DiffCVaR: Reinforcement Learning for Risk Adaptation via Differentiable CVaR Barrier Functions

Xinyi Wang[1], Taekyung Kim[1], Bardh Hoxha[2], Georgios Fainekos[2] and Dimitra Panagou[1,3]

***Abstract*— Planning through crowded environments under uncertain obstacle motions remains difficult, as stochastic interactions often induce overly conservative behavior or reduced efficiency. To address this challenge, we propose an end-to-end risk adaptation framework for crowd navigation under obstacle-motion uncertainty modeled by a Gaussian mixture model. The framework combines reinforcement learning with a differentiable quadratic-program safety layer based on a Conditional Value-at-Risk barrier function, jointly learning nominal control input, risk level, and safety margin and enforcing explicit probabilistic safety constraints. This design enables context-aware adaptation, promoting efficient behavior while invoking caution only when necessary. We conduct extensive evaluations in dynamic, uncertain, and crowded environments across varying obstacle densities, and also assess generalization under three out-of-distribution cases. Comparisons across optimization-based, RL-based, and integrated RL and optimization methods are provided, and the proposed method is shown to deliver the strongest overall performance in safety and generalization under uncertainty. [Paper Page][1] [Code] [2]**

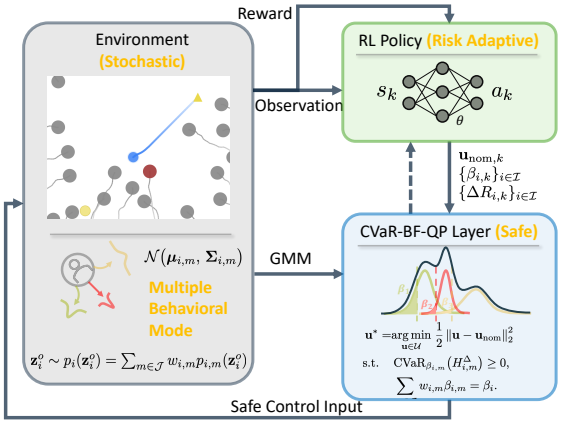

Fig. 1: Overview of DiffCVaR. The policy learns nominal control, risk level, and safety margin, while the differentiable GMM-based CVaR-BF-QP layer produces the applied safe control.

I. Introduction

Safe and efficient robot navigation in crowded environments under uncertain obstacle motion remains challenging. A key difficulty is achieving high efficiency while maintaining safety without excessive conservatism. Risk-neutral objectives optimize expected cost but may overlook dangerous tail events [1], whereas risk-averse robust optimization methods account for worst-case scenarios over distributions [2] potentially at the price of excessive conservatism.

Recent work has adopted Conditional Value-at-Risk (CVaR), a tail-sensitive risk measure that provides a tunable balance between safety and efficiency [3]–[5]. Motivated by the deterministic guarantees of Control Barrier Functions (CBFs), several works combine CBFs with CVaR to enforce probabilistic safety under uncertainty [6]–[8]. However, CVaR barrier function (CVaR-BF) methods typically use a fixed risk level throughout the trajectory, limiting adaptability in dense crowds: conservative settings can cause overly cautious behavior or even infeasibility, whereas aggressive settings improve progress at the expense of safety. Prior work partially addresses this limitation by adapting the risk level online within the CVaR-BF framework [9]. While promising, this approach relies on heuristics, and its performance in more complex settings, such as constrained dynamical robot models and multiple obstacle behavioral modes, remains unclear. These limitations motivate a learning-based method, since adaptive risk parameters are difficult to design analytically across diverse stochastic crowd interactions.

Recently, reinforcement learning (RL) has shown advantages for adaptive decision making, since context-dependent control strategies can be learned directly from interaction data [10], [11]. In addition, RL has demonstrated strong performance by learning diverse and socially compliant behaviors in highly dynamic environments [12]–[14]. However, in safety-critical settings, RL policies typically provide only empirical safety and can degrade under distribution shift, such as denser crowds or previously unseen behaviors. To address this limitation, several mechanisms have been proposed to incorporate formal safety guarantees into RL [15]. A common approach is to use external safety modules, such as CBF-based safety filters that modify nominal policy outputs to satisfy safety constraints [16], or model-predictive shielding methods that verify policy actions using predictive models and replace unsafe actions when necessary [17], [18]. These methods can improve safety at inference time, but because the safety module is not jointly optimized with the pre-trained RL policy, the resulting closed-loop behavior can remain suboptimal. Instead, we adopt an integrated differentiable optimization layer that enables gradients to pass through this safety layer during end-to-end learning [19]. Closely related work such as [20]–[22] adopts a differentiable CBF layer, but assumes deterministic obstacle motion.

To address these gaps, we propose an end-to-end framework for safe navigation that integrates learning with a

[1]Department of Robotics, [3]Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, 48109, USA {taekyung, xinywa, dpanagou}@umich.edu

[2]Toyota Motor North America, Research & Development, Ann Arbor, MI, 48105, USA <first_name.last_name>@toyota.com

[1]Project page: https://lawliet9666.github.io/rlcvarbf/

[2]Code: https://github.com/Lawliet9666/DiffCVaR.gitF

differentiable safety layer under uncertain obstacle motion. The main contributions are as follows:

- An end-to-end risk-adaptive framework in which an RL policy jointly learns nominal control input, risk level, and safety margin for context-aware adaptation.
- A differentiable CVaR-BF quadratic program (CVaR-BF-QP) safety layer under Gaussian mixture model (GMM) obstacle-motion uncertainty, yielding tractable quadratic program (QP) constraints with explicit probabilistic safety guarantees. To the best of our knowledge, this is the first tractable QP reformulation of a CVaR-based control barrier function under Gaussian-mixture uncertainty.
- A broad comparative study of optimization-based, RL-based, and integrated RL and optimization methods in challenging dynamic environments, obstacle densities, and out-of-distribution cases, showing that the proposed method achieves the strongest overall performance in safety, efficiency, robustness, and generalization.

II. Related Work

A. Robust Optimization

Robust optimization provides a standard foundation for handling uncertainty and enforcing safety in dynamic environments. For example, [3] developed a sample-based, distributionally robust CVaR safety filter that constructs safe half-spaces from predicted obstacle trajectories for uncertain obstacle avoidance. [4] further incorporated predictive motion uncertainty into a distributionally robust, chance-constrained model predictive control (MPC) framework for safe robot navigation in crowds. CVaR has also been incorporated into CBF frameworks. For example, [6] integrated Kalman filtering and worst-case CVaR to derive risk-aware safety constraints for discrete-time stochastic systems. More recently, [7] combined cooperative sensing with a CVaR-BF-QP safety filter for dynamic obstacle avoidance. These methods provide strong risk-aware safety guarantees, but they typically rely on fixed risk specifications and can become conservative in dense, highly dynamic interactions.

To mitigate this conservatism, [9] introduced a risk-adaptive CVaR-BF that adjusts risk tolerance online and handles general uncertainty via sampling-based CVaR. However, this formulation has two limitations. First, sampling-based CVaR calculation can be computationally demanding. Closed-form CVaR formulations address this issue by improving computational efficiency [3], [6], but they often rely on Gaussian motion assumptions that do not capture multiple behavioral modes. Second, heuristic risk adaptation can be difficult to tune in scenarios involving more complex interactions and multiple obstacle behavioral modes. Learning-based methods address this issue by adapting risk sensitivity from interaction data [10], but they typically provide only empirical safety rather than explicit formal guarantees. These limitations motivate an adaptive CVaR-BF method that is both computationally tractable and capable of providing real-time probabilistic safety under GMM uncertainty.

B. Safe Reinforcement Learning with Optimization

RL has become a prominent approach for learning risk-adaptive and socially compliant navigation behaviors in dynamic crowds [11]–[13]. Prior work has explored attention-based graph representations for modeling crowd interactions [12], [13], adaptive conformal inference with constrained RL for socially aware navigation [13], and confidence-aware robust dynamical-distance constrained RL for uncertain interactions [11]. Despite strong empirical performance, these methods generally do not provide explicit probabilistic safety guarantees.

To improve safety, some approaches augment learned policies with runtime safety filters. For example, policy actions can be filtered through a CBF-QP controller [16] or safeguarded by MPC-based shielding that verifies and replaces unsafe actions when necessary [17], [18]. Although these mechanisms can improve deployment-time safety, the safety layer is typically not jointly optimized with the pre-trained RL policy, which can lead to conservative or sub-optimal closed-loop behavior. A more integrated direction is to embed constrained optimization directly into the learning pipeline through differentiable layers [19]. Prior work leverages differentiable CBF-QP layers to facilitate gradient propagation during training [20]–[22]. Nevertheless, these methods rely on deterministic assumptions and do not explicitly incorporate uncertainty in an end-to-end framework.

III. Preliminaries and Problem Formulation

A. Continuous-Time Control Barrier Functions

Consider the continuous-time control-affine system:

$$\dot{\mathbf{x}}(t) = f\big(\mathbf{x}(t)\big) + g\big(\mathbf{x}(t)\big)\mathbf{u}(t), \tag{1}$$

where $\mathbf{u} \in \mathcal{U} \subset \mathbb{R}^{n_u}$ is the admissible control input and $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^{n_r}$ denotes the robot state. Let $\mathbf{p} \in \mathbb{R}^{n_p}$ denote the robot's position. For notational simplicity, when the time dependence is clear from context, we write $\mathbf{x}$ and $\mathbf{u}$ in place of $\mathbf{x}(t)$ and $\mathbf{u}(t)$. The functions $f : \mathcal{X} \to \mathbb{R}^{n_r}$ and $g : \mathcal{X} \to \mathbb{R}^{n_r \times n_u}$ are both assumed to be locally Lipschitz continuous. Let $h : \mathbb{R}^{n_r} \to \mathbb{R}$ be a continuously differentiable function. The safe set is defined as the superlevel set

$$\mathcal{X}_{\text{safe}} = \{\mathbf{x} \in \mathbb{R}^{n_r} \mid h(\mathbf{x}) \geq 0\}. \tag{2}$$

A set $\mathcal{X}_{\text{safe}} \subset \mathbb{R}^{n_r}$ is forward invariant for (1) under a control input $\mathbf{u}(t) \in \mathcal{U}$ if its solutions starting at any $\mathbf{x}(0) \in \mathcal{X}_{\text{safe}}$ satisfy $\mathbf{x}(t) \in \mathcal{X}_{\text{safe}}, \forall t \geq 0$.

Definition 1 (Control Barrier Function [23])**.** *Let* $\dot{h} = L_f h(\mathbf{x}) + L_g h(\mathbf{x})\mathbf{u}$*, where* $L_f h(\mathbf{x}) = \nabla h(\mathbf{x})^\top f(\mathbf{x})$ *and* $L_g h(\mathbf{x}) = \nabla h(\mathbf{x})^\top g(\mathbf{x})$ *are the Lie derivatives of the function* h *with respect to* f *and* g*. Given the set* $\mathcal{X}_{\text{safe}}$ *in* (2)*, a continuously differentiable function* $h : \mathbb{R}^{n_r} \to \mathbb{R}$ *is a CBF for System* (1) *if there exists an extended class-*$\mathcal{K}_\infty$ *function* α *such that*

$$\sup_{\mathbf{u} \in \mathcal{U}} \left[L_f h(\mathbf{x}) + L_g h(\mathbf{x})\mathbf{u}\right] \geq -\alpha\big(h(\mathbf{x})\big), \quad \forall \mathbf{x} \in \mathcal{X}_{\text{safe}}. \tag{3}$$

Following the CBF guarantee in [23], we define the admissible safe-control set $\mathcal{K}_{\text{CBF}}(\mathbf{x}) := \{\mathbf{u} \in \mathcal{U} \mid L_f h(\mathbf{x}) +$

$L_g h(\mathbf{x})\mathbf{u} + \alpha(h(\mathbf{x})) \geq 0\}$. If h is a CBF and a locally Lipschitz controller $\kappa(\mathbf{x})$ satisfies $\kappa(\mathbf{x}) \in \mathcal{K}_{\text{CBF}}(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{X}_{\text{safe}}$, then the set $\mathcal{X}_{\text{safe}}$ is forward invariant.

B. Probabilistic Constraints

In dynamic-obstacle scenarios, due to the inherent uncertainty of obstacle motion, the safe set in (2) cannot be computed deterministically. Let $\mathcal{I} \coloneqq \{1, \ldots, N_{\text{obs}}\}$ denote the set of obstacle indices, where N_{obs} is the number of obstacles observed. For each $i \in \mathcal{I}$, let $\mathbf{x}_i^o \in \mathbb{R}^{n_o}$ denote the state and $\mathbf{p}_i^o \in \mathbb{R}^{n_p}$ denote the position. For each obstacle, we use a continuously differentiable barrier function $h_i : \mathbb{R}^{n_r} \times \mathbb{R}^{n_o} \to \mathbb{R}$ and define the safe set as

$$\mathcal{C}_i = \{\mathbf{x} \in \mathbb{R}^{n_r} \mid h_i(\mathbf{x}, \mathbf{x}_i^o) \geq 0\}. \tag{4}$$

The multi-obstacle safe set is the intersection $\mathcal{C} = \bigcap_{i \in \mathcal{I}} \mathcal{C}_i$. Throughout the theoritical analysis below, we adopt standard distance-based barrier function

$$h_i = \|\mathbf{p} - \mathbf{p}_i^o\|_2^2 - (R_r + R_i^o)^2 \tag{5}$$

with R_i^o and R_r denoting the radius of obstacle i and the robot, respectively. Following previous work [24], we assume $\mathbf{x}_i^o$ can be observed at the current time, while future obstacle motions are predicted under uncertainty, i.e.,

$$\dot{\mathbf{x}}_i^o = f_i^o\big(\mathbf{x}_i^o\big) + \mathbf{z}_i^o, \tag{6}$$

where $f_i^o : \mathbb{R}^{n_o} \to \mathbb{R}^{n_o}$ denotes a prediction model of obstacle i, and for each fixed t, $\mathbf{z}_i^o$ is a random variable following an estimated distribution $\mathcal{D}_i$. In this work, we instantiate $\mathcal{D}_i$ as a GMM to capture multiple possible obstacle-motion modes. Let $\mathcal{J} \coloneqq \{1, \ldots, M\}$ denote the set of GMM mode indices:

$$\mathbf{z}_i^o \sim p_i(\mathbf{z}_i^o) = \sum_{m \in \mathcal{J}} w_{i,m} p_{i,m}(\mathbf{z}_i^o), \sum_{m \in \mathcal{J}} w_{i,m} = 1. \tag{7}$$

where each mode $p_{i,m}(\mathbf{z}_i^o)$ is a unimodal Gaussian distribution $\mathcal{N}(\boldsymbol{\mu}_{i,m}, \boldsymbol{\Sigma}_{i,m})$ with mean vector $\boldsymbol{\mu}_{i,m}$ and covariance matrix $\boldsymbol{\Sigma}_{i,m}$ induced by the m-th Gaussian component.

Taking the total time derivative of $h_i(\mathbf{x}, \mathbf{x}_i^o)$ gives

$$\dot{h}_i(\mathbf{x}, \mathbf{x}_i^o, \mathbf{z}_i^o) = \nabla_{\mathbf{x}} h_i\big(\mathbf{x}, \mathbf{x}_i^o\big)^\top \dot{\mathbf{x}} + \nabla_{\mathbf{x}_i^o} h_i\big(\mathbf{x}, \mathbf{x}_i^o\big)^\top \dot{\mathbf{x}}_i^o. \tag{8}$$

Thus, at each time t, $\dot{h}_i$ is random and the CBF condition should be formulated as a probabilistic constraint. Let

$$H_i \coloneqq \dot{h}_i(\mathbf{x}, \mathbf{x}_i^o, \mathbf{z}_i^o) + \alpha(h_i(\mathbf{x}, \mathbf{x}_i^o)). \tag{9}$$

We next introduce the probabilistic CBF condition used to enforce safety under obstacle-motion uncertainty.

Definition 2 (Probabilistic CBF)**.** *Let $\beta_i \in (0, 1)$ be the allowable violation probability for obstacle i. The collection of functions $\{h_i\}_{i \in \mathcal{I}}$ is a β_i-probabilistic CBF if there exists an extended class-$\mathcal{K}$ function $\alpha(\cdot)$ such that for each $\mathbf{x} \in \mathcal{C}$ there exists $\mathbf{u} \in \mathbb{R}^{n_u}$ satisfying*

$$\mathbb{P}\left(H_i \geq 0\right) \geq 1 - \beta_i, \quad \forall i \in \mathcal{I}, \tag{10}$$

where $\mathbf{z}_i^o \sim \mathcal{D}_i$ and H_i is defined in (9)*.*

For notational simplicity, the following definitions are stated for a generic random variable H. In the multi-obstacle setting, they are applied to each H_i. Following [25], the associated Value-at-Risk (VaR) of H is

$$\text{VaR}_\beta(H) = \sup_{\zeta \in \mathbb{R}} \left\{\zeta \mid \mathbb{P}\big(H \geq \zeta\big) \geq 1 - \beta\right\}. \tag{11}$$

The CVaR is then defined as follows:

Definition 3 (Conditional Value-at-Risk [25])**.** *The expected tail value of H below the threshold $\text{VaR}_\beta(H)$ is*

$$\text{CVaR}_\beta(H) \coloneqq \mathbb{E}[H \mid H \leq \text{VaR}_\beta(H)]. \tag{12}$$

An equivalent optimization form is given in [25]:

$$\text{CVaR}_\beta(H) = -\inf_{\zeta \in \mathbb{R}} \mathbb{E}\left[\zeta + \frac{(-H - \zeta)_+}{\beta}\right], \tag{13}$$

where $(\cdot)_+ = \max\{\cdot, 0\}$. Note that $\beta \to 1$ corresponds to the risk-neutral case, i.e., $\text{CVaR}_\beta(H) \to \mathbb{E}[H]$; whereas $\beta \to 0$ corresponds to the risk-averse case, i.e., $\text{CVaR}_\beta(H) \to \text{VaR}_\beta(H)$ [26]. Compared with VaR, CVaR satisfies the axioms of a coherent risk measure, which are essential for rational risk assessment [27]. It is also convex [25], which is important in our proposed method, as it enables a tractable convex optimization formulation. The CVaR condition implies the chance constraint [9]:

$$\text{CVaR}_\beta(H) \geq 0 \;\Rightarrow\; \mathbb{P}(H \geq 0) \geq 1 - \beta. \tag{14}$$

C. Problem Formulation

We now formalize the finite-horizon navigation objective: to synthesize a control policy that maintains probabilistic safety under stochastic obstacle motion with multiple behavioral modes while efficiently reaching the navigation goal.

Problem 1. *Given the initial state $\mathbf{x}(0)$, system dynamics* (1)*, obstacle uncertainty distributions $\{\mathcal{D}_i\}_{i \in \mathcal{I}}$, a time horizon $T > 0$, and a target risk $\epsilon \in (0, 1)$, find a task-efficient control policy satisfying*

$$\mathbb{P}\Big(\mathbf{x}(t) \in \mathcal{C},\ \forall t \in [0, T]\Big) \geq 1 - \epsilon, \tag{15}$$

where $\mathbf{x}(t) \in \mathcal{C}$ means $h_i(\mathbf{x}(t), \mathbf{x}_i^o(t)) \geq 0$ for all $i \in \mathcal{I}$.

Directly enforcing the condition in (15) over the continuous horizon $[0, T]$ is intractable. Next, we introduce our method, which uses the probabilistic CBF condition in (10) and reformulates these chance constraints into tractable mode-wise CVaR constraints under the GMM uncertainty model. Proposition 2 then shows that enforcing these constraints with a proper risk level guarantees (15).

IV. Proposed Method

This section presents the proposed risk-adaptive control method for Problem 1. Using the GMM introduced in (7), the resulting chance constraints are reformulated as a CVaR-BF-QP with probabilistic safety guarantees. This CVaR-BF-QP is enforced through a differentiable optimization layer, enabling safe control and end-to-end policy learning.

A. CVaR Constraint under GMM

Since $\dot{h}_i(\mathbf{x}, \mathbf{x}_i^o, \mathbf{z}_i^o)$ in (8) is affine in $\mathbf{z}_i^o$, the H_i in (9) is also Gaussian under each GMM mode, i.e.,

$$H_i \sim p_i(H_i) = \sum_{m\in\mathcal{J}} w_{i,m} p_{i,m}(H_i), \ \sum_{m\in\mathcal{J}} w_{i,m} = 1, \quad (16)$$

where each mode $p_{i,m}(H_i)$ is also a scalar Gaussian distribution $\mathcal{N}(\mu_{i,m}^H, (\sigma_{i,m}^H)^2)$ with mean and variance

$$\begin{aligned}\mu_{i,m}^H &= \nabla_{\mathbf{x}} h_i(\mathbf{x}, \mathbf{x}_i^o)^\top \big(f(\mathbf{x}) + g(\mathbf{x})\mathbf{u}\big) + \alpha(h_i(\mathbf{x}, \mathbf{x}_i^o)) \\ &\quad + \nabla_{\mathbf{x}_i^o} h_i(\mathbf{x}, \mathbf{x}_i^o)^\top \big(f_i^o(\mathbf{x}_i^o) + \boldsymbol{\mu}_{i,m}\big), \\ (\sigma_{i,m}^H)^2 &= \nabla_{\mathbf{x}_i^o} h_i(\mathbf{x}, \mathbf{x}_i^o)^\top \boldsymbol{\Sigma}_{i,m} \nabla_{\mathbf{x}_i^o} h_i(\mathbf{x}, \mathbf{x}_i^o). \end{aligned} \quad (17)$$

We use the linear extended class-$\mathcal{K}$ function $\alpha(h) = \alpha_0 h$, where $\alpha_0 > 0$. In previous work, under a GMM distribution, CVaR can be computed following [28], but it requires a numerical search for the VaR of the GMM, which increases the online computational cost. We instead impose the following mode-wise CVaR, which is a tractable sufficient surrogate for the full GMM CVaR constraint:

$$\mathrm{CVaR}_{\beta_{i,m}}(H_{i,m}) \geq 0, \quad \forall m \in \mathcal{J}. \quad (18)$$

For each $H_{i,m}$, CVaR admits a closed-form expression [29]:

$$\mathrm{CVaR}_{\beta_{i,m}}(H_{i,m}) = \mu_{i,m}^H - \frac{\phi(\Phi^{-1}(\beta_{i,m}))}{\beta_{i,m}} \sigma_{i,m}^H, \quad (19)$$

where ϕ and Φ^{-1} are the standard normal PDF and inverse CDF, respectively. Since $\mu_{i,m}^H$ is affine in $\mathbf{u}$ and $\sigma_{i,m}^H$ is independent of $\mathbf{u}$, each constraint $\mathrm{CVaR}_{\beta_{i,m}}(H_{i,m}) \geq 0$ is a affine inequality in $\mathbf{u}$.

Lemma 1 (Mode-Wise Chance Constraint)**.** *For any Gaussian mode $H_{i,m} \sim \mathcal{N}(\mu_{i,m}^H, (\sigma_{i,m}^H)^2)$, if* (18) *holds for some $\beta_{i,m} \in (0,1)$, then*

$$\mathbb{P}_{p_{i,m}}(H_{i,m} \geq 0) \geq 1 - \widehat{\beta}(\beta_{i,m}) \geq 1 - \beta_{i,m}, \quad (20)$$

where the bound is $\widehat{\beta}(b) \coloneqq \Phi\Big(-\frac{\phi(\Phi^{-1}(b))}{b}\Big)$, $b \in (0,1)$.

Proof. Let $Z \sim \mathcal{N}(0,1)$ and $b \coloneqq \beta_{i,m}$. Under the definition in (12), and since Φ is monotonically increasing,

$$\begin{aligned}\mathrm{CVaR}_b(Z) = -\frac{\phi(\Phi^{-1}(b))}{b} \leq \mathrm{VaR}_b(Z) = \Phi^{-1}(b), \\ \widehat{\beta}(b) = \Phi(\mathrm{CVaR}_b(Z)) \leq \Phi(\mathrm{VaR}_b(Z)) = b.\end{aligned}$$

For $\sigma_{i,m}^H > 0$, the standardized mode $(H_{i,m} - \mu_{i,m}^H)/\sigma_{i,m}^H$ has the same distribution as Z. Equations (18) and (19) give $\frac{\mu_{i,m}^H}{\sigma_{i,m}^H} \geq \frac{\phi(\Phi^{-1}(\beta_{i,m}))}{\beta_{i,m}}$. Thus,

$$\begin{aligned}\mathbb{P}_{p_{i,m}}(H_{i,m} < 0) &= \mathbb{P}_{p_{i,m}}\left(Z < -\frac{\mu_{i,m}^H}{\sigma_{i,m}^H}\right) = \Phi\left(-\frac{\mu_{i,m}^H}{\sigma_{i,m}^H}\right) \\ &\leq \Phi\left(-\frac{\phi(\Phi^{-1}(\beta_{i,m}))}{\beta_{i,m}}\right) = \widehat{\beta}(\beta_{i,m}) \leq \beta_{i,m}.\end{aligned}$$

If $\sigma_{i,m}^H = 0$, (18) implies $\mu_{i,m}^H \geq 0$, so $\mathbb{P}_{p_{i,m}}(H_{i,m} < 0) = 0$. □

Proposition 1 (GMM Mixture Chance Constraint)**.** *Suppose that* (18) *holds for obstacle i and the mode-wise risk allocation satisfies*

$$\sum_{m\in\mathcal{J}} w_{i,m}\beta_{i,m} = \beta_i. \quad (21)$$

Then,

$$\mathbb{P}(H_i \geq 0) \geq 1 - \sum_{m\in\mathcal{J}} w_{i,m}\widehat{\beta}(\beta_{i,m}) \geq 1 - \beta_i. \quad (22)$$

Thus, (10) *holds for obstacle i.*

Proof. By Lemma 1, the law of total probability yields

$$\begin{aligned}\mathbb{P}(H_i \geq 0) &= \sum_{m\in\mathcal{J}} w_{i,m}\mathbb{P}_{p_{i,m}}(H_{i,m} \geq 0) \\ &\geq 1 - \sum_{m\in\mathcal{J}} w_{i,m}\widehat{\beta}(\beta_{i,m}) \\ &\geq 1 - \sum_{m\in\mathcal{J}} w_{i,m}\beta_{i,m} = 1 - \beta_i. \square\end{aligned}$$

Following [30], our implementation uses the uniform mode allocation $\beta_{i,m} = \beta_i$ for all $m \in \mathcal{J}$. Since $\sum_{m\in\mathcal{J}} w_{i,m} = 1$, it satisfies (21), and the weighted tightened bound simplifies to $\sum_{m\in\mathcal{J}} w_{i,m}\widehat{\beta}(\beta_i) = \widehat{\beta}(\beta_i)$.

B. Probabilistic Safety Guarantee

Building on Proposition 1, we formulate the following tractable CVaR-BF-QP. Given a nominal input $\mathbf{u}_{\text{nom}}$, risk levels $\{\beta_i\}_{i\in\mathcal{I}}$, and risk allocations satisfying (21), this yields the following CVaR-BF-QP,

$$\mathbf{u}^* = \arg\min_{\mathbf{u}\in\mathcal{U}} \frac{1}{2}\|\mathbf{u} - \mathbf{u}_{\text{nom}}\|_2^2 \quad \text{s.t.(18) and (21).} \quad (23)$$

We now prove that solving (23) yields the finite-horizon probabilistic safety guarantee in (15) under Assumption 1. In implementation, since the controller is actually applied at discrete sampling instants, we partition the continuous horizon $[0,T]$ into sampling intervals and prove safety recursively over these intervals. Let $t_k = k\Delta t$ for $k = 0, \ldots, K-1$ and $t_K = T$, where $\Delta t > 0$ and $K = \lceil T/\Delta t \rceil$. For each k, define the safety event up to time t_k as

$$\Gamma_k \coloneqq \bigcap_{i\in\mathcal{I}} \{h_i(\mathbf{x}(t), \mathbf{x}_i^o(t)) \geq 0, \ \forall t \in [0, t_k]\}. \quad (24)$$

Assumption 1. *Under the uniform mode allocation used in our implementation, the probabilistic CBF bound in* (22)*, enforced at sampling time t_k, is assumed to remain satisfied over the interval $[t_k, t_{k+1})$, conditioned on Γ_k, i.e., $\mathbb{P}\Big(H_i(t) \geq 0, \ \forall t \in [t_k, t_{k+1}) \,\Big|\, \Gamma_k\Big) \geq 1 - \widehat{\beta}(\beta_{i,k})$.*

Assumption 1 can be verified under standard sampled-data CBF assumptions [31]: (i) zero-order-hold control and (ii) Lipschitz continuity of H_i along trajectories, $|H_i(t) - H_i(t_k)| \leq L_{i,k}|t - t_k|$. These imply that inter-sample safety holds when the sampling interval Δt is sufficiently small that $\mathbb{P}(H_i(t_k) \geq L_{i,k}\Delta t \mid \Gamma_k) \geq 1 - \widehat{\beta}(\beta_{i,k})$. Adapting the discrete-time argument of [32, Proposition 1], we obtain the following continuous-time result.

Proposition 2 (Probabilistic safety over a finite horizon)**.** *Suppose Assumption 1 holds and $\mathbf{u}_k$ is computed by* (23) *at every t_k. If $\mathbb{P}(\Gamma_0) = 1$ and there exists a $\beta \in (0,1)$ such that $\sum_{i\in\mathcal{I}} \beta_{i,k} \leq \beta$, $k \in \{0, \ldots, K-1\}$, then*

$$\mathbb{P}(\Gamma_K) \geq \prod_{k=0}^{K-1} \left(1 - \sum_{i\in\mathcal{I}} \widehat{\beta}(\beta_{i,k})\right) \geq (1-\beta)^K.$$

In particular, for any target horizon risk $\epsilon \in (0,1)$, if $\beta \le 1-(1-\epsilon)^{1/K}$, then $\mathbb{P}(\Gamma_K) \ge 1-\epsilon$.

Proof. According to Proposition 1, the solution of (23) enforces the bound in (22) for every $k \in \{0,\dots,K-1\}$. For each obstacle $i \in \mathcal{I}$, define $E_{k,i} \coloneqq \{H_i(t) \ge 0,\ \forall t \in [t_k, t_{k+1})\}$, and let $E_k \coloneqq \bigcap_{i\in\mathcal{I}} E_{k,i}$. By Assumption 1, $\mathbb{P}(E_{k,i} \mid \Gamma_k) \ge 1-\widehat{\beta}(\beta_{i,k})$ for every $i \in \mathcal{I}$. Thus, by De Morgan's law and the union bound,

$$\begin{aligned}\mathbb{P}(E_k \mid \Gamma_k) &= 1-\mathbb{P}(E_k^c \mid \Gamma_k) = 1-\mathbb{P}\left(\bigcup_{i\in\mathcal{I}} E_{k,i}^c \,\middle|\, \Gamma_k\right)\\ &\ge 1-\sum_{i\in\mathcal{I}} \mathbb{P}(E_{k,i}^c \mid \Gamma_k) \ge 1-\sum_{i\in\mathcal{I}} \widehat{\beta}(\beta_{i,k})\\ &\ge 1-\sum_{i\in\mathcal{I}} \beta_{i,k} \ge 1-\beta.\end{aligned}$$

Define the interval safety events $F_{k,i} \coloneqq \{h_i(\mathbf{x}(t), \mathbf{x}_i^o(t)) \ge 0, \forall t \in [t_k, t_{k+1})\}$ and $F_k \coloneqq \bigcap_{i\in\mathcal{I}} F_{k,i}$. By the CBF theory [23], we have $\Gamma_k \cap E_{k,i} \subseteq F_{k,i}$. Taking the intersection over all $i \in \mathcal{I}$ gives $\Gamma_k \cap E_k \subseteq F_k$. Since $\Gamma_{k+1} = \Gamma_k \cap F_k$, we have

$$\begin{aligned}\mathbb{P}(\Gamma_{k+1} \mid \Gamma_k) &= \mathbb{P}(F_k \mid \Gamma_k) \ge \mathbb{P}(E_k \mid \Gamma_k)\\ &\ge 1-\sum_{i\in\mathcal{I}} \widehat{\beta}(\beta_{i,k}) \ge 1-\beta.\end{aligned}$$

Since Γ_{k+1} augments Γ_k with safety over the entire interval $[t_k, t_{k+1})$, the events $\{\Gamma_k\}_{k=0}^K$ are nested continuous-time safety events. Thus, using $\mathbb{P}(\Gamma_0) = 1$, the chain rule gives

$$\mathbb{P}(\Gamma_K) \ge \prod_{k=0}^{K-1}\left(1-\sum_{i\in\mathcal{I}} \widehat{\beta}(\beta_{i,k})\right) \ge (1-\beta)^K.$$

If $\beta \le 1-(1-\epsilon)^{1/K}$, then $(1-\beta)^K \ge 1-\epsilon$, and hence $\mathbb{P}(\Gamma_K) \ge 1-\epsilon$. Since $t_K = T$, (24) gives the desired safety event over $[0,T]$. □

C. Adaptive CVaR-BF-QP

The proposed optimization in (23) can be implemented as a differentiable safety layer, enabling **end-to-end policy learning** while maintaining **probabilistic safety guarantees**. The resulting closed-loop system is modeled as a Markov decision process and trained with an actor-critic algorithm. At sampling step k, to encode interactions with multiple dynamic obstacles, we represent the observation $s_k \in \mathcal{S}$ as a spatio-temporal graph $\mathcal{G}_k = (\mathcal{V}_k, \mathcal{E}_k)$, following the attention-based method in [33]. Unlike [33], our encoder connects corresponding entities across consecutive observations using temporal edges, providing short-term context for changes in relative geometry and obstacle motion.

Unlike standard RL, the action $a_k \in \mathcal{A}$ in our framework is not the final control input applied to the system. Instead, it is sent into the CVaR-BF-QP layer, which solves for the safe control input $\mathbf{u}_k^*$. The graph encoder produces a robot embedding and individual obstacle embeddings. The actor network parameterized by θ maps the robot embedding through a nominal-control MLP head to the nominal control input $\mathbf{u}_{\text{nom},k}$ and applies shared obstacle MLP heads to each obstacle embedding to produce the risk levels $\{\beta_{i,k}\}_{i\in\mathcal{I}}$ and adaptive safety-margin offsets $\{\Delta R_{i,k}\}_{i\in\mathcal{I}}$. These outputs define $a_k \coloneqq \big(\mathbf{u}_{\text{nom},k}, \{\beta_{i,k}\}_{i\in\mathcal{I}}, \{\Delta R_{i,k}\}_{i\in\mathcal{I}}\big)$, while the critic estimates the state value $V_\omega(s_k)$.

We constrain the learned parameters to $\beta_{i,k} \in (0,1)$, $\sum_{i\in\mathcal{I}} \beta_{i,k} \le \beta$, and $\Delta R_{i,k} \in [\Delta R_{\min}, \Delta R_{\max}]$, where $\beta \in (0,1)$ is a user-defined total risk budget and $\Delta R_{\min}, \Delta R_{\max}$ are the minimum and maximum admissible safety-margin offsets, respectively. At sampling time t_k, the fixed policy output $\Delta R_{i,k} \ge 0$ defines the following barrier function

$$h_{i,k}^{\Delta}(\mathbf{p}, \mathbf{p}_i^o) = \|\mathbf{p}-\mathbf{p}_i^o\|_2^2 - \big(R_r + R_i^o + \Delta R_{i,k}\big)^2. \quad (25)$$

Let $H_{i,m,k}^{\Delta}$ denote the mode-wise CBF expression constructed from $h_{i,k}^{\Delta}$. Define $c_{i,k} \coloneqq 2(R_r+R_i^o)\Delta R_{i,k} + (\Delta R_{i,k})^2$. Since $\Delta R_{i,k} \ge 0$, we have $c_{i,k} \ge 0$ and $H_{i,m,k}^{\Delta} = H_{i,m,k} - \alpha_0 c_{i,k} \le H_{i,m,k}$. Therefore, by CVaR translation equivariance,

$$\begin{aligned}&\mathrm{CVaR}_{\beta_{i,m,k}}\big(H_{i,m,k}^{\Delta}\big) \ge 0 &(26)\\ \implies &\mathrm{CVaR}_{\beta_{i,m,k}}\big(H_{i,m,k}\big) \ge \alpha_0 c_{i,k} \ge 0. &(27)\end{aligned}$$

Hence, (26) implies (18) at t_k and, by Proposition 2, the finite-horizon safety guarantee. Because the CVaR-BF-QP layer is differentiable, gradients from the policy loss $\mathcal{L}_\pi$ backpropagate through the safety layer to the actor parameters θ. At sampling step k, the contribution through the QP solution $\mathbf{u}_k^*$ is

$$\begin{aligned}\frac{\partial \mathcal{L}_\pi}{\partial \mathbf{u}_k^*}\Bigg[&\frac{\partial \mathbf{u}_k^*}{\partial \mathbf{u}_{\text{nom},k}}\frac{\partial \mathbf{u}_{\text{nom},k}}{\partial \theta} + \sum_{i\in\mathcal{I}} \frac{\partial \mathbf{u}_k^*}{\partial \beta_{i,k}}\frac{\partial \beta_{i,k}}{\partial \theta}\\ &+ \sum_{i\in\mathcal{I}} \frac{\partial \mathbf{u}_k^*}{\partial \Delta R_{i,k}}\frac{\partial \Delta R_{i,k}}{\partial \theta}\Bigg].\end{aligned} \quad (28)$$

This enables the policy to optimize task performance while enforcing the safety constraints whenever feasible. In practice, fast-moving, uncooperative obstacles may render constraints infeasible. Therefore, the deployed implementation uses slack variables to maintain QP feasibility.

V. Simulation

A. Simulation Settings

Environment. We evaluate our method in a crowd-navigation simulator on a $12\,\text{m} \times 12\,\text{m}$ workspace with time step $\Delta t = 0.1\,\text{s}$. We train our policy using 20 dynamic obstacles, each modeled as a circle of radius $0.4\,\text{m}$, with speeds sampled from $[0.5, 1.5]\,\text{m/s}$ and randomized initial positions and velocities. Test results are averaged over 10 independent runs (random seeds), each evaluated on 50 episodes with randomized initial conditions.

Obstacle motion follows the Social Force Model (SFM) with uncooperative behavior. Each obstacle uses the three-mode GMM in (7): one forward mode and two lateral-deviation modes. For simplicity, all obstacles use the same GMM, with weights $[w_1, w_2, w_3] = [0.6, 0.2, 0.2]$ and standard deviations $[\sigma_1, \sigma_2, \sigma_3] = [0.1, 0.2, 0.2]$, although the framework supports obstacle-dependent parameters. The forward mode follows the SFM direction, whereas the two

Type	Policy	SR	CR	TR	Avg. Ret.	Traj. Time	Min. Dist.
Opt.	ORCA [34]	52.6	47.4	0.0	1.92 ± 0.24	**5.76 ± 0.13**	0.79 ± 0.09
	CBF-QP [23]	57.0	43.0	0.0	2.25 ± 0.18	6.60 ± 0.13	0.75 ± 0.09
	CVaR-BF-QP [8]	54.6	45.4	0.0	2.11 ± 0.21	6.63 ± 0.14	0.80 ± 0.10
	Adaptive-CVaR-BF [9]	52.0	48.0	0.0	2.01 ± 0.22	6.52 ± 0.15	0.82 ± 0.10
RL	Vanilla RL	58.2	41.8	0.0	2.07 ± 0.20	6.17 ± 0.10	**0.92 ± 0.12**
	CrowdNav++ (const vel) [12]	48.8	51.2	0.0	0.59 ± 0.10	7.84 ± 0.23	0.42 ± 0.05
	CrowdNav++ (inferred) [12]	63.6	36.4	0.0	0.73 ± 0.12	7.70 ± 0.24	0.52 ± 0.06
RL+Opt.	Vanilla RL + SF	65.0	35.0	0.0	1.96 ± 0.22	6.78 ± 0.09	0.90 ± 0.09
	CrowdNav++ (const vel) + SF	63.8	36.0	0.2	0.65 ± 0.12	8.84 ± 0.28	0.53 ± 0.05
	CrowdNav++ (inferred) + SF	68.6	31.4	0.0	0.70 ± 0.14	8.16 ± 0.29	0.60 ± 0.06
	BarrierNet [20]	83.2	16.2	0.6	2.35 ± 0.19	6.51 ± 0.18	0.75 ± 0.09
	Proposed	**97.0**	**2.6**	0.4	**2.38 ± 0.21**	6.86 ± 0.12	0.70 ± 0.08

TABLE I: Comparisons of baseline methods for the unicycle robot model in environments with 20 obstacles.

lateral modes represent sudden left or right deviations. Across multiple obstacles, these modes induce combinatorial uncertainty that can accumulate over time and create tail-risk collisions absent under deterministic obstacle-motion assumptions in prior environments [12], [13]. To isolate the contribution of the CVaR-BF-QP layer, we assume oracle access to obstacle uncertainty, i.e., the simulator provides GMM parameters that are propagated with constant-velocity prediction and then passed to the CVaR computation module. In real deployments, the same interface can be provided by a learned predictor, such as Trajectron++ [24].

Robot models. Consider a unicycle ($\dot{x} = v\cos\theta$, $\dot{y} = v\sin\theta$, $\dot{\theta} = \omega$, $\mathbf{u} = [v, \omega]^\top$), with look-ahead point $\mathbf{p}_\ell = [x + \ell\cos\theta,\ y + \ell\sin\theta]^\top$. Set $v_{\max} = 1.5\,\text{m/s}$ and $|\omega| \le \omega_{\max} = 1.57\,\text{rad/s}$. Collision avoidance with obstacle i uses Eq. (25), with $\mathbf{p} = \mathbf{p}_\ell$ and $R_r + \ell$ as radius for the unicycle [35].

Observation & action space. For each observation frame τ, we define $\bar{o}_\tau = [(\bar{o}_\tau^{\text{rg}})^\top, (\bar{o}_\tau^1)^\top, \dots, (\bar{o}_\tau^{N_{\text{obs}}})^\top]^\top$ with $N_{\text{obs}} = 20$, where the robot-goal and obstacle observations are $\bar{o}_\tau^{\text{rg}} = [(\mathbf{p}_g - \mathbf{p}_\tau)^\top, \mathbf{v}_\tau^\top, \theta_\tau, R_r]^\top \in \mathbb{R}^6$, $\bar{o}_\tau^i = [(\mathbf{p}_\tau - \mathbf{p}_{i,\tau}^o)^\top, (\mathbf{v}_{i,\tau}^o)^\top, R_i^o, m_{i,\tau}]^\top \in \mathbb{R}^6$. Here, $m_{i,\tau} \in \{0, 1\}$ is the activity mask for obstacle i. The observation at step k is then defined as $s_k := [\bar{o}_{k-2}^\top, \bar{o}_{k-1}^\top, \bar{o}_k^\top]^\top$. The graph encoder uses three attention layers with hidden dimension 64 and produces 16-dimensional node embeddings. All observed obstacles are encoded by the policy and $\mathcal{I}_k \subseteq \mathcal{I}$ denotes the $N_{\text{QP}} = 3$ nearest active obstacles selected by the safety layer. As defined in Section IV-C, the policy action parameterizes the differentiable safety layer. The policy outputs $\mathbf{u}_{\text{nom},k} \in \mathbb{R}^2$, corresponding to $[v_k, \omega_k]^\top$. We set $\sum_{i\in\mathcal{I}_k} \beta_{i,k} = \beta = 0.5$ and $\Delta R_{\max} = 1.5(R_r + R_i^o + \Delta R_{\min})$.

Reward. Following [12], the reward gives $+20$ at the goal, -20 on collision, and otherwise a progress term $r_{\text{prog},k} = 2(d_{k-1}^{\text{goal}} - d_k^{\text{goal}})$ with $d_k^{\text{goal}} = \|\mathbf{p}_k - \mathbf{p}_{\text{goal}}\|_2$ and $\mathbf{p}_{\text{goal}}$ denotes the goal position. For unicycle, we also include the rotation penalty $r_{\text{rot},k} = -\alpha_s(\omega_k \Delta t)^2$ and the backward-motion penalty $r_{\text{back},k} = -\alpha_b|v_k|$ if $v_k < 0$, where $\alpha_s, \alpha_b > 0$.

Performance metrics. Each test case j terminates at time $T_j \le T_j^{\max}$ upon reaching the goal neighborhood ($\|\mathbf{p}_j - \mathbf{p}_{\text{goal}}\|_2 < \delta$) or upon collision. Let m_t, m_s, m_c, and m_{to} denote the total, successful, collision, and timeout counts, respectively, with $m_t = m_s + m_c + m_{\text{to}}$, and let $\mathcal{M}$ denote the index set of successful cases. The episodic return is $G_j = \sum_{k=0}^{K_j} \gamma^k r_{j,k}$ where $K_j = \lfloor T_j/\Delta t \rfloor$, and the minimum distance is $d_j^{\min} = \min_{t\in[0,T_j]} \min_{i\in\mathcal{I}} \big(\|\mathbf{p}_j(t) - \mathbf{p}_i^o(t)\| - R_r - R_i^o\big)$. We report five metrics:

$$\text{SR} = \frac{m_s}{m_t},\ \text{CR} = \frac{m_c}{m_t},\ \text{TR} = \frac{m_{\text{to}}}{m_t}\ \text{Avg. Ret.} = \frac{1}{m_t}\sum_{j=1}^{m_t} G_j,$$

$$\text{Traj. Time} = \frac{1}{m_s}\sum_{j\in\mathcal{M}} T_j,\ \ \text{Min. Dist.} = \frac{1}{m_s}\sum_{j\in\mathcal{M}} d_j^{\min}.$$

B. Baselines

Table I compares 12 baselines under 20 obstacles for the unicycle robot model. The evaluated methods are: **Opt.**: ORCA [34] (rule-based reciprocal collision avoidance), CBF-QP [23] (deterministic safety control), CVaR-BF-QP [8] (risk-aware safe control with a fixed risk level), and Adaptive-CVaR-BF [9] (CVaR-BF with adaptive risk tuning). **RL:** Vanilla RL is a standard end-to-end policy that outputs control actions directly from observations without explicit safety mechanisms. CrowdNav++ (const vel) [12] incorporates a socially aware interaction encoder, while CrowdNav++ (inferred) [12] extends this with a modular prediction of obstacle intentions. **RL+Opt.**: We evaluate two RL+optimization variants: post-hoc safety filtering and differentiable-layer integration. In SF baselines, nominal RL actions are projected through (23) to satisfy probabilistic safety constraints. BarrierNet [20] is trained end-to-end with a differentiable CBF-QP layer, but assumes deterministic environments without stochastic obstacle uncertainty.

In general, our method outperforms all baselines on the key metrics, achieving the highest success rate. Although the trajectory time is not the shortest, this is expected as the reward function discourages short yet impolite behaviors. Nevertheless, the proposed method remains efficient, achieving the highest return while maintaining a comfortable distance and comparable trajectory time. Here, we further distill the following key insights from our results.

Q1: How do different categories of methods perform in dynamic environments? Optimization-based methods generally struggle in highly dynamic environments because the underlying control problems can become infeasible, particularly when surrounding obstacles exhibit uncooperative

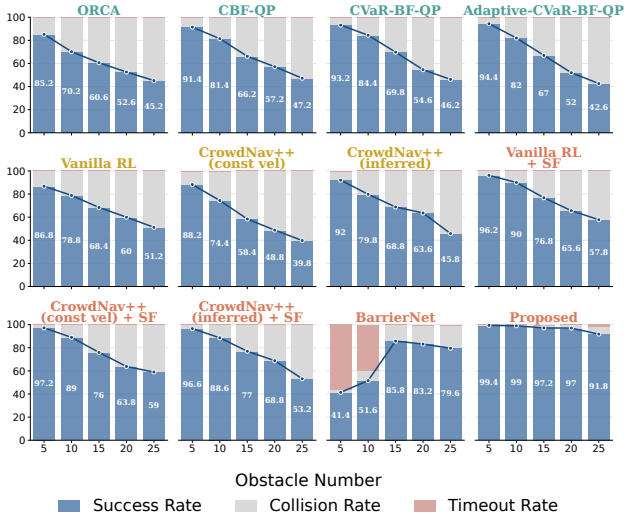

Fig. 2: Success rate versus obstacle number with different methods: optimization (green), RL (yellow), and integrated RL+ Opt. approaches (red).

behavior. In such cases, the robot cannot generate an evasive action, leading directly to collisions and lower success rates. In contrast, pure RL methods improve task performance in some cases but still exhibit relatively high collision rates without explicit safety constraints. Approaches that combine RL with optimization leverage the strengths of both and achieve more reliable overall performance.

Q2: How do different ways of integrating safety into RL affect performance? For RL methods, a post-hoc SF improves the success rate, but remains suboptimal compared to end-to-end safety integration, as it only corrects actions after learning and does not enforce safety during training, potentially degrading policy optimality. In contrast, both the proposed method and BarrierNet [20] jointly optimize the policy and safety constraints through a differentiable safety layer, enabling adaptive behavior. However, unlike BarrierNet, the proposed method explicitly models stochastic uncertainty and learns adaptive risk parameters, which leads to stronger performance.

Q3: How robust are all methods across obstacle densities? Fig. 2 evaluates performance as the number of obstacles increases. As obstacle density grows, all methods exhibit a clear decline in success rate, reflecting the increased difficulty of navigation in dense environments. However, the rate of degradation varies across methods: ours is comparatively more stable, while other methods degrade more rapidly.

C. Generalization

Table II evaluates generalization under out-of-distribution (OOD) scenarios We compare four representative methods spanning optimization-based, RL-based, post-hoc safety filtering, and differentiable-layer approaches.

Q4: How robust are different types of methods under OOD cases? Vanilla RL exhibits a noticeable degradation in performance, often performing worse than optimization-based methods such as CVaR-BF-QP. This suggests that optimization-based approaches provide

Case	Method	SR	CR	TR	Avg. Ret.	Traj. Time	Min. Dist.
I	CVaR-BF-QP	40.4	59.6	0.0	1.16 ± 2.31	6.76 ± 1.98	0.68 ± 0.77
	RL	42.4	57.6	0.0	1.40 ± 2.24	**5.85 ± 1.27**	**0.77 ± 0.73**
	RL + SF	48.2	51.8	0.0	1.53 ± 2.33	6.33 ± 1.47	0.75 ± 0.69
	Proposed	**88.2**	**7.8**	4.0	**2.53 ± 1.99**	16.25 ± 7.36	0.64 ± 0.45
II	CVaR-BF-QP	47.2	52.8	0.0	1.51 ± 2.47	6.87 ± 2.06	0.79 ± 0.82
	RL	49.6	50.4	0.0	1.84 ± 2.27	**6.03 ± 1.32**	**0.86 ± 0.75**
	RL + SF	58.2	41.8	0.0	2.07 ± 2.33	6.43 ± 1.48	0.81 ± 0.72
	Proposed	**95.2**	**3.2**	1.6	**3.10 ± 1.40**	14.47 ± 6.29	0.73 ± 0.55

TABLE II: Generalization performance under two OOD scenarios. Case I: more obstacles (30 obstacles). Case II: increased obstacle radius (0.5 m).

Policy	SR	CR	TR	Avg. Ret.	Traj. Time	Min. Dist.
Policy 1	**97.0**	**2.6**	0.4	3.21 ± 1.26	14.42 ± 6.32	0.77 ± 0.57
Policy 2	92.0	7.6	0.4	3.18 ± 1.49	13.00 ± 5.60	0.77 ± 0.60
Policy 3	85.4	14.0	0.6	2.86 ± 1.83	13.01 ± 6.22	0.73 ± 0.64
Policy 4	84.8	14.4	0.8	2.89 ± 1.88	12.82 ± 6.74	0.73 ± 0.66
Policy 5	80.4	19.6	0.0	**3.31 ± 1.86**	**6.36 ± 1.48**	**0.80 ± 0.75**

TABLE III: Ablation study on the active-set risk allocation $\{\beta_{i,k}\}_{i\in\mathcal{I}_k}$ and adaptive safety margin ΔR. Policy 1: Proposed (adaptive risk allocation, adaptive ΔR). Policy 2: fixed uniform risk allocation ($\beta = 0.5$), adaptive ΔR. Policy 3: adaptive risk allocation, fixed $\Delta R = 0.05$. Policy 4: fixed uniform risk allocation ($\beta = 0.5$), fixed $\Delta R = 0.05$. Policy 5: w/o safety layer (Vanilla RL with graph-based observation).

greater robustness to environmental variations compared to purely learned policies. At the same time, incorporating optimization-based safety mechanisms into RL (e.g., RL with SF) significantly mitigates this degradation, recovering much of the lost performance and bringing it closer to that of optimization-based methods. However, a performance gap still remains. The proposed method consistently outperforms the baselines using SF, achieving higher success rates and returns while maintaining competitive safety. This robustness arises from the end-to-end learning of adaptive risk parameters, which enables the policy to adjust its behavior in response to changing scenarios.

D. Ablation Study

Table III reports an ablation under the same framework that isolates the individual contributions. Success rate degrades monotonically as adaptive components are removed, confirming that each factor contributes independently. This reflects the complementary roles of the two mechanisms: β reactively adjusts how much risk the policy tolerates as perceived danger changes, whereas ΔR proactively expands the physical buffer before an obstacle becomes imminent, giving the robot the space and time to act on that risk assessment. Without the safety layer, the robot collides far more often (CR 19.6%). This demonstrates the effectiveness of our differentiable CVaR-BF-QP layer.

VI. Conclusions

This paper presented an end-to-end risk-adaptive framework for safe robot navigation in uncertain dynamic crowds. By coupling an RL policy with a differentiable CVaR-BF-QP safety layer under a GMM obstacle-motion uncertainty model, the method jointly learns nominal control input, risk level, and safety margin, enabling adaptive conservatism

with probabilistic safety guarantees. Experiments with the unicycle robot show that the method maintains safety while improving efficiency relative to optimization-based, RL-based, and integrated RL with optimization methods. The gains remain evident in the high-density OOD case with 30 obstacles: for the unicycle robot, the proposed method achieves $2.08\times$ the success rate of Vanilla RL.

REFERENCES

[1] M. Black, G. Fainekos, B. Hoxha, and D. Panagou, "Risk-aware fixed-time stabilization of stochastic systems under measurement uncertainty," in *American Control Conference*, 2024, pp. 3276–3283.

[2] S. Xu, H. Ruan, W. Zhang, Y. Wang, L. Zhu, and C. P. Ho, "Distributionally robust chance constrained trajectory optimization for mobile robots within uncertain safe corridor," in *2024 IEEE International Conference on Robotics and Automation*, 2024, pp. 88–94.

[3] S. Safaoui and T. H. Summers, "Distributionally robust CVaR-based safety filtering for motion planning in uncertain environments," in *IEEE International Conference on Robotics and Automation*, 2024, pp. 103–109.

[4] K. Ryu and N. Mehr, "Integrating predictive motion uncertainties with distributionally robust risk-aware control for safe robot navigation in crowds," in *IEEE International Conference on Robotics and Automation*, 2024, pp. 2410–2417.

[5] T. Kim, R. I. Kee, and D. Panagou, "Learning to refine input constrained control barrier functions via uncertainty-aware online parameter adaptation," in *IEEE International Conference on Robotics and Automation*, 2025, pp. 3868–3875.

[6] M. Kishida, "Risk-aware control of discrete-time stochastic systems: Integrating Kalman filter and worst-case CVaR in control barrier functions," in *IEEE 63rd Conference on Decision and Control*, 2024, pp. 2019–2024.

[7] P. Y. Chang, Q. Xu, V. Renganathan, and Q. Ahmed, "Risk aware safe control with cooperative sensing for dynamic obstacle avoidance," *arXiv preprint arXiv:2511.01403*, 2025.

[8] M. Ahmadi, X. Xiong, and A. D. Ames, "Risk-averse control via CVaR barrier functions: Application to bipedal robot locomotion," *IEEE Control Systems Letters*, vol. 6, pp. 878–883, 2021.

[9] X. Wang, T. Kim, B. Hoxha, G. Fainekos, and D. Panagou, "Safe navigation in uncertain crowded environments using risk adaptive CVaR barrier functions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2025, pp. 7669–7676.

[10] J. Choi, C. Dance, J.-E. Kim, S. Hwang, and K.-s. Park, "Risk-conditioned distributional soft actor-critic for risk-sensitive navigation," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 8337–8344.

[11] K. Zhu, T. Xue, and T. Zhang, "Confidence-aware robust dynamical distance constrained reinforcement learning for social robot navigation," *IEEE Transactions on Automation Science and Engineering*, 2025.

[12] S. Liu, P. Chang, Z. Huang, N. Chakraborty, K. Hong, W. Liang, D. L. McPherson, J. Geng, and K. Driggs-Campbell, "Intention aware robot crowd navigation with attention-based interaction graph," in *IEEE International Conference on Robotics and Automation*, 2023, pp. 12 015–12 021.

[13] J. Yao, X. Zhang, Y. Xia, A. K. Roy-Chowdhury, and J. Li, "Towards generalizable safety in crowd navigation via conformal uncertainty handling," in *Conference on Robot Learning*, 2025.

[14] J. R. Han, H. Thomas, J. Zhang, N. Rhinehart, and T. D. Barfoot, "DR-MPC: Deep residual model predictive control for real-world social navigation," *IEEE Robotics and Automation Letters*, 2025.

[15] L. Brunke, M. Greeff, A. W. Hall, Z. Yuan, S. Zhou, J. Panerati, and A. P. Schoellig, "Safe learning in robotics: From learning-based control to safe reinforcement learning," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, no. 1, pp. 411–444, 2022.

[16] Z. Gao, G. Yang, and A. Prorok, "Online control barrier functions for decentralized multi-agent navigation," in *International Symposium on Multi-Robot and Multi-Agent Systems*, 2023, pp. 107–113.

[17] T. Kim, A. D. Menon, A. Trivedi, and D. Panagou, "Backup-based safety filters: A comparative review of backup CBF, model predictive shielding, and Gatekeeper," *arXiv preprint arXiv:2604.02401*, 2026.

[18] A. Banerjee, K. Rahmani, J. Biswas, and I. Dillig, "Dynamic model predictive shielding for provably safe reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 37, pp. 100 131–100 159, 2024.

[19] B. Amos and J. Z. Kolter, "OptNet: Differentiable optimization as a layer in neural networks," in *International conference on machine learning*. PMLR, 2017, pp. 136–145.

[20] W. Xiao, T.-H. Wang, R. Hasani, M. Chahine, A. Amini, X. Li, and D. Rus, "Barriernet: Differentiable control barrier functions for learning of safe robot control," *IEEE Transactions on Robotics*, vol. 39, no. 3, pp. 2289–2307, 2023.

[21] Y. Emam, P. Glotfelter, Z. Kira, and M. Egerstedt, "Safe reinforcement learning using robust control barrier functions," *IEEE Robotics and Automation Letters*, vol. 10, no. 3, pp. 2886–2893, 2025.

[22] C. Wang, X. Wang, Y. Dong, L. Song, and X. Guan, "Multi-constraint safe reinforcement learning via closed-form solution for log-sum-exp approximation of control barrier functions," in *7th Annual Learning for Dynamics & Control Conference*, 2025, pp. 698–710.

[23] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," *2019 18th European Control Conference*, pp. 3420–3431, 2019.

[24] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *European Conference on Computer Vision*. Springer, 2020, pp. 683–700.

[25] S. Sarykalin, G. Serraino, and S. Uryasev, "Value-at-risk vs. conditional value-at-risk in risk management and optimization," in *State-of-the-art decision-making tools in the information-intensive age*. Informs, 2008, pp. 270–294.

[26] P. Akella, A. Dixit, M. Ahmadi, L. Lindemann, M. P. Chapman, G. J. Pappas, A. D. Ames, and J. W. Burdick, "Risk-aware robotics: Tail risk measures in planning, control, and verification," *IEEE Control Systems*, vol. 45, no. 4, pp. 46–78, 2025.

[27] A. Majumdar and M. Pavone, "How should a robot assess risk? towards an axiomatic theory of risk in robotics," in *Robotics Research: The 18th International Symposium ISRR*. Springer, 2019, pp. 75–84.

[28] X. Yang, Y. Hu, H. Gao, K. Ding, Z. Li, P. Zhu, Y. Sun, and C. Liu, "Risk-aware non-myopic motion planner for large-scale robotic swarm using CVaR constraints," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2024, pp. 5784–5790.

[29] M. Norton, V. Khokhlov, and S. Uryasev, "Calculating CVaR and bpoe for common probability distributions with application to portfolio optimization and density estimation," *Annals of Operations Research*, vol. 299, no. 1, pp. 1281–1315, 2021.

[30] K. Ren, H. Ahn, and M. Kamgarpour, "Chance-constrained trajectory planning with multimodal environmental uncertainty," *IEEE Control Systems Letters*, vol. 7, pp. 13–18, 2022.

[31] B. Hoxha, M. Black, K. Maji, H. Okamoto, G. Fainekos, and D. Prokhorov, "Bayesian risk-aware CBFs for discrete-time stochastic systems with learned dynamics," in *American Control Conference*, 2026.

[32] P. Mestres, B. Werner, R. K. Cosner, and A. D. Ames, "Probabilistic control barrier functions: Safety in probability for discrete-time stochastic systems," *arXiv preprint arXiv:2510.01501*, 2025.

[33] T. Kim, R. I. Kee, and D. Panagou, "Learning to adapt control barrier functions under epistemic and aleatoric uncertainty," *arXiv preprint arXiv:2504.03038*, 2026.

[34] J. Alonso-Mora, A. Breitenmoser, M. Rufli, P. Beardsley, and R. Siegwart, "Optimal reciprocal collision avoidance for multiple non-holonomic robots," in *Distributed autonomous robotic systems: The 10th international symposium*. Springer, 2013, pp. 203–216.

[35] S. Wilson, P. Glotfelter, L. Wang, S. Mayya, G. Notomista, M. Mote, and M. Egerstedt, "The Robotarium: Globally impactful opportunities, challenges, and lessons learned in remote-access, distributed control of multirobot systems," *IEEE Control Systems Magazine*, vol. 40, no. 1, pp. 26–44, 2020.