Learning to Nudge: A Scalable Barrier Function Framework for Safe Robot Interaction in Dense Clutter

Anonymous Authors

Abstract— **Robots operating in everyday environments must navigate and manipulate within densely cluttered spaces, where physical contact with surrounding objects is unavoidable. Traditional safety frameworks treat contact as unsafe, restricting robots to collision avoidance and limiting their ability to function in dense, everyday settings. As the number of objects grows, model-based approaches for safe manipulation become computationally intractable; meanwhile, learned methods typically tie safety to the task at hand, making them hard to transfer to new tasks without retraining. In this work we introduce Dense Contact Barrier Functions(DCBF). Our approach bypasses the computational complexity of explicitly modeling multi-object dynamics by instead learning a composable, object-centric function that implicitly captures the safety constraints arising from physical interactions. Trained offline on interactions with a few objects, the learned DCBFcomposes across arbitrary object sets at runtime, producing a single global safety filter that scales linearly and transfers across tasks without retraining. We validate our approach through simulated experiments in dense clutter, demonstrating its ability to enable collision-free navigation and safe, contact-rich interaction in suitable settings.**

I. INTRODUCTION

The ability for robots to safely interact with and maneuver through densely cluttered environments is a critical step towards their widespread deployment in everyday, unstructured spaces. Such capabilities could enable household robots to safely navigate through crowded pantries, or allow warehouse robots to restock shelves in tight, shared-use commercial settings. These settings contain dense, dynamic arrangements of objects, often forcing the robot to make physical contact to complete its tasks. For example, a robot retrieving an item from a full pantry may need to gently push aside neighboring items, while a logistics robot may need to make contact with fragile goods without knocking them over.

Research on robot safety has historically focused on collision avoidance. The majority of previous work from classical methods like Hamilton-Jacobi Reachability (HJR) [1], Control Barrier Functions (CBFs) [2] and Artificial Potential Fields (APF) [3] are all predicated on maintaining some minimum distance from all obstacles. Even works focused on complex manipulation often handle safety by incorporating the manipulated object into the set of items with which collision must be avoided [4], [5]. However, this criterion of collision avoidance is fundamentally limiting in cluttered spaces. A home robot focused on strictly avoiding collisions would be paralyzed with the task of reaching any object beyond the first row of items in a pantry. Humans, in contrast, routinely make contact, nudging and shifting objects while implicitly understanding the interactions required to prevent spills or damage. This work aims to bridge this gap, giving robots the capacity to reason about and execute safe contact.

While some research has explored physical interaction, these methods often fail to address the scalability and generalizability required for these dense multi-object scenarios. Approaches that explicitly model the complex, coupled dynamics between the robot and object [6] become computationally intractable as the number of objects grows, due to the combinatorial explosion of potential interactions. Works such as [6], [7] successfully demonstrate safe contact with a single complex deformable object, but fail to provide a clear path to scaling to multi-object systems. Learned methods, such as those in reinforcement learning [8], [9], can handle complex systems but often entangle safety and performance, requiring complete retraining for each new task. This highlights the need for a method that balances performance with safety during these interactions, in a manner that scales to a large, variable number of objects and readily generalizes across diverse arrangements.

In this paper we introduce Dense Contact Barrier Functions(DCBF), object-centric control barrier functions that compute implicit interactions to achieve safe, scalable interaction in dense object environments. The core of our framework is a composable DCBF that is trained offline on interactions with just a few objects. At runtime, this function is evaluated for each object in the scene, and the results are aggregated into a single safety filter. This compositional approach allows our method to scale to environments with an arbitrary number of objects and generalizes across diverse configurations without retraining. Our method avoids the intractability of explicit multi-object interaction modeling and reasoning by learning to implicitly capture the safety constraints arising from complex interactions in an object-centric, scalable manner. Our primary contributions are:

- An implicit interaction CBF formulation that captures the multi-object interaction effects on safety without explicit modeling.
- A composable, object-centric framework that enables scalable safety reasoning for a variable number of objects.
- A high-interaction refinement procedure to reduce the conservativeness of the learned CBF
- Validation of our approach in simulated experiments for safe interaction in densely cluttered scenes.

II. RELATED WORKS

The development of safe autonomous systems is a critical area of research, particularly for robots operating in complex dynamic environments such as households or commercial
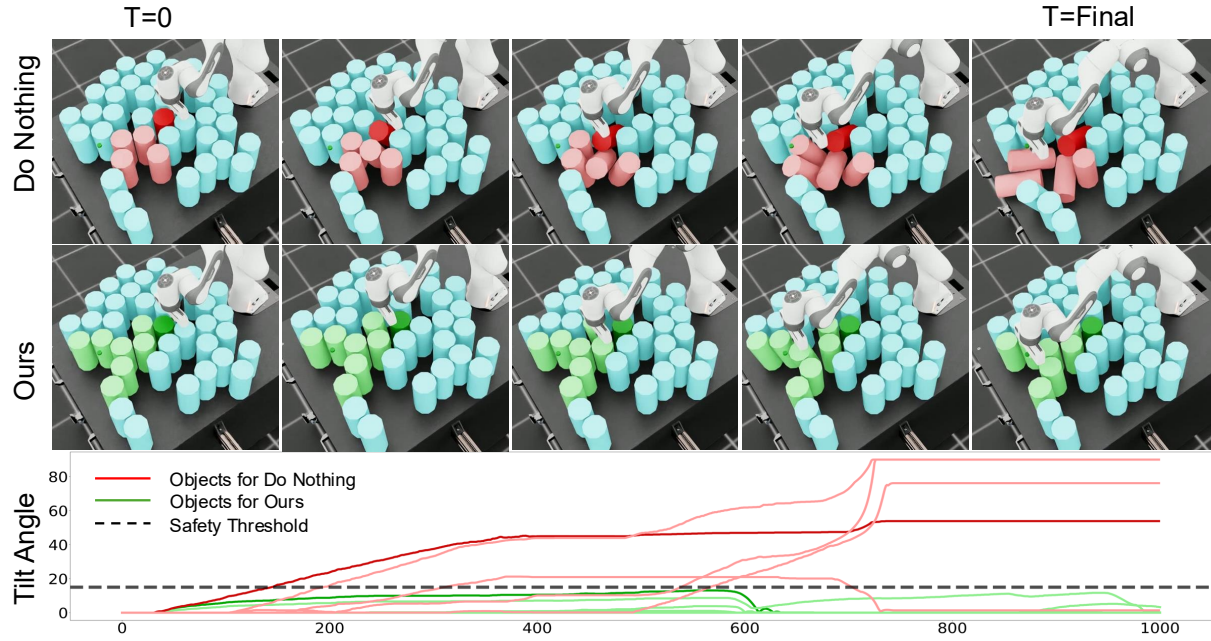
Fig. 1: Simulated Robot trajectory from start to goal across different methods. The first row shows the *Do Nothing* baseline, where the robot directly pushes through the clutter, causing multiple objects to be knocked over. Interacted objects are highlighted in red, with the object starting closest to the end-effector shown in dark red and others in light red. The second row shows our proposed method, which safely interacts with objects to reach the goal without violating safety. Interacted objects are highlighted in green, with the starting closest object shown in dark green and others in light green. The third row plots the tilt angles of the interacted objects over time, with curves matching the same color coding (dark/light red and dark/light green) with a dashed line denoting the safety threshold, $15°$. The red curves exceed tilt threshold, indicating safety violations, whereas the green curves remain below the threshold, demonstrating safe interactions.

facilities. While a large body of work exists on ensuring robot safety, much of it has centered on collision avoidance. This paradigm, though effective in open spaces, is overly restrictive in dense settings where robots must make physical contact and interact with objects to complete their tasks. Our work addresses this gap by proposing a scalable method for safe robot–object interaction through composable barrier functions and implicit interaction modeling.

Safe Autonomy for Collision Avoidance: Robot safety has traditionally been synonymous with avoiding collisions. Classical methods like Hamilton-Jacobi Reachability (HJR) compute safety value functions that define control invariant safe sets [1]. These functions have been used online for optimal control and as minimally invasive safety filters [10]. However, HJR suffers from the curse of dimensionality and requires analytical models, limiting its use to low dimensional, control affine systems and consequently simpler collision avoidance based safety. Recent works like [11], [12] extend HJR using learning-based approximations, alleviating dimensionality restrictions but they still depend on known analytic models, limiting their scalability to environments with unknown dynamic object interactions.

This focus on collision avoidance persists even in manipulation tasks where object interaction is the goal [13]. Classical methods such as potential fields [3], [14] and

MPC-based planners [15] have been focused on maintaining separation from and minimal contact with obstacles. More recent work in cluttered [5] and deformable object manipulation [4] extends this notion of safe collision avoidance by treating the manipulated object as part of the system, thereby ensuring the object also avoids collisions with the environment. These approaches remain overly conservative for crowded environments where avoiding all contact makes many tasks infeasible.

Challenges in Scalability and Generalization: Methods that do explore more complex systems and even physical interaction often fail to scale to dense object environments. Reinforcement learning approaches have been used to handle complex dynamics, either by integrating safety as a soft constraint [16], [17] or by using Bellman-inspired updates [8], [9]. However, these methods typically entangle safety with task-specific policies, limiting their use as general, task-agnostic safety filters. Other approaches have leveraged advances in simulation technology such as XPBD [18] and IsaacLab [19] to enable planning for complex systems. For instance, [6] uses XPBD with model based planning to manipulate deformables under safety constraints for active sensing. Other approaches like [7] leverage learned latent space dynamics to reason about more complex definitions of safety in deformable manipulation tasks. Methods like [20]

even incorporate additional sensing modalities, like force feedback to safely manipulate a single articulated object. While effective for single-object settings, these methods remain computationally intensive or fail to generalize well to multi-object, cluttered environments. A key requirement is a scalable method that can reason about a variable number of movable objects without being overly conservative.

Control Barrier Functions: To address the challenges presented above, we build upon the framework of Control Barrier Functions (CBFs). A CBF is a function that defines a control invariant safe set in the system's state space, guaranteeing the existence of a control action that keeps the agent in this set [2], [21]. CBFs have been widely used to enforce safety online as minimally invasive safety filters, with extensions such as Robust CBFs (RCBFs) [22] generalizing them to uncertain dynamics.

A key challenge for classical CBFs is their synthesis [23], as it often requires precise system knowledge. This limitation has historically restricted their use to simpler tasks such as collision avoidance. While HJR-based synthesis is possible [10], it suffers from poor scalability. To overcome this, Neural Control Barrier Functions (NCBFs) learn safety constraints directly from data [24], enabling a new class of methods that incorporate perception [25], handle uncertainty [26], and scale to dense environments [27]. Other works continue to build upon this, enabling refining candidate NCBFs with HJR [28]. Works such as [27] have improved the scalability of safety in dense environments, leveraging composable CBFs to enable safe navigation in crowds. Despite this progress, the application of CBFs remains focused on collision avoidance, leaving a critical gap in methods that can manage safe physical interaction.

In this work, we tackle the problem of safe interaction in dense, multi-object environments where physical contact is unavoidable. We extend CBFs beyond collision avoidance by introducing Dense Contact Barrier Functions which reason about per-object safety in a composable manner. This enables scalable and generalizable safety filters that can handle multiple objects in diverse configurations without sacrificing the ability to interact and complete tasks.

III. PRELIMINARIES

A. Problem Statement

We consider the problem of a robot safely operating in environments densely populated with objects, such as a household pantry or the shelves of a store or warehouse. The robot may physically interact with the objects, but must ensure that its actions do not cause the objects to enter an unsafe state. Though our formulation generalizes to a variety of state-based safety constraints, in this paper we focus on the failure condition of objects tipping beyond a prescribed tilt angle. This problem statement captures a variety of relevant-tasks such as a manipulator reaching for a glass in the back of a pantry, a ground robot pushing aside furniture, or warehouse robots reorganizing shelves. In each case, safety is defined in terms of object stability, while the robot must remain effective at task execution.

We begin by formalizing this problem. We define the state of the robot as $\mathbf{r} \in \mathbb{R}^n$ with dynamics

$$\mathbf{r}^{t+1} = f(\mathbf{r}^t, u^t), \tag{1}$$

where $u^t \in \mathbb{R}^a$ is the control input. The environment contains N objects each with a state $\mathbf{o}_i \in \mathbb{R}^m \ \forall i \in [1, N]$. The safety of each object is defined by a function $q_i(\mathbf{o}_i)$, where a violation occurs if this safety cost exceeds a known threshold, i.e. $q_i(\mathbf{o}_i) \geq q_T$.

For a scene with a single object, $\mathbf{o}$, the dynamics are given by: $\mathbf{o}^{(t+1)} = p(\mathbf{r}^t, u^{(t)}, \mathbf{o}^{(t)})$. However for multiple objects the dynamics, given by

$$\begin{aligned} \{\mathbf{o}_1^{(t+1)}, &\mathbf{o}_2^{(t+1)}, \dots, \mathbf{o}_N^{(t+1)}\} \\ &= F(\mathbf{r}^{(t)}, u^{(t)}, \{\mathbf{o}_1^{(t)}, \mathbf{o}_2^{(t)}, \dots, \mathbf{o}_N^{(t)}\}), \end{aligned} \tag{2}$$

are influenced by both the robot and inter-object interactions. A key challenge with reasoning over the interactions in F is that the number of inter-object interactions scales exponentially with the number of objects. Reasoning about the safety of each object through explicit modeling quickly becomes infeasible. Additionally, physics models for contact for dense environments are often too computationally inefficient for use. In this paper we assume access to the forward dynamics model of the robot (1), but do not have runtime access to the physics model (2), that forward simulates the objects.

B. Control Barrier Functions

Consider a discrete-time system with dynamics $\mathbf{x}^{t+1} = f(\mathbf{x}^t, u^t)$ and state space $\mathbf{x} \in \mathbb{R}^n$. We say that a set $\mathcal{C} \subseteq \mathbb{R}^n$ is *control-invariant* if for any $\mathbf{x} \in \mathbb{R}^n$, there is an action u which keeps the subsequent state in the set, i.e. $f(\mathbf{x}, u) \in \mathcal{C}$.

Moreover, we say a function $h : \mathbb{R}^n \to \mathbb{R}$ is a (discrete-time) *control barrier function* (CBF) if for each state $\mathbf{x} \in \mathbb{R}^n$, the condition

$$h(f(\mathbf{x}, u)) - h(\mathbf{x}) \geq -\gamma(h(\mathbf{x}))$$

holds, where $\gamma : \mathbb{R} \to \mathbb{R}$ is any extended class $\mathcal{K}$ function.

We note that the super-level set $\mathcal{C}_h := \{\mathbf{x} \in \mathbb{R}^n \mid h(\mathbf{x}) \geq 0\}$ for a CBF h is control-invariant.

IV. DENSE CONTACT BARRIER FUNCTIONS:

We aim to implicitly encode the effects that the inter-object contact dynamics have on each object into the structure of a learned CBF $B(\mathbf{r}_i^{t+1}, \mathbf{O}_i^t)$, where $\mathbf{r}_i^{t+1}$ and $\mathbf{O}_i^t$ are the object-centric robot state and the object-centric history of the robot-object interaction corresponding to object i. This will result in a task-agnostic safety filter that allows the robot to make safe contact with the objects while moving through the environment. In this section, we will explain three critical intricacies of our learned CBF model: 1) implicit interactions, wherein we implicitly encode inter-object interactions into the CBF, 2) state history, which improves implicit knowledge of interaction dynamics, and 3) object-centric state representation, which allows for composition and generalizability to environments of varying densities.

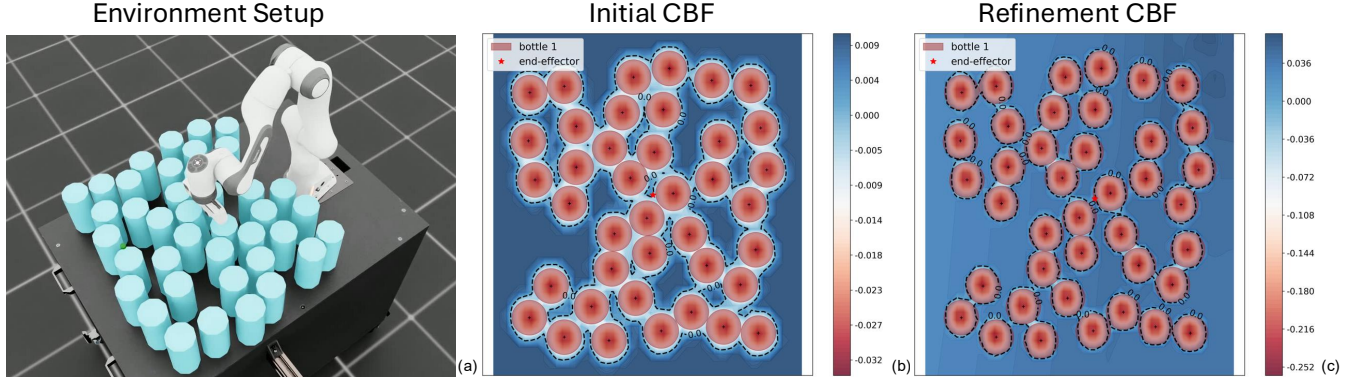| Environment Setup	Initial CBF	Refinement CBF

Fig. 2: Comparison of safe boundaries for the initial model and refined model ($\sigma = 0.01$): (a) Environment snapshot with contact. The end-effector makes slight contact with a bottle, pushing it by a small angle while keeping it within the safe region. (b) Global CBF value plot for the initial model under contact. The conservative boundary incorrectly treats the interaction as unsafe, often leading to the robot stalling. (c) Global CBF value plot for the refined model under contact. The boundary accommodates safe interactions, reflecting less conservative, realistic reasoning about safety for dense environments

A. Implicit interaction

To explain how the robot-object interactions are implicitly encoded in B, we first begin by considering the case of a single-object environment with robot dynamics (1) and object dynamics $\mathbf{o}^{t+1} = p(\mathbf{o}^t, \mathbf{r}^t, u^t)$. As contact dynamics can be difficult to analytically model and computationally inefficient to simulate, we will need to learn a data-driven nominal contact model $\mathbf{o}^{t+1} \approx \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)$. In other words,

$$p(\mathbf{o}^t, \mathbf{r}^t, u^t) = \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t) + e(\mathbf{o}^t, \mathbf{r}^t, u^t). \quad (3)$$

where e represents the error between the true and learned models.

Following [22], we say a function $h(\mathbf{r}^t, \mathbf{o}^t)$ is a *robust CBF* with respect to the learned dynamics $\hat{p}$ if for some $\sigma > 0$, the inequality

$$\sup_{u^t \in \mathbb{R}^m} h\big(f(\mathbf{r}^t, u^t), \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) - h(\mathbf{r}^t, \mathbf{o}^t) - \sigma$$
$$\geq -\gamma(h(\mathbf{r}^t, \mathbf{o}^t))$$

holds for each $\mathbf{r}^t$ and $\mathbf{o}^t$. The parameter σ is called the *robustness margin*. The corresponding *robust feasible control set* is:

$$K_{\text{RCBF}}(\mathbf{r}^t, \mathbf{o}^t) = \Big\{ u \in \mathbb{R}^m | \quad (4)$$
$$h\big(f(\mathbf{r}^t, u^t), \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) - h(\mathbf{r}^t, \mathbf{o}^t) - \sigma \geq -\gamma(h(\mathbf{r}^t, \mathbf{o}^t)) \Big\}.$$

To understand the utility of a robust CBF, consider the following. Suppose we have some robust CBF h for which the model mismatch is bounded above by the robustness margin, i.e.

$$h\big(f(\mathbf{r}^t, u^t), \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) - h\big(f(\mathbf{r}^t, u^t), p(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) \leq \sigma$$

for all $\mathbf{r}^t$, $\mathbf{o}^t$, and u^t. It then follows that

$$\sup_{u^t \in \mathbb{R}^m} h\big(f(\mathbf{r}^t, u^t), p(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) - h(\mathbf{r}^t, \mathbf{o}^t)$$
$$= \sup_{u^t \in \mathbb{R}^m} \Bigg[\underbrace{h\big(f(\mathbf{r}^t, u^t), \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) - h(\mathbf{r}^t, \mathbf{o}^t)}_{\text{nominal}}$$
$$- \underbrace{\Big(h\big(f(\mathbf{r}^t, u^t), \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) - h\big(f(\mathbf{r}^t, u^t), p(\mathbf{o}^t, \mathbf{r}^t, u^t)\big)\Big)}_{\text{mismatch}} \Bigg]$$
$$\geq \sup_{u^t \in \mathbb{R}^m} h\big(f(\mathbf{r}^t, u^t), p(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) - h(\mathbf{r}^t, \mathbf{o}^t) - \sigma$$
$$\geq -\gamma(h(\mathbf{r}^t, \mathbf{o}^t)).$$

Thus if h is a robust CBF with respect to the learned dynamics and the model mismatch is bounded above by the robustness margin, then h is also a CBF with respect to the true dynamics. In particular, any controller $u = k(\mathbf{r}^t, \mathbf{o}^t) \in K_{\text{RCBF}}(\mathbf{r}^t, \mathbf{o}^t)$ guarantees safety, not just under the learned dynamics, but under the true dynamics as well.

It is for this reason that we will seek to learn both the object dynamics and a robust CBF (although we will do so implicitly, as discussed shortly). Further, note that by the mean-value theorem, $h(f(\mathbf{r}^t, u^t), \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)) - h\big(f(\mathbf{r}^t, u^t), p(\mathbf{o}^t, \mathbf{r}^t, u^t)\big) = -\frac{\partial h}{\partial \mathbf{o}}(f(\mathbf{r}^t, u^t), \mathbf{o}^*)e(\mathbf{o}^t, \mathbf{r}^t, u^t)$ for some $\mathbf{o}^*$ between $\hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)$ and $p(\mathbf{o}^t, \mathbf{r}^t, u^t)$. Given a Lipschitz bound L_o on how the CBF value h changes with the object positions, and an error bound $\varepsilon \geq \|e\|$, a sufficient robustness margin is given by

$$\sigma^* := L_0 \varepsilon \geq -\frac{\partial h}{\partial \mathbf{o}}(f(\mathbf{r}^t, u^t), \mathbf{o}^*)e(\mathbf{o}^t, \mathbf{r}^t, u^t)$$
$$\geq h(f(\mathbf{r}^t, u^t), \hat{p}(\mathbf{o}^t, \mathbf{r}^t, u^t)) - h\big(f(\mathbf{r}^t, u^t), p(\mathbf{o}^t, \mathbf{r}^t, u^t)\big).$$

As the model $\hat{p}$ improves, $\|e\|$ and thus σ^* shrinks proportionally. This conceptually shows that as training data becomes more representative of possible interactions, enabling $\hat{p}$ to better capture the dynamics, this sufficient robustness margin σ^* decreases.
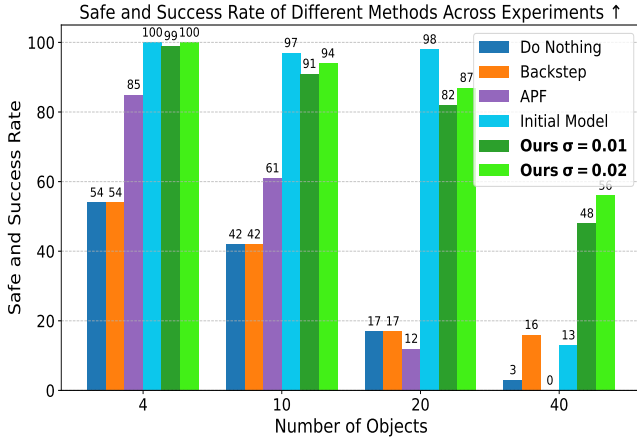
Fig. 3: Safe and success rate of different methods evaluated across environments with increasing object density (4, 10, 20, and 40 objects). This metric illustrates percentage of trajectories where the robot both reached the goal and stayed safe throughout the trajectory. Results are computed over 100 randomized trajectories for each method in each environment. The results highlight the limitations of the baselines, especially at higher object densities and the robustness of our method in maintaining both safety and task success.

Finally, instead of learning a CBF h and the dynamics p explicitly and separately, we propose doing so implicitly and simultaneously by instead learning the function $B(\mathbf{r}^{t+1}, \mathbf{o}^t) \approx h(\mathbf{r}^{t+1}, \mathbf{o}^{t+1})$. Operationally, the function B can be used at runtime as follows. Suppose we have a robot-object state $(\mathbf{r}^t, \mathbf{o}^t)$. Before choosing an action u^t, we determine what the next robot state will be under this action using the robot model, i.e. we evaluate $\mathbf{r}^{t+1} = f(\mathbf{r}^t, u^t)$. Next, we query the value of $B(\mathbf{r}^{t+1}, \mathbf{o}^t) \approx h(\mathbf{r}^{t+1}, \mathbf{o}^{t+1})$ to determine the safety value of the chosen action. If the corresponding robot-object state is safe, i.e. $B(\mathbf{r}^{t+1}, \mathbf{o}^t) \geq 0$, we can safely take the chosen action.

B. Object state history

Inspired by [27], we modify the neural CBF model to take in not just the single-step object state $\mathbf{o}^t$, but a history of the object states over the past T time steps: $\mathbf{O}_{\text{abs}}^t = [\mathbf{o}^t, \mathbf{o}^{t-1}, \ldots, \mathbf{o}^{t-(T-1)}]$. We do so to better encode latent interaction effects (robot-object and object-object) and unobserved attributes (e.g. friction, object mass).

C. Object-centric representation

Implicit interaction modeling also readily generalizes to multiple objects. However, modeling joint multi-object interactions can be intractable due to combinatorial complexity and prohibitive data requirements. Instead inspired by [29], [27] we adopt an object-centric approach, leveraging a change in input representation to learn a single model that can be scaled to any number of objects.

The core of our method is a single implicit interaction Control Barrier Function (CBF), B, that reasons about safety from each object's individual perspective. We achieve this

by transforming the system state into the relative coordinate frame for each object. Given transformation $R(a, b)$ that maps the state a to the relative coordinates of b, we change the model input for object i to be relative to the initial position of object i, at time $t - T$: $(\mathbf{r}_i^t = R(\mathbf{r}^t, \mathbf{o}_i^{t-T}), \mathbf{O}_i^t = R(\mathbf{O}_{\text{abs}}^t, \mathbf{o}_i^{t-T}))$. This representation helps to implicitly infer effect of interaction on each object independently: if an object starts moving to the left when pushed forward, this likely means it is interacting with another object that is affecting its motion. This object-centric formulation is thus scalable and data-efficient, as a single model trained on representative interactions can generalize to complex scenes with an arbitrary number of objects and novel arrangements.

The trained object-centric model, $B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \in \mathbb{R}$, predicts that the interaction with the robot at state, $\mathbf{r}_i^t$, given object history $\mathbf{O}_i^{t-1}$ is safe when $B(\cdot) \geq 0$. We compose N individual object-centric barrier functions into a global scene level CBF that certifies joint safety for all objects:

$$B_{\text{global}}(\mathbf{r}_{1:N}^t, \mathbf{O}_{1:N}^{t-1}) = \min_{i=1,\ldots,N} B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \quad (5)$$

This composition is computationally efficient, scaling linearly with the number of objects, with the individual barrier computation being trivially parallelizable.

V. METHOD IMPLEMENTATION:

In this section we elaborate on the details of how we train and deploy our approach to enable safe, scalable interaction in dense object environments. Our method is implemented in two stages: offline learning of the CBF and online deployment as a safety filter. The offline process consists of: **1) Data Collection:** Collecting interaction-rich data in simulation **2) Initial Training:** Learning an NCBF from the collected data **3) Refinement:** Iteratively refining the learned CBF to reduce conservativeness. The online deployment uses the refined CBF as a safety filter for control synthesis. This section describes the details of each process.

A. Online: Safe Set and Safety Filter

Given the discrete CBF B, the induced control invariant set to keep object i safe is $\mathcal{C} = \{(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) | B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \geq 0\}$. Equivalently the global CBF induces a control invariant set to maintain safety with respect to all objects, $\mathcal{C}_{\text{global}} = \{(\mathbf{r}_{1:N}^{t+1}, \mathbf{O}_{1:N}^t) | B_{\text{global}}(\mathbf{r}_{1:N}^{t+1}, \mathbf{O}_{1:N}^t) \geq 0\}$. At runtime, the global CBF acts as a minimally invasive safety filter. Given B_{global}, a nominal control, u_{nom}, and the set of possible robot actions, $\mathbb{U}$, a constrained optimization problem can be solved to generate the minimally invasive safe control u_{safe} that satisfies the CBF invariance condition:

$$u_{\text{safe}} = \arg \min_{u \in \mathbb{U}} ||u - u_{\text{nom}}|| \; s.t. \quad (6)$$

$$B_{\text{global}}(f(\mathbf{r}_{1:N}^t, u), \mathbf{O}_{1:N}^t) - B_{\text{global}}(\mathbf{r}_{1:N}^t, \mathbf{O}_{1:N}^{t-1})$$
$$\gamma(B_{\text{global}}(\mathbf{r}_{1:N}^t, \mathbf{O}_{1:N}^{t-1})) \geq 0 \quad (7)$$

The above optimization problem may be computationally challenging because we do not assume control-affine dynamics. As a practical alternative we employ the following sampling based approach. A robot's safety can be verified

by checking the global CBF value for a potential next state. A nominal action, u_{nom}, is considered safe if it satisfies $B_{\text{global}}(f(\mathbf{r}_{1:N}^t, u_{\text{nom}}), \mathbf{O}_{1:N}^t) \geq 0$. If the nominal action is unsafe, we can construct a candidate set of actions, $\mathbb{U}$, by sampling around the nominal control, u_{nom}. From the subset of safe actions, $\mathbb{U}_{\text{safe}} = \{u \in \mathbb{U} | B_{\text{global}}(f(\mathbf{r}_{1:N}^t, u), \mathbf{O}_{1:N}^t) \geq 0\}$, we select the one closest to the nominal control: $u_{\text{safe}} = \arg\min_{u \in \mathbb{U}_{\text{safe}}} ||u - u_{\text{nom}}||$. This ensures minimal deviation while maintaining safety.

If not done in parallel, the CBF computation for every object prior to each action can be computationally expensive. In practice, it is often sufficient to only check objects that are either within a fixed distance of the robot or have changed state within their recent history. These objects are most relevant for safety because their interactions with the robot, whether direct or through other objects, could critically impact the robot's safety. This selective evaluation can reduce computational demands without compromising safety.

B. Offline: Data Collection

To effectively learn our CBF with implicit interaction modeling, we begin by collecting interaction-rich data in simple simulated environments with up to N randomly initialized objects. Here N can be much less than the number of objects we want to deploy this model on, as long as it captures most of variations of the robot-object and object-object interactions that will be seen with more objects. The dataset must also be representative, using a control policy that captures both safe and unsafe actions to allow the CBF to properly learn the safety boundary. This process produces transition pairs for each object i: $\{(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}), (\mathbf{r}_i^{t+1}, \mathbf{O}_i^t)\}, \forall i \in [1, \ldots, N]$, each labeled as safe or unsafe depending on whether the latter state satisfies the ground-truth safety criterion. This forms our safe and unsafe dataset, $\mathcal{D}_s$ and $\mathcal{D}_u$ respectively. In addition to the datapoints, we save the corresponding full state of the environment, which is needed for refinement.

C. Offline: Initial Training

We use a neural network to represent B. The network's architecture consists of an LSTM to encode the historical object states and a Multi-Layer Perceptron (MLP) for the robot state. These features are fused and passed through a final MLP to output the scalar barrier value.

The initial training optimizes the parameters, θ, of the NCBF by minimizing the loss function $L(\theta)$:

$$L_s(\theta) = \frac{1}{|\mathcal{D}_s|} \sum_{(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \in \mathcal{D}_s} \left[-B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \right]_+, \quad (8)$$

$$L_u(\theta) = \frac{1}{|\mathcal{D}_u|} \sum_{(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \in \mathcal{D}_u} \left[B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \right]_+, \quad (9)$$

$$L_d(\theta) = \frac{1}{|\mathcal{D}|} \sum_{((\mathbf{r}_i^t, \mathbf{O}_i^{t-1}), (\mathbf{r}_i^{t+1}, \mathbf{O}_i^t)) \in \mathcal{D}} \left[(1-\gamma) B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) - B(\mathbf{r}_i^{t+1}, \mathbf{O}_i^t) + \sigma \right]_+, \quad (10)$$

$$L(\theta) = \eta_s \cdot L_s(\theta) + \eta_u \cdot L_u(\theta) + \eta_d \cdot L_d(\theta). \quad (11)$$

Where $[\cdot]_+$ is the ReLU function. Here, $L_s(\theta)$ enforces that the CBF has positive outputs on safe samples ($\mathcal{D}_s$), $L_u(\theta)$ enforces negative outputs for unsafe samples ($\mathcal{D}_u$), and $L_d(\theta)$ enforces a penalty for violations of the CBF condition ($\mathcal{D} := \mathcal{D}_s \cup \mathcal{D}_u$).

D. Offline: Refinement

The CBF trained with this initial policy is often overly conservative, classifying many states as unsafe due to the suboptimal nature of the initial data-collection policy. This conservativeness can impede the robot's ability to complete its task. Similarly, poor initial training can lead to misclassifications of unsafe states as safe, compromising control invariance of the safe set. To address this, we refine the safety boundary to reduce conservativeness and improve invariance. We achieve this by identifying near boundary samples from the dataset where $|B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1})| \leq \delta$ where δ is a tunable boundary refinement threshold. We then re-simulate these critical scenarios, using the full multi-object scene to capture complex interactions. From these states, we apply the "safest" control, $u_{\text{refinement}}$, which is the action that maximizes the global barrier value:

$$u_{\text{refinement}} = \arg\max_{u \in \mathbb{R}^m} B_{\text{global}}(f(\mathbf{r}_{1:N}^t, u), \mathbf{O}_{1:N}^t) \quad (12)$$

We take up to s refinement steps, using this safest control strategy and collect all the data pairs, relabeling the states as safe or unsafe, based on the resulting safety in the ground truth simulation. This refinement process can be tweaked to solely focus on samples for which $B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \leq 0$ or $B(\mathbf{r}_i^t, \mathbf{O}_i^{t-1}) \geq 0$ to make the CBF more or less conservative, respectively. The CBF is then retrained on this updated dataset. This iterative process can continue until the CBF converges and no misclassifications are found in the dataset, producing our final, control-invariant learned CBF.

VI. EXPERIMENTS AND RESULTS

We evaluate our method in simulation to demonstrate safe, scalable interaction in cluttered environments where physical contact is unavoidable. We designed experiments to assess how performance and generalization change with an increasing number of objects and greater scene complexity.

Experiment Setup: Experiments are conducted in a custom IsaacLab environment containing a Franka Emika Panda arm and several cylindrical bottles shown in Fig. 2(a) The robot is tasked with reaching a random target location without causing any bottle to tip beyond a 15° safety threshold. The robot's end effector is constrained to a 2D (x, y) planar motion and actions are defined as 2D waypoints with a maximum step size of 1cm while training. Each bottle's state is described by its Cartesian coordinates and tipping angle: (x, y, z, θ), with relative transforms computed

by subtracting (x, y) in the cartesian space. We test our approach's robustness by randomizing unobservable physical properties for every bottle in every trial, including mass $(1.3 - 2.0\text{kg})$ and friction coefficients (static: $(0.5 - 0.7)$, dynamic: $(0.3 - 0.49)$). Observable parameters including the bottle radius (5cm) and height (20cm) were fixed. However, observable variations can also be incorporated into our framework through input conditioning.

Safe Trajectory Rates of Different Methods Across Experiments ↑

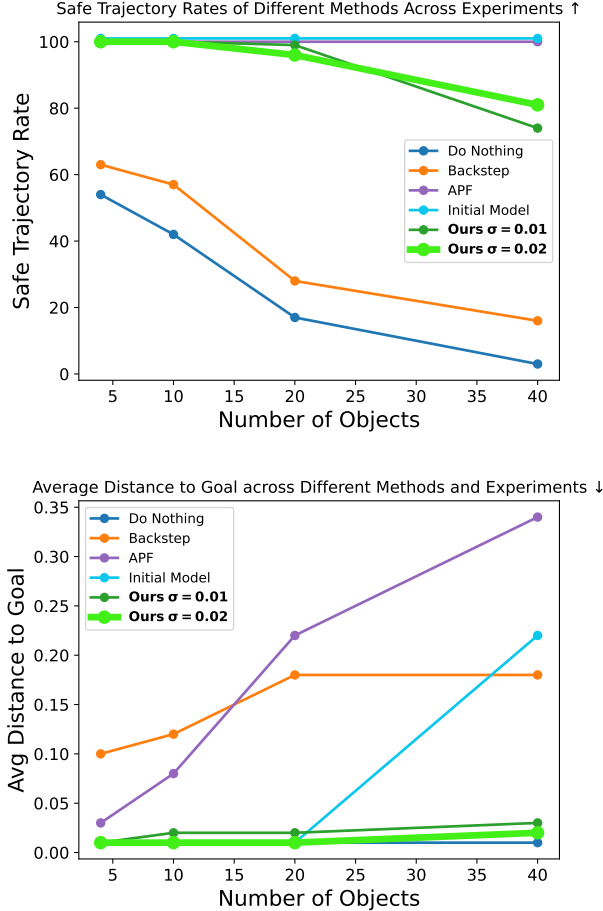Average Distance to Goal across Different Methods and Experiments ↓

Fig. 4: Performance of different methods in environments of varying object densities (4, 10, 20, and 40 objects). The Safe Rate measures the percentage of safe trajectories (higher is better), and the Average Final Distance to Goal measures final distance to the target (lower is better). These metrics were computed over the same experiment shown in Fig. 3, and provide further insight into the performance of each method. These results highlight the limitations of baselines (*Do Nothing*, *Backstepping*) in staying safe, the conservativeness of *APF*, the conservativeness of the *Initial Model* at higher densities, and the robustness of our proposed method, balancing safety and task success.

Training and Refinement: Our CBF is trained in environments containing only 4 bottles. We trained using $\sigma = 0.01$ and $\sigma = 0.02$. We first collected an initial dataset of 9600000 data pairs from 1200 trajectories using a naive

back-stepping policy. This policy steps back to the robot's previous waypoint when nearing the safety threshold. This dataset was balanced, by discarding samples in free space, to ensure a rich mix of both safe and near-unsafe interaction examples. To address the limitations of collecting data solely from a naive controller, specifically the difficulty in obtaining data near the safety boundary, we iteratively refine the model. Using IsaacLab's parallel simulation capabilities, we initialized states with high object interaction, collected new data batches with the refinement steps set to $s = 4$, and fine-tuned the CBF. This process was repeated for 47 batches.

Visualizing the Learned Safety Boundary: To visualize the learned CBF, we generated a global value plot for the 40-object scene in Fig. 2(b)(c). The plot shows that the Initial CBF (before refinement) is overly conservative, creating impassable regions that cause the robot to freeze. In contrast, the refined CBF learns a tighter safety boundary that identifies safe pathways with physical interaction, enabling successful navigation through the dense environment.

Quantitative Evaluation: We evaluated our final model against four baselines in environments with $4, 10, 20$ and 40 objects, where success is defined as reaching the goal while maintaining safety. The baselines include: **Do Nothing:** moves straight towards the goal; **Back-stepping:** the reactive policy used for initial data collection; **Artificial Potential Fields (APF)**: a classical method using repulsive fields; and **Initial:** our CBF model before the refinement procedure. We also compare our method trained with different robustness margins $\sigma = 0.01$ and $\sigma = 0.02$. The Do Nothing Baseline was used as our nominal controller. Back-stepping was tuned to step back for any angle approaching $14°$, and APF parameters were tuned to $K_P = 5.0$ (attractive potential gain), $\eta = 50.0$ (repulsive potential gain), a potential area with length of 1.2m (size of the table), and an oscillation detection length of 3, to allow for some contact without being overly conservative. Fig. 1 depicts samples from trajectories using our method (Ours) and the Do Nothing baseline, along with the tilt angles for the highlighted objects of interest. The baseline method violates safety for multiple objects, while our method keeps all objects below the safety threshold.

The quantitative results in Fig. 3 show that our proposed method consistently outperforms all baselines in safety and task success, especially as object density increases. This metric measures the percentage of trajectories where the robot is both safe and reaches its target. The results in Fig. 4 break down the safe rate and average final distance to goal for each method. The naive Do Nothing and Back-stepping approaches fail to adequately consider the effect of robot-object interactions, regularly violating safety and causing objects to tip. Conversely, APF and our unrefined initial model are overly conservative. While safe, they freeze further from the goal in dense scenes, leading to high task failure rate and large final distance from goal. Our final refined method strikes a necessary balance, achieving high reach success rates and safety rates when maneuvering through dense object arrangements. Comparing $\sigma = 0.01$ and $\sigma = 0.02$ we see that under-approximating the safety margin can potentially

lead to safety violations. In principle, an over-approximation could lead to excessive conservativeness. Even when our method violates, the average maximum tilt angles in failed trajectories were $17.65°$ and $18.88°$ for the 20 and 40 object cases respectively, showing only minor violations, while the 4 and 10 objects cases remained entirely safe (reported for $\sigma = 0.02$). In comparison, the baseline method Do Nothing had much larger violations of $66.81°, 68.45°, 68.16°$, and $46.40°$. Notably, despite being trained on only 4 object scenarios, our model generalizes effectively to the complex 40 object environment, showcasing its scalability and performance.

VII. CONCLUSION

In this work, we introduced Dense Contact Barrier Functions, a scalable framework that enables safe robot interaction in densely cluttered environments. Our approach successfully generalizes from training on a few objects to complex scenes with many objects, without requiring explicit modeling of intractable multi-object dynamics. While our results are promising, several exciting directions for future work remain. We plan to validate our method's ability to handle more diverse objects, including those with varying observable parameters. Furthermore, we aim to extend our method to more complex scenarios involving unconstrained, full-body robot motion, deformable manipulation and, ultimately, demonstrate its effectiveness on physical hardware. This research marks a critical step toward deploying robots that can safely and competently maneuver through the unstructured, contact-rich environments of the real world.

REFERENCES

[1] S. Bansal, M. Chen, S. Herbert, and C. J. Tomlin, "Hamilton-jacobi reachability: A brief overview and recent advances," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. IEEE Press, 2017, p. 2242–2253. [Online]. Available: https://doi.org/10.1109/CDC.2017.8263977

[2] 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.

[3] M. C. Lee and M. G. Park, "Artificial potential field based path planning for mobile robots using a virtual obstacle concept," in *Proceedings 2003 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM 2003)*, vol. 2, 2003, pp. 735–740 vol.2.

[4] Y. Tang, X. Chu, J. Huang, and K. W. Samuel Au, "Learning-based mpc with safety filter for constrained deformable linear object manipulation," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2877–2884, 2024.

[5] X. Ding, H. Wang, Y. Ren, Y. Zheng, C. Chen, and J. He, "Safety-critical optimal control for robotic manipulators in a cluttered environment," 11 2022.

[6] N. Shinde, X. Liang, F. Liu, Y. Zhang, F. Richter, S. Herbert, and M. Yip, "Jiggle: An active sensing framework for boundary parameters estimation in deformable surgical environments," 07 2024.

[7] K. Nakamura, L. Peters, and A. Bajcsy, "Generalizing safety beyond collision-avoidance via latent-space reachability analysis," 2025. [Online]. Available: https://arxiv.org/abs/2502.00935

[8] O. So, C. Ge, and C. Fan, "Solving minimum-cost reach avoid using reinforcement learning," 2024. [Online]. Available: https://arxiv.org/abs/2410.22600

[9] W. Sharpless, D. Hirsch, S. Tonkens, N. Shinde, and S. Herbert, "Dual-objective reinforcement learning with novel hamilton-jacobi-bellman formulations," 2025. [Online]. Available: https://arxiv.org/abs/2506.16016

[10] A. Begzadic, N. Shinde, S. Tonkens, D. Hirsch, K. Ugalde, M. Yip, J. Cortes, and S. Herbert, "Back to base: Towards hands-off learning via safe resets with reach-avoid safety filters," in *Proceedings of the 7th Annual Learning for Dynamics & Control Conference*, ser. Proceedings of Machine Learning Research, N. Ozay, L. Balzano, D. Panagou, and A. Abate, Eds., vol. 283. PMLR, 04–06 Jun 2025, pp. 1154–1166. [Online]. Available: https://proceedings.mlr.press/v283/begzadic25a.html

[11] S. Bansal and C. Tomlin, "Deepreach: A deep learning approach to high-dimensional reachability," 2020. [Online]. Available: https://arxiv.org/abs/2011.02082

[12] Z. Feng, L. Qiu, and S. Bansal, "Bridging model predictive control and deep learning for scalable reachability analysis," 2025. [Online]. Available: https://arxiv.org/abs/2505.03830

[13] D. Morton and M. Pavone, "Safe, task-consistent manipulation with operational space control barrier functions," 2025. [Online]. Available: https://arxiv.org/abs/2503.06736

[14] D. A. Dsouza, S. Shenoy, M. Wang, and A. R. Chowdhury, "A comprehensive safety architecture for human–robot collaboration in confined workspaces using improved artificial potential field," *Robotica*, vol. 43, no. 4, pp. 1373–1393, apr 2025.

[15] J. Lee, M. Seo, A. Bylard, R. Sun, and L. Sentis, "Real-time model predictive control for industrial manipulators with singularity-tolerant hierarchical task control," 2022.

[16] J. Achiam, D. Held, A. Tamar, and P. Abbeel, "Constrained policy optimization," 2017. [Online]. Available: https://arxiv.org/abs/1705.10528

[17] S. Gu, B. Sel, Y. Ding, L. Wang, Q. Lin, M. Jin, and A. Knoll, "Balance reward and safety optimization for safe reinforcement learning: A perspective of gradient manipulation," 2025. [Online]. Available: https://arxiv.org/abs/2405.01677

[18] M. Macklin, M. Müller, and N. Chentanez, "Xpbd: position-based simulation of compliant constrained dynamics," in *Proceedings of the 9th International Conference on Motion in Games*, 2016, pp. 49–54.

[19] M. Mittal, C. Yu, Q. Yu, J. Liu, N. Rudin, D. Hoeller, J. L. Yuan, R. Singh, Y. Guo, H. Mazhar, A. Mandlekar, B. Babich, G. State, M. Hutter, and A. Garg, "Orbit: A unified simulation framework for interactive robot learning environments," *IEEE Robotics and Automation Letters*, vol. 8, no. 6, pp. 3740–3747, 2023.

[20] L. Wei, J. Ma, Y. Hu, and R. Zhang, "Ensuring force safety in vision-guided robotic manipulation via implicit tactile calibration," 2024.

[21] E. Shakhesi, A. Katriniok, and W. P. M. H. M. Heemels, "Counterexample-guided synthesis of robust discrete-time control barrier functions," *IEEE Control Systems Letters*, vol. 9, p. 1574–1579, 2025. [Online]. Available: http://dx.doi.org/10.1109/LCSYS.2025.3578971

[22] A. Alan, T. G. Molnar, A. D. Ames, and G. Orosz, "Parameterized barrier functions to guarantee safety under uncertainty," *IEEE Control Systems Letters*, vol. 7, pp. 2077–2082, 2023.

[23] K. Garg, J. Usevitch, J. Breeden, M. Black, D. Agrawal, H. Parwana, and D. Panagou, "Advances in the theory of control barrier functions: Addressing practical challenges in safe control synthesis for autonomous and robotic systems," 2023. [Online]. Available: https://arxiv.org/abs/2312.16719

[24] S. Liu, C. Liu, and J. Dolan, "Safe control under input limits with neural control barrier functions," 2022. [Online]. Available: https://arxiv.org/abs/2211.11056

[25] S. Raja, A. Kumar, and P. Gupta, "Ogm-cbf: Occupancy-grid map based control barrier functions for safe robot control," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, 2024, pp. 5678–5685.

[26] M. Tayal, H. Zhang, P. Jagtap, and A. Clark, "Learning a formally verified control barrier function in stochastic environment," 2024.

[27] H. Yu, C. Hirayama, C. Yu, S. Herbert, and S. Gao, "Sequential neural barriers for scalable dynamic obstacle avoidance," 2023. [Online]. Available: https://arxiv.org/abs/2307.03015

[28] S. Tonkens and S. Herbert, "Refining control barrier functions through hamilton-jacobi reachability," 2022. [Online]. Available: https://arxiv.org/abs/2204.12507

[29] N. U. Shinde, J. Johnson, S. Herbert, and M. C. Yip, "Object-centric representations for interactive online learning with non-parametric methods," 2023. [Online]. Available: https://arxiv.org/abs/2307.10063