3 \, \mathcal{C}_{\text{stab}}, \tag{6a}$$

$$\text{s.t.} \quad u_{\min} \leq u \leq u_{\max}, \tag{6b}$$

$$h_{t+1} + \alpha(h_t) \geq 0. \tag{6c}$$

The resulting trajectory adheres to both the system constraint (6b) and the CBF constraint (6c) making it traversability safe. To encourage goal-directed behavior, we incorporate a cost term, $\mathcal{C}_{\text{goal}}$, which penalizes control actions leading to states that deviate from the goal position as shown in equation (7). x_g and y_g are x and y component of goal position and w_x and w_y are weights for x and y components respectively.

$$\mathcal{C}_{\text{goal}} = w_x |x_t - x_g| + w_y |y_t - y_g| \tag{7}$$

Additionally, we introduce a stability cost, $\mathcal{C}_{\text{stab}}$ shown in equation (8), to penalize control actions that result in states with excessive pitch p_t and roll r_t which are weighted by w_p and w_r respectively.

$$\mathcal{C}_{\text{stab}} = w_r |r_t| + w_p |p_t| \tag{8}$$

Given the challenges of mathematically formulating robot dynamics on unstructured terrain, we employ a learned forward kinodynamic model, TAL [21], to predict new states based on control inputs. $\lambda_1, \lambda_2, \lambda_3$ are weights with $\lambda_1 \gg \lambda_2 \gg \lambda_3$ to give more importance on minimizing the control action. The optimization is run in loop for the planning horizon of 10 steps which provides adequate foresight to ensure safe and stable motion towards the goal. The optimization terminates once the robot reaches within 0.1 m of the goal state.

V. Experiments

To validate that T-CBF provides traversiability safety on unstructured vertically challenging terrain and is able to generalize to new environment, we compare the performance with WM-VCT [4] and TAL [21], previously developed planners to solve the same problem. We conduct experiments both in simulation and on a physical robot in real world environment.

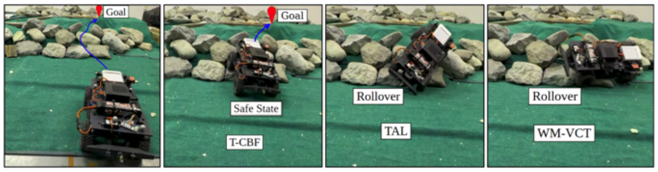

Fig. 3: T-CBF validation test (from left to right): V4W receives start and goal (red) states in safe region. We intentionally set a global path that goes through a non-traversable terrain patch (blue). T-CBF navigates the robot towards the goal state and pauses the robot in a safe state. Robot topples and immobilizes with WM-VCT and TAL when attempting to traverse unsafe terrain.

A. Robot and Testbed

The robot used for the physical experiments is a V4W platform [3], an open-source, 1/10th-scale unmanned ground vehicle. The robot features a low-high gear switch and lockable front and rear differentials, enhancing its mobility on vertically challenging terrain. For perception, we utilize the onboard Microsoft Azure Kinect RGB-D camera to generate elevation map and to run Visual Inertia Odometry [22]. Real-time elevation maps are generated using an open-source tool that processes the depth data and all onboard computation runs on an NVIDIA Jetson Orin NX.

To support experimentation, we construct a 3.1m × 5m rock testbed with a maximum height of 0.6m. For reference, the V4W is 0.2m high, 0.249m wide, 0.523m long, with a wheelbase of 0.312m. The testbed consists of reconfigurable rocks, enabling flexible data collection and mobility testing across diverse terrain configurations.

We categorize the testbed into three difficulty levels—low, medium, and high—based on elevation changes. In the low-difficulty setting, rocks are evenly arranged to minimize elevation variation, ensuring a continuous surface without gaps. The medium difficulty level introduces increased elevation changes along with small gaps, which may cause the robot to lose traction or become temporarily stuck. At the high-difficulty level, significant elevation variations are introduced, with rock placements designed to challenge the robot's stability, potentially causing rollovers or immobilization. Additionally, wooden slats and spray foam are incorporated to introduce further variations in terrain complexity.

B. T-CBF Validation

To evaluate the safety performance of T-CBF, we configure the testbed to simulate unsafe traversable scenarios. We conduct experiments using the WM-VCT and TAL algorithms, which lack explicit safety constraints, and compare their results with T-CBF. Throughout our experiments, we assume that the robot's start and goal positions are located in a safe region. Unlike the other planners, T-CBF successfully prevents the robot from entering in unsafe states and getting immobilized. The robot remains mobile and capable of searching alternate paths towards goal position. In contrast, WM-VCT and TAL become immobilized while attempting to navigate the unsafe terrain. The results are presented in Fig. 3. This experiment validates that T-CBF is able to

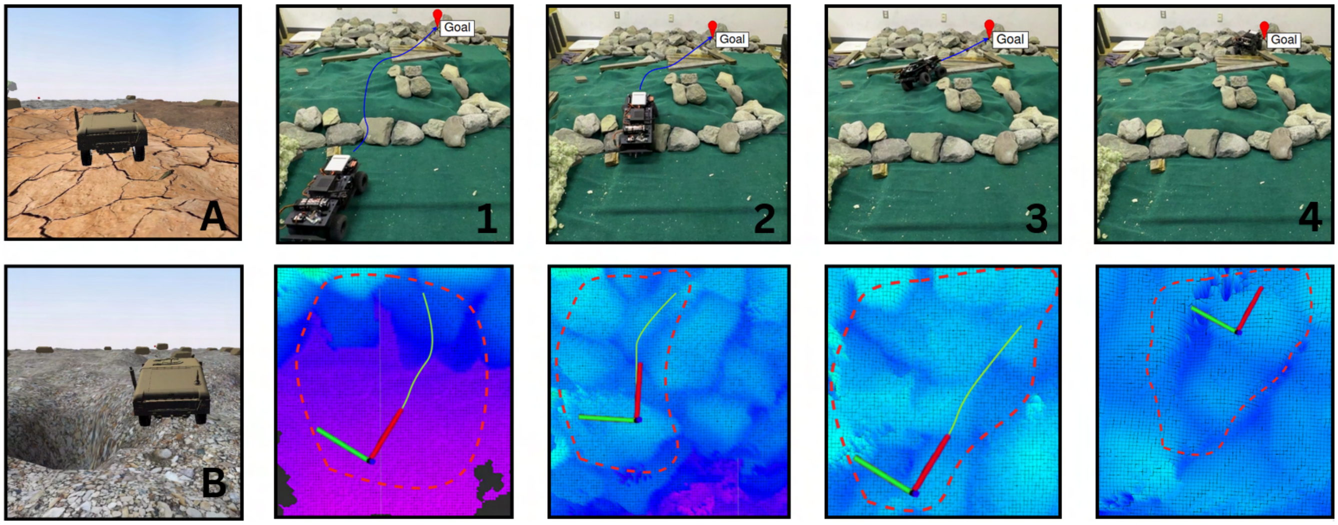

Fig. 4: Experimental evaluation of T-CBF. A & B: Simulation run of robot avoiding unsafe regions while navigating challengin terrain. 1: Robot initialization with start and goal pose in safe region. 2-4: Robot navigating unstructured vertically challenging terrain and avoiding nontraversable unsafe regions to reach the goal position with corresponding bottom images showing traversability-based safe regions (dotted red boundary) on elevation map computed by T-CBF and resulted trajectories in green.

Task	Planner	Success Rate ↑	Traversal Time (s) ↓	Mean Roll (rad) ↓	Mean Pitch (rad) ↓
Simulation	T-CBF	**60.7%**	26.80 ± 2.30	**0.100** ± 0.30	**0.073** ± 0.08
	TAL	58.2%	25.20 ± 2.75	0.105 ± 0.35	0.076 ± 0.07
	WM-VCT	40.12%	**24.37** ± 2.52	0.154 ± 0.38	0.089 ± 0.11
Real World - Low	T-CBF	**10/10**	17.92 ± 2.08	**0.131** ± 0.13	**0.059** ± 0.08
	TAL	10/10	**16.63** ± 2.27	0.136 ± 0.24	0.061 ± 0.07
	WM-VCT	9/10	17.10 ± 2.10	0.154 ± 0.10	0.079 ± 0.07
Real World - Medium	T-CBF	**10/10**	18.70 ± 1.10	**0.154** ± 0.13	**0.087** ± 0.07
	TAL	9/10	**17.39** ± 2.27	0.166 ± 0.24	0.091 ± 0.04
	WM-VCT	8/10	18.16 ± 2.10	0.173 ± 0.20	0.102 ± 0.06
Real World - High	T-CBF	**10/10**	17.92 ± 1.08	**0.176** ± 0.10	**0.115** ± 0.03
	TAL	7/10	16.99 ± 2.27	0.189 ± 0.14	0.131 ± 0.02
	WM-VCT	6/10	**14.16** ± 1.10	0.197 ± 0.20	0.153 ± 0.08

TABLE I: Performance metrics for various planners across different tasks (best in bold).

provide traversability safety when navigating unstructured, vertically challenging terrain. While this validation analysis provides useful evidence, it does not constitutes a formal proof. Practical limitations like perception and localization noise and learned neural network architecture may prevent the theoretical grantees in Section IV from fully applying in real world scenarios. To support the practical effectiveness of our learned certificates, the following section presents experimental results demonstrating that the controller consistently operates safely and achieves its objectives across diverse simulated and physical environments.

C. Performance Comparison

After validating T-CBF's traversability safety beyond collision avoidance, we compare its performance with WM-VCT and TAL in both simulation and real-world experiments. Differential locks of V4W are disabled for the real-world experiments.

1) Simulation: For simulation we use Verti-Bench [23], a comprehensive and scalable off-road mobility benchmark designed for extremely rugged, vertically challenging terrain featuring diverse unstructured off-road elements. Built on the high-fidelity multi-physics dynamics simulator Chrono, Verti-Bench captures variations across four orthogonal dimensions, i.e., geometry, semantics, obstacles, and scalability. It provides metrics to compare the performance of different planners on unstructured vertically challenging terrain with different configurations and difficulty levels. We run WM-VCT, TAL, and T-CBF on 30 Verti-Bench environments with terrain difficulty level categorized as low, medium and high. In Each simulation scenario the robot has a specific start and goal position in safe region, and the robot has to navigate the challenging terrain with various surface geometry and semantics to reach the goal location. The success rate in simulation is the percentage of time the robot successfully reaches the goal position. Table I shows the overall success rate, traversal time, mean roll, and mean pitch for low, medium, and high difficulty levels combined. T-CBF outperforms other planners in reaching goal location while remaining safe and maintains mobility. It achieves lower roll and pitch angles with the smallest variance, demonstrating T-CBF's superior safety. It is not as aggressive as TAL and WM-VCT and results in longer traversal time. The performance margin against TAL is small because of excess tire slipping in different terrain surface textures and the distribution shift in the observations of elevation map in simulation as training for T-CBF is done on real world data only.

2) Real World: For real-world evaluation, the planners are tested in the rock testbed designed to create varying terrain complexities. The testbed includes reconfigurable rocks to adjust difficulty levels, spray foam covered with green carpet, and wooden slats to introduce surface variations. Each planner is executed 10 times at each of three difficulty levels, resulting in a total of 90 physical runs. The success criteria in real world testing is defined as the ability to keep robot safe, i.e., the robot can move around by executing control actions and searching for feasible paths towards the goal position. Table I summarizes the experimental results for the three planners on low, medium and high difficulty levels. In the low difficulty environment all planners work well. However, as the complexity of the environment increases, WM-VCT and TAL are unable to generate trajectories to navigate without getting immobilized by either getting stuck or rolling over in many trials. T-CBF outperforms other planners in keeping the robot safe and mobile in all difficult and complex scenarios. It also achieves the smallest and most stable roll and pitch angles. The traversal time of T-CBF planner is longer compared to others because it has to navigate around the unsafe regions to reach goal location.

VI. Conclusions

This work introduces a novel approach for leveraging learned Control Barrier Functions (CBFs) to enable wheeled robots to safely navigate unstructured, vertically challenging terrain. Unlike traditional safety measures that primarily focuse on collision avoidance, the proposed T-CBF framework learns traversability safety from a limited dataset of manually collected runs. Experimental results demonstrate and validate the effectiveness of T-CBF in maintaining both safety and mobility across diverse, vertically challenging environments, showcasing its adaptability to varying terrain configurations and complexities. Given the significant impact of terrain interactions on navigation in unstructured and vertically challenging terrain, incorporating multiple perception inputs can enhance situational awareness and traversability safety. In future work, we aim to extend this framework by integrating observations from camera sensors to enrich terrain perception and further increasing the robustness of T-CBF for field robotics applications.

References

[1] Y. Emam, P. Glotfelter, S. Wilson, G. Notomista, and M. Egerstedt, "Data-driven robust barrier functions for safe, long-term operation," *IEEE transactions on robotics*, vol. 38, no. 3, pp. 1671–1685, 2021.

[2] M. Harms, M. Kulkarni, N. Khedekar, M. Jacquet, and K. Alexis, "Neural control barrier functions for safe navigation," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 10 415–10 422.

[3] 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, pp. 16 322–16 329.

[4] A. Datar, C. Pan, and X. Xiao, "Verti-wheelers: Wheeled mobility on vertically challenging terrain," in *ICRA 2024 Workshop on Resilient Off-road Autonomy*, 2024.

[5] L. Wijayathunga, A. Rassau, and D. Chai, "Challenges and solutions for autonomous ground robot scene understanding and navigation in unstructured outdoor environments: A review," *Applied Sciences*, vol. 13, no. 17, p. 9877, 2023.

[6] N. Wang, X. Li, K. Zhang, J. Wang, and D. Xie, "A survey on path planning for autonomous ground vehicles in unstructured environments," *Machines*, vol. 12, no. 1, p. 31, 2024.

[7] X. Xiao, B. Liu, G. Warnell, and P. Stone, "Motion planning and control for mobile robot navigation using machine learning: a survey," *Autonomous Robots*, vol. 46, no. 5, pp. 569–597, 2022.

[8] 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.

[9] 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 *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 3294–3301.

[10] P. Atreya, H. Karnan, K. S. Sikand, X. Xiao, S. Rabiee, 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, pp. 11 789–11 795.

[11] J. Seo, J. Mun, and T. Kim, "Safe navigation in unstructured environments by minimizing uncertainty in control and perception," *arXiv preprint arXiv:2306.14601*, 2023.

[12] P. Rabiee and J. B. Hoagg, "Guaranteed-safe mppi through composite control barrier functions for efficient sampling in multi-constrained robotic systems," *arXiv preprint arXiv:2410.02154*, 2024.

[13] R. Grandia, A. J. Taylor, A. D. Ames, and M. Hutter, "Multi-layered safety for legged robots via control barrier functions and model predictive control," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 8352–8358.

[14] B. Dai, R. Khorrambakht, P. Krishnamurthy, V. Gonçalves, A. Tzes, and F. Khorrami, "Safe navigation and obstacle avoidance using differentiable optimization based control barrier functions," *IEEE Robotics and Automation Letters*, vol. 8, no. 9, pp. 5376–5383, 2023.

[15] Z. J. Patterson, W. Xiao, E. Sologuren, and D. Rus, "Safe control for soft-rigid robots with self-contact using control barrier functions," in *2024 IEEE 7th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2024, pp. 151–156.

[16] K. Long, C. Qian, J. Cortés, and N. Atanasov, "Learning barrier functions with memory for robust safe navigation," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4931–4938, 2021.

[17] C. Dawson, B. Lowenkamp, D. Goff, and C. Fan, "Learning safe, generalizable perception-based hybrid control with certificates," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 1904–1911, 2022.

[18] H. Abdi, G. Raja, and R. Ghabcheloo, "Safe control using vision-based control barrier function (v-cbf)," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 782–788.

[19] Y. Emam, P. Glotfelter, Z. Kira, and M. Egerstedt, "Safe model-based reinforcement learning using robust control barrier functions," *arXiv preprint arXiv:2110.05415*, 2021.

[20] F. Blanchini, "Set invariance in control," *Automatica*, vol. 35, no. 11, pp. 1747–1767, 1999.

[21] A. Datar, C. Pan, M. Nazeri, A. Pokhrel, and X. Xiao, "Terrain-attentive learning for efficient 6-dof kinodynamic modeling on vertically challenging terrain," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024.

[22] K. Chen, R. Nemiroff, and B. T. Lopez, "Direct lidar-inertial odometry: Lightweight lio with continuous-time motion correction," in *2023 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2023, pp. 3983–3989.

[23] T. Xu, C. Pan, M. B. Rao, A. Datar, A. Pokhrel, Y. Lu, and X. Xiao, "Verti-bench: A general and scalable off-road mobility benchmark for vertically challenging terrain," *arXiv preprint arXiv:2502.11426*, 2025.